Mesopotamian journal of Cybersecurity
Vol.5,No.2, **pp**. 375–394
DOI: https://doi.org/10.58496/MJCS/2025/024; ISSN: 2958-6542
https://mesopotamian.press/journals/index.php/cybersecurity

Research Article

DAGchains: An Improved Blockchain Structure Based on Directed Acyclic Graph Construction and Distributed Mining

Abbas Mahdi [1,*], Furkan Rabee [1]

[1] *Computer Science Department, Faculty of Computer Science and Mathematics, University of Kufa, Najaf,54001, Iraq.*

ARTICLE INFO

Article History

Received 06 Aug 2024
Revised 10 Feb 2025
Accepted 21 Mar 2025
Published 28 May 2025

Keywords

Blockchain

Directed Acyclic Graph

Consensus Mechanism

Scalability

Mempools

Cryptocurrency

ABSTRACT

Blockchain has revolutionized cryptocurrency and completely changed the management of data and transactions in the digital world because of its decentralized nature, improved transparency, increased security measures, ability to facilitate commercial trading between untrusted parties, and contribution to preventing fraudulent activity. However, the primary issue with blockchain systems is their limited scalability, as they can only process a maximum of 30 transactions per second (TPS), like Ethereum and Bitcoin. In this paper, we introduce an approach using the Nakamoto protocol and the Directed Acyclic Graph (DAG) to develop an improved infrastructure known as a DAGchains that can increase the processing ceiling of the TPS and enable peers to reach Proof of Work (PoW) consensus on a wide scale. Furthermore, a novel allocation for transactions has been presented in miners' mempools on the basis of the Balanced Assignment of Mempool Transactions Protocol (BAMTP), which guarantees the absence of collisions and duplicate transactions, overcomes delays in completing microtransactions, and ensures an efficient distribution of reward fees among all miners. Experimental tests have proven the proposed system's effectiveness in increasing scalability by 24000 TPS compared with the conventional approach without sacrificing the security and decentralization inherent in existing blockchain systems.

1. INTRODUCTION

Blockchain is a secure, decentralized solution that addresses trust, transparency issues, and central authority and eliminates middlemen or third parties. It is a very promising technology that will support the future of a digital financial society.[1]. It has garnered significant interest from academia in a wide variety of fields [2-6]. Blockchain is a decentralized system that uses distributed ledger technology (DLT) to build trust and consensus in P2P networks. In 2009, Satoshi Nakamoto introduced it as a foundational technology for the earliest cryptocurrency, Bitcoin. The decentralization of blockchain is mostly due to its consensus mechanism, which facilitates peer-to-peer trading in a distributed way and utilizes the computational capacity of the entire network to guarantee the immutability and non-tamping of the recorded data [7]. However, blockchain has critical challenges in terms of scalability, cost, unsuitability for real-life scenarios, and efficiency, which hinder its use in applications requiring efficient microtransactions. These limitations significantly affect its ability to adapt to Internet of Things (IoT) applications[8]. Owing to the widespread use of blockchain technology in recent years, it is currently encountering a significant bottleneck: its limited capacity to process a small number of transactions per second (TPS), such as Bitcoin, which processes approximately 7 TPSs, and Ethereum, which does not exceed 15 TPS compared with the high volume of transactions generated by network participants [9].

While the issue of capacity is not the sole concern plaguing existing blockchain networks, cryptographic riddles are becoming increasingly difficult due to the continuous growth in hashing power. Miners with a specific hashing power can mine several blocks and receive rewards. Nevertheless, significant fluctuations in the actual compensation compel the majority of miners to participate in mining pools to smooth their earnings, consolidating computational capacity into a handful of prominent mining pools. Another critical issue is the presence of significant latency due to the duration required for transaction confirmation. Miners, driven by their self-interest, attempt to gather high-fee transactions together in their blocks to maximize their payout. Consequently, the likelihood of executing microtransactions with low fees is slim[10][41]. Blockchain redefines service systems, especially financial ones, provides new alternatives that enhance transparency, reduce costs, eliminate intermediaries, and give individuals greater control over their assets and identities without relying on central institutions and being subject to their laws.

*Corresponding author. Email: abbassf.wahab@uokufa.edu.iq

Therefore, addressing or eliminating these constraints will be a significant milestone in advancing blockchain technology, enabling various applications without compromising or affecting system decentralization and security. This paper explores the concept of upgrading the chain to a directed acyclic graph (DAG) while considering the Nakamoto consensus's characteristics of immutability, decentralization, resistance to attacks, and security. We have introduced a hybrid DLT that combines the features of blockchain with the DAG, which offers high-speed capacities to enhance scalability and increase throughput. The following key points summarize our objectives:

• **Enhanced scalability**: Developing an effective infrastructure for first-layer solutions in blockchain based on the hybrid mechanism of the Nakamoto protocol with a DAG structure to increase scalability, significantly increasing transaction throughput to thousands.

• **Multiple miners**: Implementing the distributed technique in process mining to substitute a single miner with multiple miners to accelerate the conclusion and validation of transactions, aiming to achieve the shortest possible time and acquire the maximum confirmed transactions.

• **Novel reward strategy:** Guarantees carry out microtransactions and distribute all transactions fairly via the BAMTP strategy, ensuring almost equivalent rewards for each miner.

• **Improved computational efficiency:** Distribute the mining power more evenly to keep the resources and not exhaust computational effort in competition and waste time.

The remaining sections of this paper are as follows: Section 2 provides an overview of the existing research on scalability solutions. Section 3 presents the fundamentals of DLT. Section 4 introduces DAGchains as a proposed system, while Section 5 provides the implementation results and discussion. Section 6 summarizes the conclusions of this work.

2. RELATED WORKS

Many companies and research teams have proposed diverse approaches to solve the scalability problems of blockchain. The hierarchical structure of blockchain can be characterized into two essential layers, briefly explained as sharding. The on-chain layer covers the blockchain architecture, including consensus processes, networking protocols, security procedures, and the main-chain data structure. Second layer (off-chain) technologies improve blockchain system scalability and efficiency. Layer 2 solutions execute specific transactions and send complicated computations off-chain to alleviate the restrictions of layer 1 and minimize the load of the core layer, such as high fees, sluggish processing, and limited transaction throughput. There are a few studies related to our search.

Luu et al. [11] proposed the notion of sharding in the realm of blockchain for the first time via their publication. The proposed approach successfully improves the scalability of blockchain systems and overcomes the constraints of conventional blockchain structures by dividing the blockchain network into smaller, separate subsets called shards. Each shard can handle a certain number of transactions and smart contracts independently. Elastico and Zilliqa are two exemplary projects that use the sharding method. [12] Both projects implement proof of work (PoW) as a sharding mechanism and leverage the PBFT algorithm for consensus. Du et al. introduced a new consensus mechanism, known as Mixed Byzantine Fault Tolerance (MBFT), which is specifically designed for consortium blockchains. This mechanism employs sharding technology and a two-layer consensus algorithm to address the issues of scalability and fault tolerance in a consortium-distributed ledger. The experimental results from this work indicated that the MBFT mechanism has robust security, scalability, and a high level of throughput [13].

Wang et al. [10] suggested the use of a structured, directed acyclic graph (DAG) to help peers reach a consensus via proof of work (PoW) on a large scale. This work reduces the likelihood of multiple miners repeatedly handling a transaction via a mempool transaction assignment technique that is based on a DAG structure. The results from this work showed that the proposed methods can enhance the scalability of blockchain systems while maintaining levels of security and decentralization. In 2019, Kwon and E. Buchman [14] proposed an innovative blockchain framework, the Cosmos Network. The new approach comprises numerous autonomous blockchains called zones interconnected via a central hub. The proposed techniques improved the blockchain limitations by enabling secure and efficient transfers of transactions without the requirement of exchanging liquidity across zones. Poon and T. Dryja. [15] introduced an off-chain solution based on theBitcoin Lightning Network. This work allows the provision of atwo-way payment channel between parties. The network significantly reduces transaction costs and processing times because it eliminates the need for all network members to validate each transaction. Xu and colleagues [16] employed an innovative blockchain system called SlimChain to overcome the scalability constraints of current blockchain technologies. The proposed system successfully decreases on-chain storage requirements by 97% ~ 99%, improves transaction speed by 1.4X ~ 15.6X, utilizes off-chain storage for ledger states, and enables parallel transaction processing.

S. Shahriar and Q. H. Mahmoud [17] proposed a parallel proof-of-work approach instead of traditional solo mining to ensure that two or more miners do not put equal effort into solving the puzzle and adding the valid block to the blockchain by

selecting a manager who coordinates the distribution of the hash range (SHA256) among the participants in the mining process. When there are five peers, the scalability increases by 34%, and the mining time decreases as the number of peers increases (mining takes 48.34 minutes to generate 15 blocks at difficulty target 10 with 25 peers). However, the manager is responsible for the distribution and collection of the results, which leads to a single point of failure. N. Sohrabi and Z. Tari [18] suggested the ZyConChain design, which consists of three types of blocks—parent block, side block, and state block—to enable transaction organization. It generates side blocks at a high frequency and stores them in a pool, which facilitates fast processing. Additionally, ZyConChain employs a sharding method that facilitates network expansion, enabling each shard to uphold its state chain and conduct transactions autonomously.

A. Tokhmetov et al.[19] A model that uses a DAG to build a blockchain and manage blocks nonlinearly is proposed, adopting the "Ed25519" twisted Edwards curve as an advanced cryptographic measure to ensure the integrity and security of the data in the network. Consequently, the system executed approximately 9,000 transactions in 0.7 seconds. The central node poses major security risks and does not state the consensus mechanisms used. Chorey and N. Sahu introduced the concept of "checkpoints" in the blockchain as a solution for scalability, where nodes can start from them without processing the entire blockchain history, which leads to faster synchronization, and transactions confirmed after the last checkpoint can be considered more secure, which reduces the number of confirmations needed. However, this approach involves centralization, as a trusted party's creation of this checkpoint exposes the network to significant security threats. The paper does not present any results.[20]. The IOTA Tangle[21]Explored by Serguei Popov, a directed acyclic graph (DAG)-based distributed ledger, has been extensively studied for its unique approach to overcoming the challenges inherent in traditional blockchains. The Tangle's distinctive structure, where each transaction approves two prior transactions, provides inherent scalability, making it particularly promising for applications in the Internet of Things (IoT).

Although existing solutions in the literature attempt to mitigate scalability challenges to some extent, it is important to acknowledge that many may sacrifice blockchain's fundamental characteristics and introduce additional security concerns. Consequently, our study strives for a truly decentralized paradigm that can grow, upholds the security features of the Nakamoto model, and provides a distinctive method for processing transactions and reward distributions among miners.

TABLE I. A GATHERING OF PAPERS ABOUT THE INVESTIGATION OF SCALABILITY SOLUTIONS FOR BLOCKCHAIN TECHNOLOGY

Ref.	Layered Solutions	Proposed scalability	Contributions	limitations
[9] 2019	On-blockchain and DAG	Proof of Work	Theoretically, the proposal increases productivity while maintaining the current blockchain properties.	The paper lacks an overall evaluation and has no actual results
[10] 2016	On-blockchain	Proof of Work, PBFT, and shards	Increase the rate of blocks created compared to Bitcoin.	Due to PBFT and Pow, communication overhead may increase delays and reduce performance.
[12] 2020	On-blockchain	Modified PBFT, shards	The MBFT utilizes sharding technology and a two-layer PBFT consensus algorithm to enhance throughput TPS.	The communication overhead is high due to the PBFT algorithm's reliance on multiple rounds of communication between participating nodes, which increases complexity with increased participation. Consequently, the latency increases, and the transaction completion process slows down.
[13] 2019	Cross-chain "Cosmos"	Tendermint BFT consensus algorithm	enhancing overall throughput and reducing bottlenecks	the architecture of thecosmos relies heavily on the security of the Cosmos Hub, and any infraction in the Hub's security could affect the entire network
[14] 2016	Off-blockchain “BLN”	Payment channels	Smart contracts are used for the creation of a two-way payment channel between parties to enhance the scalability of the blockchain	Less secure and not for much amount of money transfer
[15] 2021	Off-blockchain " SlimChain”	Shard, parallel mechanism	decreases on-chain storage requirements by maintaining on-chain only short commitments, storing transactions, and increasing thespeed of transaction execution off-chain nodes by smart contracts	Complex coordination between off-chain and on-chain parts seems to increase latency, distrust, security vulnerabilities, and transparency
[16] 2020	On-blockchain	parallel proof-of-work, PSO algorithms	A parallel PoW approach and PSO algorithm are used to select the best manager who coordinates the distribution of the hash range (SHA256) among the miners to enhance throughput.	Centralization by the chosen manager may lead to a single-point failure or potential attack.
[18]		"Ed25519" twisted Edwards	uses DAG to build a blockchain and adopts the "Ed25519" twisted Edwards	The central authority of some nodes makes them vulnerable to failure or security threats.

2024	On-blockchain	curve cryptography	curve as an advanced cryptographic measure to ensure the integrity and security of data in the network, and the transaction completion rate is high frequency.	
[19] 2024	On-blockchain	Checkpoints protocol	introduced the concept of "checkpoints" in the blockchain as a solution for scalability	This checkpoint by a trusted party exposes the network to significant security threats. The paper does not present any experimental results
[20] 2019	DAG "IOTA"	Tangle “No consensus”	DAG enables faster concurrent processing and confirmation of transactions, resulting in a higher throughput	the coordinator's presence surrounds the system with central risks, and there are no consensus algorithms applied
Proposed system	On-blockchain and DAG	DAGchains	Integrate the Nakamoto protocol and DAG technology for increased scalability in a fully decentralized environment and novel, equitable, and balanced reward mechanisms.	The proposed system may not align with the IoT concept because it requires a significant amount of computing power, which may not be sufficient for the lightweight resources used in IoT.

3. DISTRIBUTED LEDGER TECHNOLOGY

Distributed Ledger Technology (DLT) is a digital technology that allows for the decentralized and synchronized management of a shared database or ledger among various participants or nodes. Its purpose is to offer a clear, safe, and unchangeable record of transactions or other data types. DLT functions on a decentralized network, where every member possesses a duplicate of the ledger and collectively verifies and modifies its contents via a consensus mechanism [22]The key characteristic of DLT is the decentralization of the ledger over several nodes, obviating the necessity for a central authority or middleman to authenticate and store data. This system's decentralized design improves transparency, security, and resilience by minimizing the possibility of a single point of failure or manipulation. [23]. The participants in a DLT network collaborate to maintain the ledger's integrity, guaranteeing the accuracy and consistency of the recorded data. [6, 24]. DLT commonly uses cryptographic methods to ensure that data security and transactions arekept on the ledger. These methods involve the use of cryptographic hashing, digital signatures, and consensus algorithms to ensure that nodes agree on the order and legitimacy of transactions. [25].

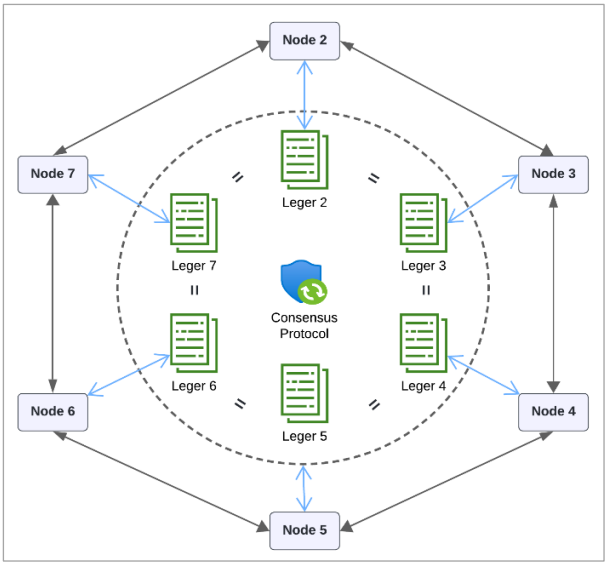

Fig. 1. Distributed Ledger Technology (DLT) platform

3.1 DLT Permissions Models

DLT includes a variety of technologies that allow for the decentralized and synchronized management of shared digital ledgers. Comprehending the applications and implications of DLT in academic and practical contexts requires a thorough understanding of its numerous types and categories to enable the development of innovative solutions. [26, 27]. DLTs can be categorized on the basis of permission models, which determine the network's participants and their access to the ledger. Public Distributed Ledger Technologies (PDLTs), such as Bitcoin and Ethereum, function on an open and permissionless framework. This means that any individual can participate in the network and verify transactions. These systems utilize consensus procedures such as proof of work (PoW) or proof of stake (PoS) to guarantee the ledger's integrity. Only authorized participants can view and participate in private distributed ledger technologies (PrDLTs) such as Corda and Hyperledger

Fabric, setting them apart from other systems. Authorized entities can only join the network and verify transactions, making it perfect for enterprise environments prioritizing privacy, scalability, and access control. Consortiums Distributed Ledger Technologies (CDLTs) are a hybrid category incorporating features from both public and private models. A CDLT involves a collective of organizations or entities responsible for governing the network and upkeeping the ledger. The participants, who possess equal rights and obligations, work together to reach an agreement and maintain the ledger's integrity. R3 Corda, when used in consortium mode, and certain Hyperledger Fabric implementations are examples of CDLTs. By uniting reliable organizations, CDLTs offer a compromise between public transparency and private control. This makes them well suited for situations that demand secure cooperation and data exchange within a predetermined group. [28].

TABLE II. ATTRIBUTES OF DLT PERMISSIONS MODELS [29]

Category	Permissioned	Permissionless
Secure	Less secure	More secure
Decentralized	Partially	Full
Read permission	Could be public or restrict	Public
Efficiency	Low	High
Privacy	Exclusive membership	Transparent and open to all
Cost	cost-effective solution	Highly costly
Immutability	Could be impacted	Extremely difficult to manipulate

3.2 DLT Structure Models

The underlying data structure that DLTs use is another aspect of categorization. Blockchain, the most renowned DLT, employs a series of blocks, each with a roster of transactions. Cryptographic hashes connect the blocks to create an unchangeable ledger. They use an efficient consensus technique to authenticate and add new blocks to the chain [29]. A directed acyclic graph (DAG) is another type of DLT that deviates from the linear structure of a blockchain. DAG-based ledgers depict transactions as nodes within a graph, with edges indicating the interdependencies between transactions. DAG-based systems seek to enhance scalability and expedite transaction processing by facilitating parallel processing rather than following a linear chain structure and eliminating the requirement for conventional miners. This is considered a very efficient approach for IoT infrastructures. Swirlds developed Hashgraph, which uses a unique consensus method and data structure to efficiently validate and sequence transactions. Hashgraph employs a gossip protocol in which nodes swiftly distribute information regarding transactions among each other. By engaging in multiple rounds of gossip and virtual voting, Hashgraph nodes reach a collective agreement on the sequence of transactions, thereby attaining consensus without relying on resource-intensive proof-of-work procedures. The key advantages of this approach over previous DLT alternatives are high throughput, low latency, and fairness in transaction sequencing [30]. Holochain is an innovative framework for DLT that distinguishes itself from its agent-centric architecture and focuses on peer-to-peer networking principles. It grants more independence to network users and ensures data integrity via cryptographic security measures. Within the Holochain framework, every individual node has a hash chain, which contains data pertinent to its engagements with other nodes. The decentralized data management method allows scalability enhancements, as nodes are not required to process and store the complete ledger [31].

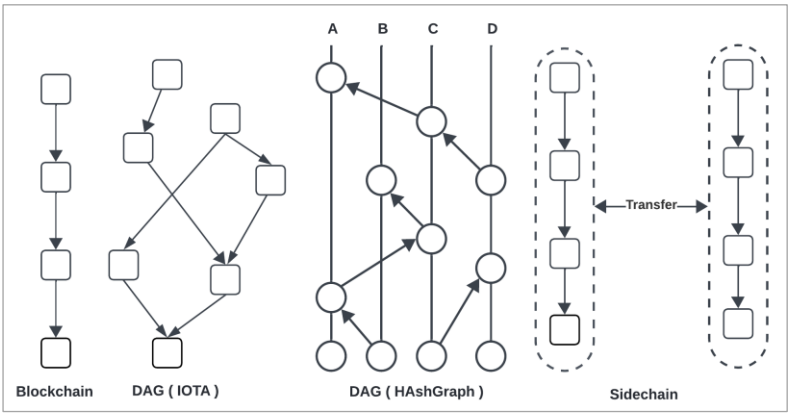

Fig. 2. An overview of structural DLSs [32]

4. DAGCHAINS AS AN IMPROVED PROPOSED DLT

The DAG inspired the proposed system structure concept. The primary motivation for altering the data structure is transitioning from series operations to parallel operations. Unlike chains, it does not need to group all transactions onto a single yarn, allowing several miners to add the transactions concurrently. This section provides a comprehensive description of the planned structure of DAGchains. This framework aims to help miners around the network reach an agreement on specific sets of transactions as independent blocks and add them to the DAGchains.

The main idea of the proposed system is to incorporate a Nakamoto chain into a DAG by creating an extremely interconnected structure. By relying on the known Nakamoto consensus, the presented approach ensures decentralization, security, increased transaction processing speed, and reduced latency. In addition, it offers a systematic distribution of transactions, enabling all honest peers (miners) to construct an identical public ledger once they have achieved consensus without transaction conflict. Fig. 3 illustrates the structure of the DAGchains.

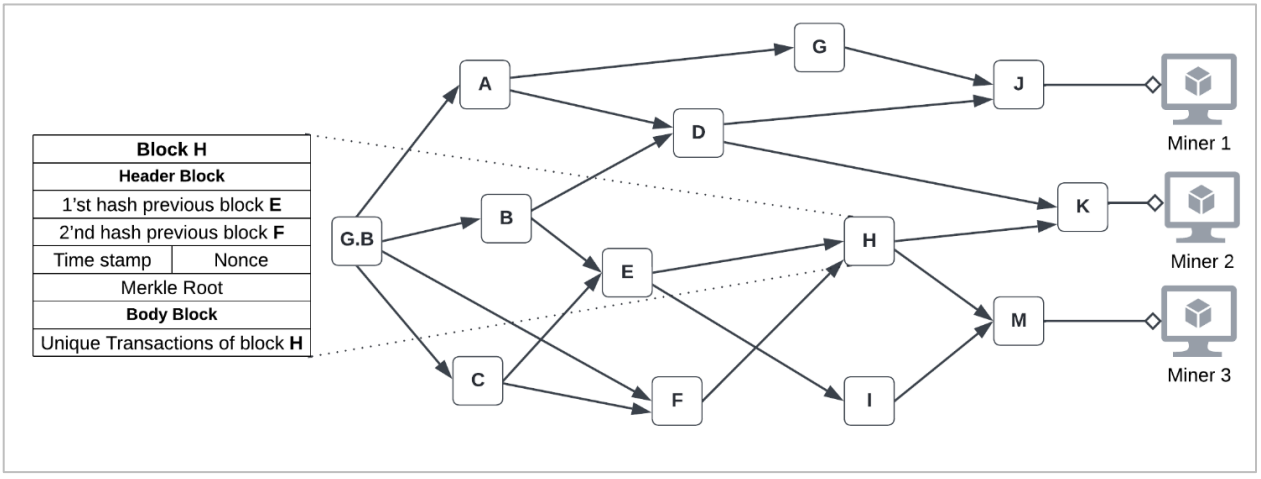

Fig. 3. DAGchains structure for three miners available in the P2P network

Figure 3 presents a DAG-based blockchain structure in which multiple miners, such as Miner 1, Miner 2, and Miner 3, work concurrently to enhance scalability. Each block, such as Block H, has a header containing hashes of previous blocks (E and F), a timestamp, a nonce, and a Merkle root, which secures transaction verification. The genesis block (GB) initiates the structure, branching into subsequent blocks labelled A, B, C, and so forth, with directional arrows indicating their dependencies. For example, blocks B and C stem from G.B., whereas Block H references E and F as their parents, showing the multiparent capability of the DAG structure. With Miner 1 working on Block J, Miner 2 working on Block K, and Miner 3 working on Block M, this framework enables miners to operate on separate branches simultaneously without waiting for previous blocks to complete, as in blockchain. This approach also reduces bottlenecks and enables faster finality, making it well suited for high-demand applications requiring rapid and efficient processing. By enabling concurrent block validation and the ability to connect through multiple parent blocks, the improved structure significantly improves transaction throughput and scalability, making it more efficient than traditional single-chain block systems. Generally, the block size in the DAGchains is one megabyte, mimicking Bitcoin's typical block size and adding many categorized transactions within each block. In addition, an average transaction size of 400 bytes was established, so the number of transactions included within each block does not surpass 3,000 TX_S.

The DAGchains originate from the genesis block, which is free of transactions and has no previous hash information. The workflow for creating the next blocks is the same. When creating a new block, every miner must choose valid transactions from their mempool on the basis of the BAMTP mechanism, which will be discussed later in this section. Every new block must confirm the earliest two previous tip blocks (i.e., those that consumed less time in the mining process) or the genesis block. Suppose that a miner has no prior blocks to establish a connection with the DAGchains via edges to shorten the confirmation delay. In that case, the maximum number of confirmations for each block is 2. If the preceding tip blocks surpass the confirmation threshold, the miner selects the subsequent older blocks as new tips to achieve balance in the growth of the structure and not accumulate the link at a certain point.

Once the blocks are prepared, the miners will hash them via SHA256 and find the nonce. Valid blocks are determined by a specific pattern in the hashing result, such as having all 7 preceding bits equal 0. The blocks are then disseminated among complete nodes in the P2P network to be verified and incorporated into their DAGchains framework. The proposed mechanism enables capacity growth and is considered an extension of blockchain technology. The following are the fundamental elements of the proposed strategy:

4.1 Block

A block is a fundamental component of the DAGchains. Represented by the symbol B, it includes the data required for certain applications, such as Bitcoin and Ethereum cryptocurrencies [33]. It also has extra data in the block header, which includes the version, previous block hash, merge root, timestamp, nonce, difficulty target, and additional metadata. This helps check the integrity and determine where it is in the DAGchains. For practical applications, we use SHA256, a cryptographic hash function represented as H, to identify the block. The IDs are unique identities of some previous blocks in the DAGchains, represented by 64 hexadecimal numbers. We denote the miner responsible for creating this block as M, and the nonce represents the solution to the cryptographic problem. Blocks in graph theory consist of a collection of vertices, with each pointer representing a directed edge. Within the collection of blocks, a distinct block known as the genesis block exists (GB). This block acts as the sole origin of the entire set of DAGchains:

$$B = H(ID', ID'', TXs, M, Nonce, MR) \quad (1)$$

Equation (1), the identities of the previous blocks in the DAGchains are represented by (ID', ID") TXs, which refer to sets of transactions. M is the miner responsible for creating a block. Nonce is the solution to the cryptographic problem. MR is an abbreviation for the Merkle Root, which is a hierarchical arrangement of cryptographic hashes [34]. All the elements remain immutable except the nonce, which continuously changes as it attempts to solve a challenging puzzle. This puzzle involves finding a nonce that produces a hash lower than a certain target when it hashed with the block's header. We refer to this as proof of work (PoW). Other consensus mechanisms, such as proof of stake (PoS), might be used, where validators are chosen to create a new block on the basis of the number of coins they hold and are willing to stake [35].

4.2 PoWs

To ascertain a miner's ability to create blocks and concurrently append them to DAGchains, we need a proof of work (PoW) for every block. The hash function H (B) produces a sequence of 64 characters in the hexadecimal system. Miners must continually change their nonce until H (B) displays a specific pattern to demonstrate labor. The value of H (B) is required tobe less than d, where d represents the level of difficulty (number of prefix zeros) in obtaining a winner nonce. Before calculating the hash value, it is important to understand that a miner should package and handle all block components, such as the header, main data, and nonce. In PoW, as complexity increases, the consumption of computing resources and time also increases. Algorithm 1 illustrates the process of distributed mining for multiple miners.

4.3 Pointers

The pointers establish a link among blocks, allowing us to systematically discover all previous blocks associated with a certain block. This process expands the block's understanding to include all of its predecessors. Therefore, the current block (B_c) may be defined as confirming another previous block (B_p) if B_p is a predecessor of B_c, indicating a directed path from B_p to B_c. This is a mathematical formulation for verifying previous blocks.

4.4 DAGchains structure

DAGchains are directed and cyclic graphs, meaning that they have no loopback. Formally, it is a pair G = (B, E), where B is a finite set of valid blocks, and $E \subseteq B \times B$ is a set of directed edges. Each edge (B_p, B_c) $\in E$ indicates a directed connection from the previous block B_p to the current block B_c. We can say that the DAGchains are valid if all of their blocks are valid, i.e., satisfied. [36] and directed.

Fig. 4 depicts the steps for adding two blocks to the DAGchains. We can observe the existence of five distinct categories of blocks, each characterized by a special color. The red color represents the genesis block (GB), which is regarded as the initial component of the structure and serves as the foundation for the subsequent blocks. It lacks actual inputs, but it may contain the rules and information for the DAGchains, such as the block size, miner reward, level of difficulty, etc. The color green denotes blocks confirmed by at least two upcoming blocks, whereas the newly joined blocks (Tips) are indicated in yellow after verifying two previous blocks on the basis of the minimum consumption in the previous mining processes. The blue color designates blocks that have received confirmation from just one block. The purple color signifies that the blocks are currently engaged in proof of work and have not yet been placed within the structure. The following steps offer an entire clarification of the figure shown below.

- Fig. 4 (a) illustrates the process of miners adding 5 blocks (B_1–B_5) and directly connecting them to the genesis block because there are no existing earlier blocks. The blocks are highlighted in yellow as tips.
- The second step in Fig. 4 (b) involves appending five more blocks (B_6–B_{10}) so that each block references two of the preceding blocks on the basis of the priority established, which was previously mentioned. The blocks are confirmed and displayed in green, except for block B_5, which appears in blue owing to its connection to only block B_{10} and incomplete verification, leaving all transactions within this block pending.

- In Fig. 4 (c), four more blocks (B_{11}–B_{14}) arrive to confirm the remaining tips and pending blocks. B11 confirmed that the block awaits validation B5, changing its hue to green and confirming B_6.
- We include further blocks (B_{15}, B_{16}) via the same methodology to first emphasize the pending blocks and then the tip blocks, as shown in Fig. 4 (d) and (e).

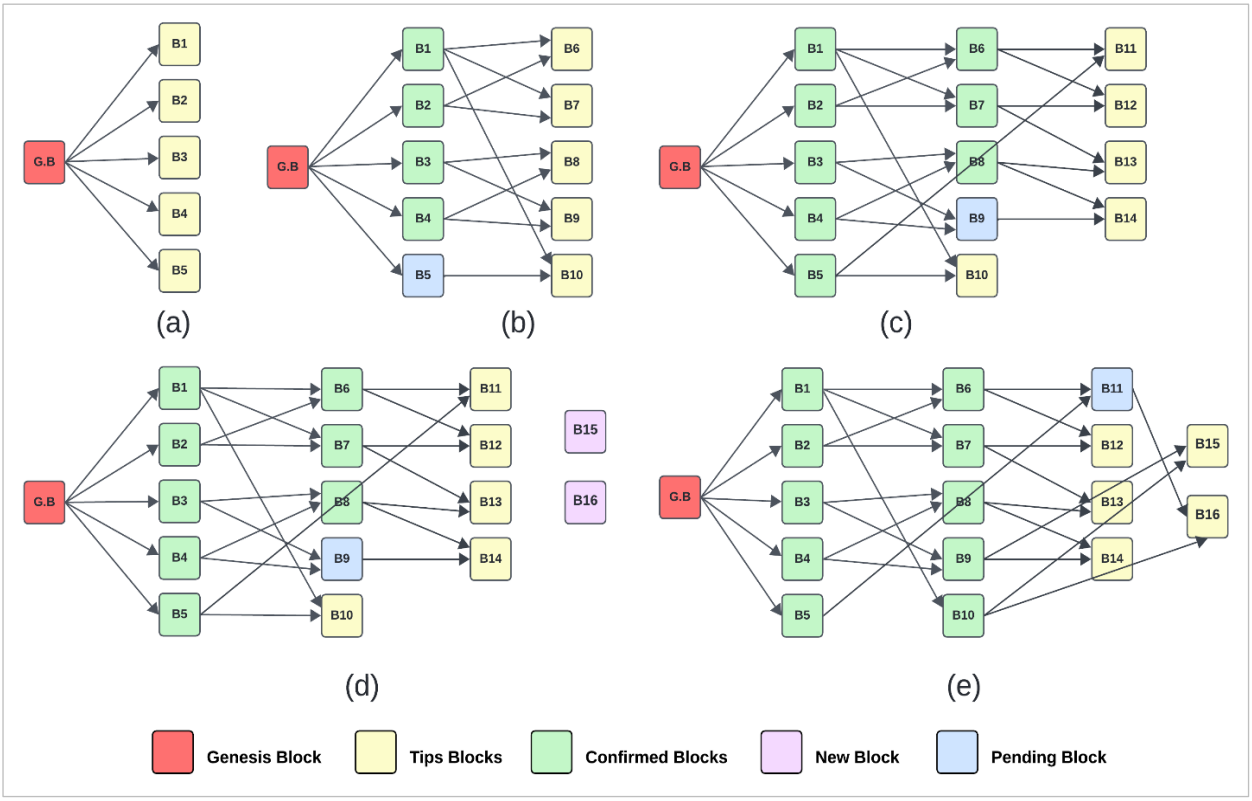

Fig. 4. DAGchains workflow mechanism

Algorithm 1. DAGchains mining process

Inputs:

Customers on a P2P network release a set of transactions (TXs).
A group of miners (Miner1, Miner2, ..., Miner N) are present.

Variables:

Mempools: A repository for storing incoming network transactions for each miner.
Block: place to store valid transactions after being checked by miners.
Nonce: a numerical value that ranges from 0 to 2^{256}.

Procedure:

Each miner within the group operates simultaneously:

1. Mempool [Miner]← store TXs.
2. Block←Select and store specific TXs from the
3. miner's Mempool based on the BAMTP strategy.
4. H(B)←SHA256(Block)
5. For Nonce = 0 to 2^256 do:
6. If not, PoW Satisfied H(B):
7. Nonce<- Nonce +1
8. H(B)<- SHA256(Block)
9. Else If Is_Block_Valid H(B)
10. Connectivity new block into DAGchains
11. Broadcast valid block

End for

4.5 Balanced Assignment of the Mempool Transactions Protocol (BAMTP)

Blockchain technology's memory pool, referred to as the mempool, serves as a temporary storage space for transactions awaiting validation, such as Bitcoin[37]. Once a user begins a transaction, it is immediately shared with the P2P network and included in the mempool of every full node as a miner. The mempool is a queue for these transactions, storing them until a miner or validator verifies and adds them to a block. Each full node in the network possesses a mempool, and different nodes may receive transactions at different times. Transactions in the mempool may exist in two distinct states: queued and pending. Full nodes verify queued transactions by checking the digital signatures, ensuring sufficient funds, adhering to network rules such as block size and target difficulty, and transforming them into pending transactions, meaning that they are valid for inclusion in a block. Only miners or validators can select pending transactions from the mempool on the basis of factors such as transaction fees.

The mempool plays a crucial role in maintaining the fair and orderly processing of transactions. The dimensions and composition of the mempool may substantially influence the efficiency and integrity of the blockchain network. For example, when the mempool reaches an excessive size, it might result in network congestion and an increase in transaction costs. On the other hand, an insufficiently sized mempool might not handle all the transactionsthe network receives. This may result in delays and significant financial losses for miners or validators. [38].

4.5.1 Methodology of BAMTP

Miners often select transactions with high fees to maximize their reward and obtain the highest income. Therefore, several miners are likely to process these transactions simultaneously. In the end, the miners add the transactions to the valid blocks according to their speed of verification and inclusion, disregarding the efforts of the other miners who struggled to obtain valid blocks owing to their inferior computational power. This naturally results in a significant waste of both computational power and time. Therefore, we propose a new strategy, the Balanced Assignment of Mempool Transactions Protocol (BAMTP), to prevent redundancy or multiple miners from processing the same transactions, which is based on the steps described below:

- Every miner collects all transactions from the distributed network and stores them as queued transactions in their private mempool.
- The coordinator, acting as a smart contract in the DAGchains network, uses a k-means algorithm to cluster transactions into three categories (high, medium, and low) depending on their fees for all miner's mempools. Each miner will implement the algorithm. Once the coordinator determines the three global seeds and the number of iterations (k) needed.
- The categorization of transaction fees results in the creation of three hash tables, each of which determines the ranking of the transactions and saves them in its respective hash table. To implement the hash table mechanism, a unique digital ID is necessary. We extract the numerical digits from the transaction identification and the hash ID and exclude any alphabetical characters to obtain the digit's unique ID.
- Picking the suitable transactions, the coordinator assigns a value ranging from 1 to N to each miner, where N represents the total number of miners on the network and the number of hops over transactions in the hash tables. The miners choose a transaction from each category (low, medium, and high) in their mempool, beginning with the N values given to them. The system continually updates the previous values by increasing them by N until it has selected all available transactions and switched them to the pending status.

For a better understanding of the protocol, we present the following example. Let us assume that there are three miner mempools, each containing 18 similar transactions. Initially, each miner employs the k-means algorithm to categorize the incoming transactions into three distinct groups (low, medium, and high) by algorithm 2. Next, the hash tables store the classified transactions, readying them for selection. Finally, since three miners are available in the network, the first miner chooses transactions with the first index and multiples of three (Tx_1,Tx_7,Tx_{13},Tx_4,Tx_{10},Tx_{16}). This process continues with the remaining miners. The second miner selects the transactions (Tx_2,Tx_8,Tx_5,Tx_{11},Tx_{13}), and the third selects the transactions (Tx_3,Tx_9,Tx_{15},Tx_6,Tx_{12},Tx_{18}). As a result, each miner will process a different set of transactions at roughly equivalent fees, avoiding duplication, investing each miner's computing power, and preventing wasted time, as shown in Fig. 5.

4.5.2 Mathematical model of the BAMTP

To present the Balanced Allocation of Mempool's Transaction Protocol (BAMTP) as a mathematical model, we can represent the steps and processes systematically via variables, functions, and algorithms. Below is a high-level description of the BAMTP formalized mathematically.

- **Transaction Representation**

Let $Tx = \{Tx_1, Tx_2, \ldots, Tx_m\}$ be the set of all the transactions in the network, where each transaction Ti is characterized by:

Transaction ID: ID (Tx_i),

Transaction fee: f (Tx_i), which is the fee associated with the transaction Tx_i,

Verification status: queued (before validation) or pending (after validation).

- **Mempool Definition:**

Each miner Mj (for j=1, 2…, N, where N is the number of miners) maintains a private mempool Mj, which is a collection of transactions that the miner has received and is in a queued state. The miner's mempool Mj can be represented as:

$$MP_j = \{Tx_i \mid Tx_i \in Queued\ by\ Miner\ M_j\} \quad (2)$$

- **Coordinator's role:**

The coordinator, which operates as a smart contract, is responsible for the global coordination of the transaction categorization process. The coordinator uses a k-means clustering algorithm to categorize the transactions into three fee categories: high, medium, and low. The categorization step can be mathematically described as follows:

Let f (Tx_i)f be the fee of a transaction Tx_i.

The coordinator computes the centroids C_1, C_2, C_3 corresponding to the high, medium, and low categories, respectively.

The transactions are grouped on the basis of their fees relative to the centroids:

$$C = \{C_1, C_2, C_3\}, where\ C_1 = \{Tx_i \mid f(Tx_i) \in High\}, C_2 = \{Tx_i \mid f(Tx_i) \in Medium\}, C_3 = \{Tx_i \mid f(Tx_i) \in Low\} \quad (3)$$

- **Transaction Ranking via Hash Tables:**

Once the transactions are categorized, they are ranked within each category on the basis of their transaction $ID(Tx_i)$A hash table H_K for each fee category k∈ {1,2,3} is created:

$$H_K = \{Tx_i \mid f(Tx_i) \in C_k\}, for\ k \in \{1,2,3\} \quad (4)$$

Each transaction Txi in Hk is assigned a unique digit ID derived from its transaction $ID(Tx_i)$ by extracting numerical digits:

$$ID(Tx_i) \rightarrow Digit\ ID(Tx_i) \quad (5)$$

This ensures that each transaction can be uniquely identified within each hash table.

- **Transaction Allocation:**

To prevent redundancy and ensure fair processing, the coordinator assigns miners to select transactions from the categorized hash tables. Let N be the total number of miners in the network. The coordinator assigns each miner Mj a unique index j in the range [1, N].

Each miner selects one transaction from each of the three categories C1, C2, and C3 as follows:

Miner Mj starts by selecting the $j-th$ transaction in each category.

$$Tx_{i_j}^{(k)} \in H_k, for\ K \in \{1,2,3\} \quad (6)$$

where $Tx_{i_j}^{(k)}$ is the index corresponding to the j-th transaction in category Hk.

The selection is updated in subsequent rounds by incrementing the transaction index for each category by N, ensuring that no two miners select the same transaction.

The selection of transactions by miner Mj from the three categories over n rounds is given by:

$$M_j\ selects\ Tx_{i_j}^{(k)}, where\ i_j^{(k)} = (j-th) + N \quad (7)$$

- **Pending Transaction Update:**

Once miners have selected their transactions, the transactions are marked as pending: for each $Tx_i \in M_j$, if Tx_i is selected by miner Mj, then Tx_i is moved to the pending state. This transition occurs once the miner successfully adds the transaction to the block and is validated.

Algorithm 2. Reward strategy

Inputs:

The private mempool of each miner.

Outputs:

Miner's transactions share and total fees.

Procedure:

Step 1: The K-means algorithm clusters TXs into three specific sets based on fees, namely, Low- priority cluster: transactions with fees less than 1$
Medium-priority cluster: transactions with moderate fees, ranging from $1 to $5
High-priority cluster: transactions with fees exceeding $5
Step 2: Generate a unique digital ID for each cluster TX.
Step 3: Create three tables for indexing categorized TXs using the hash table mechanism.
Low-hash table: a collection of indexed TXs that have low fees.
Medium-hash table: a collection of indexed TXs that have medium fees.
High-hash table: a collection of indexed TXs that have high fees.
Step 4: For each miner, select and verify certain TXs and place them in the temporary store as pending transactions to prepare a valid block. The miner then receives his rewards.

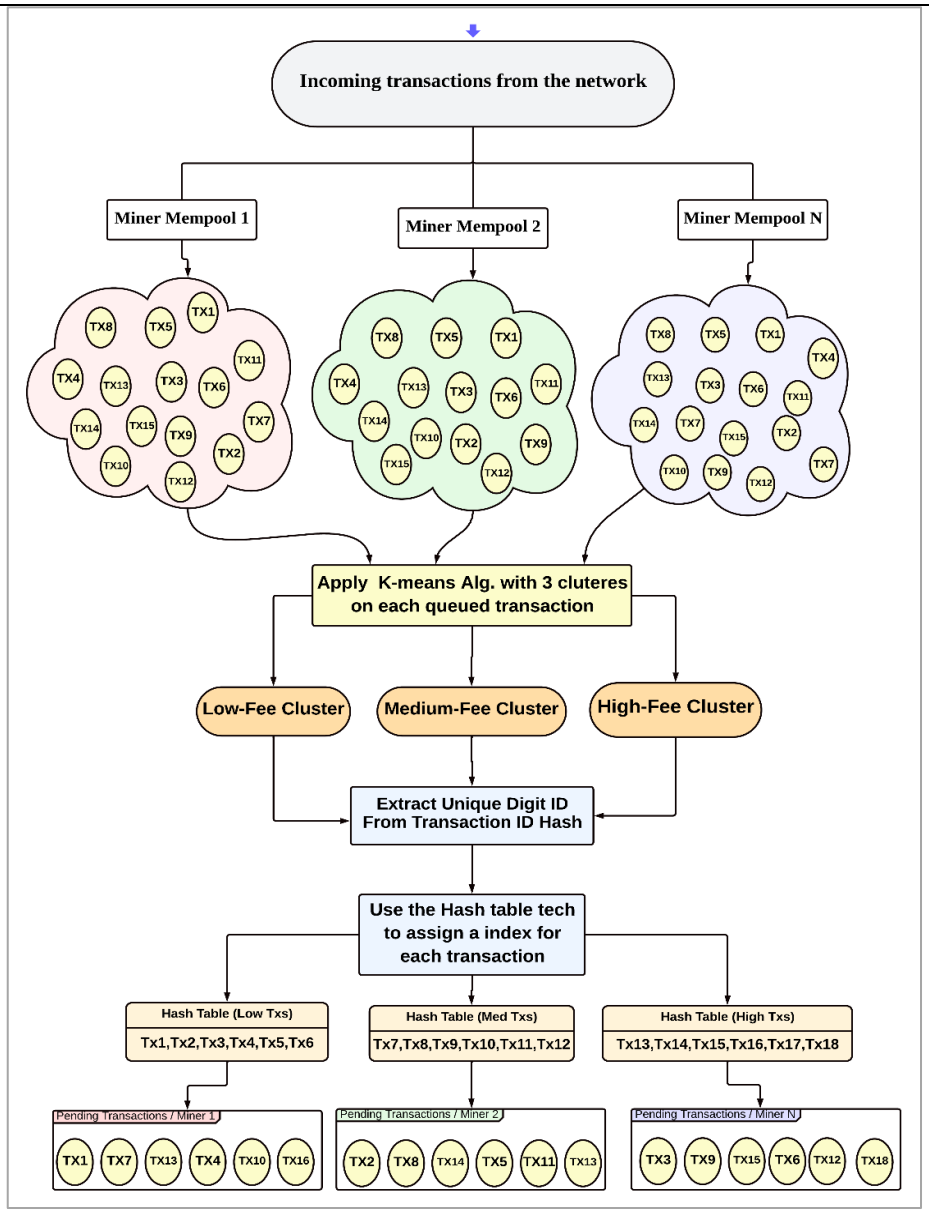

Fig. 5. Chart depicting the sequence of actions in the BAMTP

4.5.2 Impact of the BATMP on the CBR Protocol

Blockchain networks such as Bitcoin use the Compact Block Relay CBR mechanism to reduce the bandwidth required for propagating newly mined blocks. Instead of transmitting the entire block, only a compact representation is sent, which includes a list of short 6-byte transaction identifiers for each transaction to the receiving node, reducing the amount of data sent. This allows nodes to reconstruct the full block via transactions they already have in their mempool, preventing data duplication and enhancing network efficiency. Compact block relay improves the speed and efficiency of block propagation while maintaining the integrity and security of the blockchain.

For a single block containing up to 2,000–3,000 transactions (common in Bitcoin), sending transaction IDs typically takes only a few milliseconds to a few hundred milliseconds over modern internet connections. Validation and full propagation of the compact block across the Bitcoin network generally occur within 1–2 seconds and do not exceed 10 seconds in congested networks and high latency.

The BAMTP significantly improves the performance of the compact block relay (CBR) protocol in peer-to-peer (P2P) networks by facilitating more efficient transfer of compact blocks. BAMTP uses coordinated transaction identifiers (CTIDs) that were prepared among miners, eliminating the need for the protocol (CBR) to propagate all transaction identifiers and parsimony to send full transactions when recipient nodes request them from miners or neighbor nodes only. This improvement allows verifiers to reconstruct blocks directly from their local mempool by selecting presorted transactions from three hash tables categorized on the basis of CTIDs. As a result, this approach reduces bandwidth consumption, speeds up block transfers, and supports increased transaction throughput, improving the overall performance of the blockchain network.

5. SYSTEM IMPLEMENTATION

At target difficulty 4, both the traditional and the proposed systems were implemented via the Python programming language and Mininet network simulation on a personal computer equipped with an Intel (R) CoreTM i5-8300H CPU with eight cores. Fig. 6 depicts the construction of a network with nine interconnected nodes. Node 1 creates transactions with a size of 1 MB (3000 TXs per block) for each node and transmits them via socket services to the 9 nodes that act as miners to form valid blocks by finding the winning nonces. Node 0 then receives the blocks for validation and adds them to the blockchain\DAGchains.

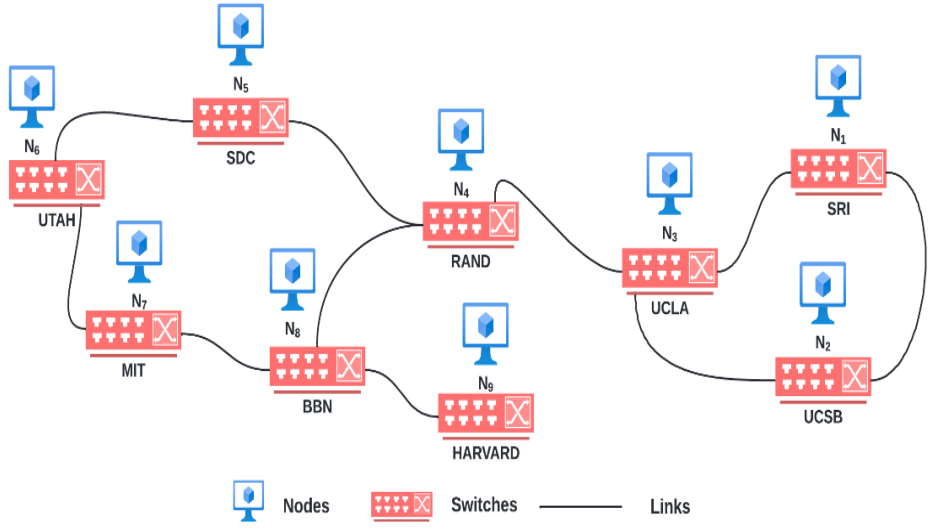

Fig. 6. P2P Mininet network with nine nodes and switches

5.1 Conventional PoW Approach

The classic blockchain generates 6,000 addresses, with 3,000 senders and 3,000 recipients. Each sender address is associated with a single transaction with a size not exceeding 400 KB. To pay a certain amount to the recipient address, this transaction includes the sender address, recipient address, timestamp, sent amount, etc. The senders sign these transactions with their private keys before sending them to the P2P network. The miners receive, validate, and collect transactions into their blocks. The miner who demonstrates the fastest computational power of the PoW algorithm in extracting the winner receives permission to append transactions as a valid block to the blockchain, thereby earning profits.

TABLE III. RESULTS OF THE TRADITIONAL POW MINING BY MININET NETWORK SIMULATION

No. Block	Miners	No. TXs per Block	Winner Nonce	Consumed Time(sec)
1	M_5	3000 TXs	48989	870
2	M_1		9093	165
3	M_1		24235	430
4	M_7		73848	1368
5	M_3		18172	325
6	M_5		19547	347
7	M_2		49862	888
8	M_8		20210	379
Total consumed time				4772

In Table III, eight miners compete to build every block in the blockchain, attempting to identify the winner nonce as rapidly as possible, depending on their computational power to append a single block to the blockchain. For example, the fifth miner created the first block in 870 seconds, whereas the same miner took 347 seconds to produce the sixth block. The table shows the winning number and the mining time consumed for processed blocks. Importantly, the mining time is directly proportional to the computational power and the number of wins. This is seen in Block 4, which was formed by the seventh miner after a significant amount of time compared with the rest of the blocks. This delay was caused by the length of the winning number (73848), given that the miners' processing ability is not highly efficient in our device. The mining process's cumulative time to add the eight blocks was approximately 4700 s, including a total of 24,000 transactions and a throughput rate of 5 transactions per second (TPS), which can be calculated via Eq. (2) [39].

$$TPS = \frac{Number\ of\ Transactions}{Total\ Time\ Spent} \tag{2}$$

Additionally, it is important to note that in real-world PoW networks, mining times can vary significantly owing to adjustments in network difficulty and fluctuations in available hashing power. This variability further complicates the scalability of PoW-based blockchains, especially as transaction volumes increase, highlighting the need for innovative structures such as DAGchains.

5.2 The Proposed Approach

On the other hand, the proposed system has been implemented on the same network to incorporate the other 8 blocks by miners labelled M_1, M_2, ..., and M_8 into the DAGchains, utilizing the distribution performance of the PoW algorithm. Each block contains approximately 3,000 distinct transactions and is completely independent of the contents of the other blocks selected on the basis of the previously described BAMTP in Section 4.

TABLE IV. THE MINER'S POWS RESULTS BY DISTRIBUTION WAY

No. Block	Miners	No. TXs per block	winner nonce	Consumed Time (sec)
1	M_2	3000 TXs per block	359	11
2	M_4		5533	110
3	M_3		15156	269
4	M_6		23401	389
5	M_7		35613	517
6	M_3		56991	806
7	M_5		60796	839
8	M_1		76858	978
The overall duration of the parallel mining process				978

Table IV presents the results from the DAGchains system using distributed proof-of-work (PoW) mining. Unlike traditional PoW, where miners compete sequentially, the distributed approach allows multiple miners to work in parallel, each processing a block concurrently. This significantly reduces the overall mining time and increases the transaction throughput. For example, miner M_2 could process 3,000 transactions in just 11 seconds for the first block, whereas other miners, such as M_1, took longer (978 s) owing to the increased complexity of the winner nonce (76858). Despite some variation, the parallel

approach allows for the cumulative processing of 24,000 transactions across eight blocks within a total mining time of 978 seconds. This results in an improved transaction throughput of approximately 24 TPS—four times higher than that of traditional systems. The table illustrates the efficiency of the proposed DAGP system, which leverages parallelism to improve scalability and minimize latency. The distributed structure not only optimizes processing speed but also reduces the issue of delays due to nonce complexity, as miners can process different transactions simultaneously. The DAGP system thus represents a significant advancement over traditional PoW, allowing blockchain networks to handle higher transaction volumes without compromising decentralization or security.

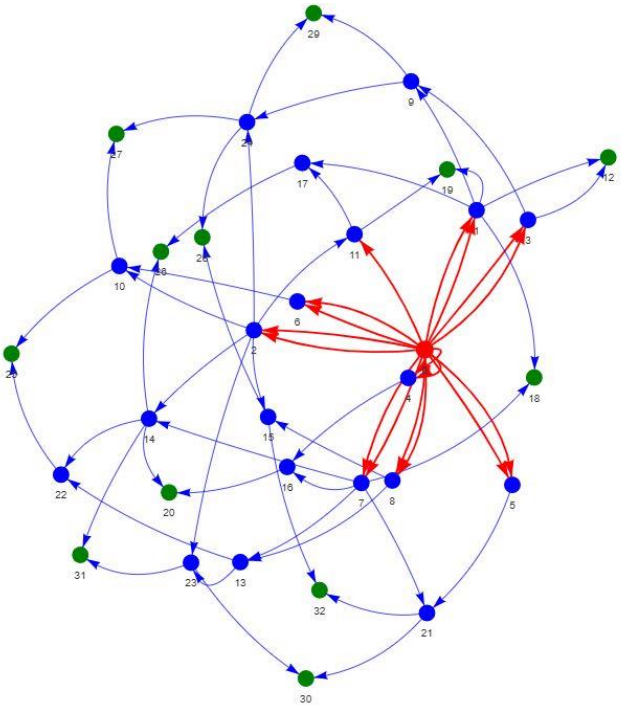

Fig. 7. DAGchains with 40 added blocks

Figure 7 illustrates the incorporation of 40 blocks into the DAGchains at an increment of 8 blocks for each instance, including a total of 120,000 unique transactions. This procedure depends on the presence of 8 miners in the network. The green block as the tip block refers to the red block, also known as the genesis block, through the previous two hash values that came out of the same block as the origin of the DAG chain structure. For example, block 15 linked to the blue-labelled hash values of internal blocks 2 and 8, whereas block 16 referenced both blocks 4 and 7, and so on, on, on the basis of the reference to the two previous blocks that have the least mining time.

TABLE V. ANAN ANALYSIS OF DIFFERENCES BETWEEN BLOCKCHAIN AND DAGCHAINS

Aspect	Traditional PoW (Table III)	Distributed PoW in DAGchains (Table IV)
Mining Approach	Sequential, with miners competing to append a single block	Parallel, with multiple miners working simultaneously on different blocks
Total Mining Time	4,772 seconds for eight blocks	978 seconds for eight blocks
Transaction Throughput	~5 TPS	~24 TPS
Mining Variability	Significant variability due to nonce complexity and miner power	Reduced variability; independent block mining mitigates nonce complexity delays
Scalability	Limited bottlenecks as miners wait for each block to be mined	High; concurrent block creation enhances scalability
Resource Utilization	High resource wastage due to redundant mining efforts	Optimized: miners work on unique transactions, reducing redundancy

The comparison in Table V highlights the clear advantages of the DAGchains system over traditional PoW mining. The distributed, parallel mining approach in DAGchains significantly reduces the mining time (from 4,772 to 978 s), with a difference of 3494 s, and quadruples the transaction throughput, enabling miners to process different blocks simultaneously. Unlike traditional PoW, where sequential mining and nonce complexity create bottlenecks and result in resource waste,

DAGchains ensure optimized resource use by assigning unique transactions to each miner. This approach improves scalability and minimizes mining delays, making DAGchains a more efficient and scalable solution for high-transaction environments.

TABLE VI. POW ALGORITHM TIME RESULTS (SEC) FOR EIGHT CONSECUTIVE EXPERIMENTS USING STANDARD AND PROPOSED BLOCKCHAIN METHODOLOGIES.

	Traditional PoW	**DAGchains PoW**
Exp 1	2840	699
Exp 2	4048	1858
Exp 3	6216	2036
Exp 4	2891	978
Exp 5	6438	2537
Exp 6	4772	1878
Exp 7	3907	1060
Exp 8	4783	2016

Table VI presents the time taken (in seconds) for eight consecutive mining experiments using both traditional PoW and DAGchains PoW. Across all the experiments, the DAGchains PoW consistently achieves significantly lower mining times than the traditional PoW does. For example, in Experiment 1, DAGchains PoW completed the process in 699 seconds, whereas traditional PoW took 2,840 seconds—a reduction of nearly 75%. All the experiments observe this pattern of time savings, with DAGchains PoW demonstrating faster results even as complexity increases. The data highlight the effectiveness of the parallel mining structure of the DAGchains, which decreases the reliance on sequential mining and enables the simultaneous processing of multiple transactions. By minimizing time requirements across various test conditions, DAGchains PoW substantially outperforms traditional PoW, making it a more suitable option for environments with high transaction throughput and reduced latency.

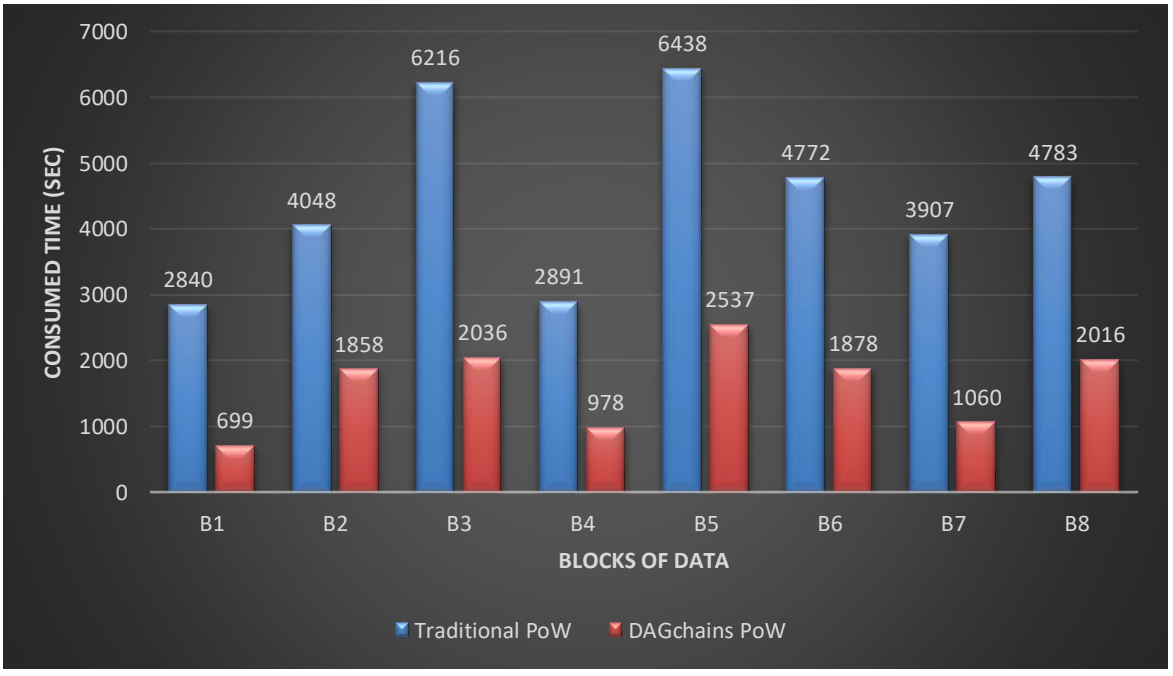

Fig. 8. A chart displays the results of eight tests conducted on traditional PoW- and DAG-chain PoW approaches.

5.3 Rewards strategy

The reward system was devised for miners to receive equitable and proximate compensation for their block creation efforts. After categorizing transactions into three distinct categories on the basis of their fees (low, medium, and high), miners meticulously select transactions from their hash tables, adhering to the abovementioned guidelines to prevent transaction collisions or double-spending. The conventional approach typically prioritizes high-value transactions, leading to a delay in

processing lower-value transactions, also called microtransactions.[40]The proposed reward system distinguishes itself by effectively distributing categorized transactions to miners operating concurrently within the network. This approach reduces the computational effort and time required in the mining process and enhances scalability.

We demonstrate the reward system's workflow and assume that eight miners are in the network, engaging in the mining process to create 8 blocks. After mutual verification, these blocks are added to the miner's respective copies in the DAGP structure. A total of 10,000 transactions were generated and sent, containing the essential elements (sender address, receiver address, fees) for all miners to independently classify and systematically select transactions from their mempools according to the BAMTP mechanism, followed by the phase of mining and valid block creation. Tables II and 3 display the rewards system's outcomes for an eight-cohort of miners.

TABLE VII. OUTCOMES OF NUMBER-CATEGORIZED TRANSACTIONS FOR EIGHT MINERS BASED ON BAMTP

Miners	No. Low fee transactions	No. Medium fee transactions	No. High fee transactions	Total of classified transactions
M_1	417	424	410	1251
M_2	417	424	410	1251
M_3	417	424	410	1251
M_4	417	424	410	1251
M_5	417	424	409	1250
M_6	416	424	409	1249
M_7	416	424	409	1249
M_8	416	423	409	1248

Table VII reveals a balanced distribution of transactions among eight miners, categorized into low-, medium-, and high-fee groups. Each miner processes a similar number of transactions, with a total close to 1,251, showing that the allocation mechanism—likely BAMTP—effectively ensures workload balance. Minor variations exist: while most miners handle 1,251 transactions, M_5 has 1,250, M_6 and M_7 have 1,249, and M_8 has 1,248. Across categories, each miner processes approximately 417 low-fee, 424 medium-fee, and 410 high-fee transactions, with minor differences indicating consistent allocation. These slight variances indicate that the BAMTP maintains equitable transaction distribution among miners, fostering fairness and reducing the need for centralized mining pools, which helps enhance scalability and miner engagement.

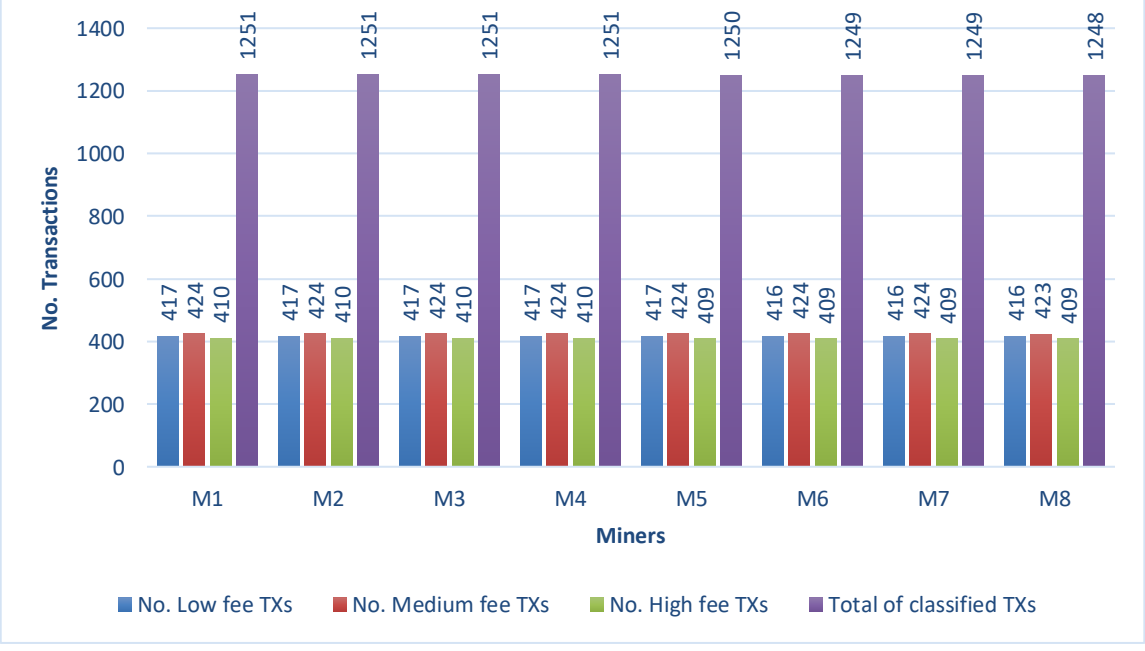

Fig. 9. The chart illustrates the number of categorized transactions for each miner based on the BAMTP.

TABLE VIII. RESULTS OF TOTAL FEE TRANSACTIONS FOR EIGHT MINERS BASED ON BAMTP

Miners	Sum. Low fee transactions	Sum. Medium fee transactions	Sum. High fee transactions	Total sum of classified transactions
M_1	3,603	10427	17,175	31,206
M_2	3,340	10685	17,048	31,074
M_3	3,562	10583	17,091	31,237
M_4	3,342	10629	17,167	31,139
M_5	3,562	10911	17,070	31,544
M_6	3,683	10661	16,987	31,332
M_7	3,692	10677	17,053	31,423
M_8	3,676	10677	16,988	31,341

Table VIII illustrates the total transaction fees earned by each miner across low-, medium-, and high-fee categories, showing how BAMTP effectively ensures a fair distribution. The total fees for miners generally fall between 31,000 and 31,500 coins, with M_5 at the top with 31,544 coins and M_2 at the bottom with 31,074 coins, indicating minimal disparity in overall earnings. In the low-fee category, miners' earnings range from 3,340 coins (M_2) to 3,692 coins (M_7), reflecting some variability but remaining within a close range. Medium-fee transactions show particularly consistent earnings, with totals spanning from 10,427 coins (M_1) to 10,911 coins (M_5), highlighting BAMTP's success in ensuring uniform distribution within this category. High-fee transaction earnings vary slightly from 16,987 coins (M_6) to 17,175 coins (M_1), demonstrating an even spread without any miner disproportionately benefiting from high-fee transactions. Overall, the close alignment of total fees across all categories confirms the BAMTP's effectiveness in creating an equitable system that reduces competition for high-fee transactions and distributes earnings fairly among miners. This balanced reward structure promotes miner participation and supports a decentralized, sustainable network by minimizing fee-based disparities. Fig. 5 illustrates the miners' workflow based on the BAMTP mechanism for transactions from the peer-to-peer network. This process begins when users release the transactions and ends when the miner stores them as pending transactions in temporary blocks.

Fig. 10. Chart of total fee transactions for eight miners based on the BAMTP

5.4 Analysing the Impact of the BAMTP on Optimizing CBR Performance

The table presents a comparative analysis of the CBR and BATMP protocols, focusing on the time required for transaction identifiers to propagate by miners through the network and accelerate validators' verification of mining processes. The experiment involved nine blocks, each containing varying numbers of transactions. The results demonstrate that the time required by the CBR protocol to transmit transactions fluctuates significantly and is influenced by factors such as the number of identifiers, bus speeds, and network congestion, which can either accelerate or slow down data transfer. In contrast, the proposed BATMP protocol results in a consistently low and stable time for transaction propagation, highlighting its efficiency and effectiveness in the block verification process.

TABLE IX. RESULTS OF SIMULATION CBR AND BATMP

No. transactions	Time sent of CBR (ms)	Time sent of BATMP (ms)
5360 Tx	365.36	0.03
2803 Tx	191.03	0.05
4298 Tx	292.66	0.02
1648 Tx	112.98	0.01
6142 Tx	418.75	0.05
4947 Tx	337.31	0.01
7217 Tx	491.55	0.03
3546 Tx	241.96	0.02

The time consumption for the CBR protocol shows a wide range of variability across the different blocks, influenced by factors such as the number of transaction identifiers, network bus speeds, and levels of congestion. For example, at 1648 transactions (Tx), the CBR time is the shortest, at 112.98 ms. Conversely, at 7217 Tx, the time peaks at 491.55 ms, reflecting an almost fivefold increase. Other values fluctuate significantly, such as 418.75 ms at 6142 Tx and 191.03 ms at 2803 Tx. This fluctuation indicates that the CBR protocol lacks adaptability to dynamic network conditions. As network congestion or transaction volume increases, the protocol's performance degrades, leading to delays in data transfer.

In contrast, the BATMP protocol consistently demonstrates minimal and stable performance across all blocks, with propagation times ranging from 0.01 ms to 0.05 ms, regardless of the transaction volume. For example, at 1648 Tx, the BATMP time is only 0.01 ms, whereas at 7217 Tx, it remains negligible at 0.03 ms. Across all other blocks, the time remains similarly low, with no significant deviation, highlighting the protocol's robustness and efficiency.

This consistency confirmed BATMP’s ability to maintain optimal performance even under varying network conditions. Unlike CBR, BATMP is unaffected by the number of transaction identifiers, suggesting the use of advanced optimization techniques that minimize delay and ensure rapid transaction propagation.

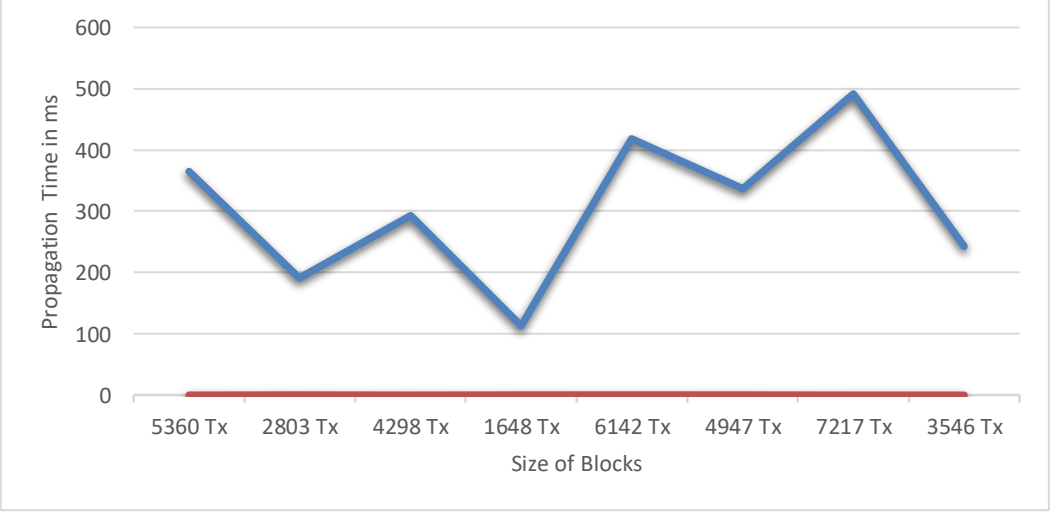

Fig. 11. Consumption time of CBR vs BATMP

6. CONCLUSION

In this paper, we propose DAGchains, an improved distributed ledger technology that integrates a DAG with a blockchain protocol to address the scalability issue that plagues existing blockchain systems. In contrast to the conventional blockchain, which appends a single block to the chain for each mining process, DAGchains introduce a more scalable model and efficiency that leverages the parallel mechanism of PoW among miners to produce multiple valid blocks and append them to the structure simultaneously. Additionally, the proposed approach presents a novel method (BAMTP) for reliably allocating and distributing transactions effectively among miners, thereby preventing repetition, minimizing computational resource waste, discouraging reliance on a large mining pool, ensuring the prompt execution of microtransactions, and maintaining the equitable distribution of all transaction fees. The experimental results showed that the DAGchains improve efficiency 4x more than the PoW conventional system does, offering a promising resolution to the bottleneck issue and representing a significant step towards scalable and sustainable blockchain solutions.

Conflicts of interest

None

Funding

None

Acknowledgement

The authors are thankful to the computer science department, computer science and mathematics faculty, and the University of Kufa for allowing us to work on servers and labs and creating a conducive research environment.

References

[1] Y. Li *et al.*, "Direct acyclic graph-based ledger for internet of things: Performance and security analysis," *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1643-1656, 2020.

[2] A. W. Reza, K. Islam, S. Muntaha, O. B. Abdur Rahman, R. Islam, and M. S. Arefin, "Education Certification and Verified Documents Sharing System by Blockchain," *International Journal of Intelligent Engineering & Systems*, vol. 15, no. 6, 2022.

[3] P. D. Hakim and V. M. Vaze, "Blockchain for Secure Medical Records Storage and Medical Service Framework using SHA 256–Verifiable Key," *Int J Intell Eng Syst*, vol. 14, no. 6, pp. 1-9, 2021.

[4] P. Ramya, T. Jashwanth, and D. V. Sathvik, "A Hybrid Proof of Stake-Trust Block Chain Model in Pervasive Social Networking for E-voting System," *International Journal of Intelligent Engineering & Systems, INASS*, vol. 15, no. 4, pp. 1-10, 2022.

[5] N. Sadeq, Z. Hamzeh, G. Nassreddine, and T. ElHassan, "The impact of Blockchain technique on trustworthy healthcare sector," *Mesopotamian Journal of CyberSecurity*, vol. 2023, pp. 104-114, 2023.

[6] M. Al-Zubaidie and W. Jebbar, "Transaction security and management of blockchain-based smart contracts in e-banking-employing microsegmentation and yellow saddle Goatfish," *Mesopotamian Journal of CyberSecurity*, vol. 4, no. 2, pp. 1-19, 2024.

[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Social Science Research Network (SSRN)*, pp. 1-9, August 21, 2008.

[8] A. Sohofi, A. Amidian, and Y. Farjami, "SocioChain: A Distributed Ledger-based Social Framework for the Internet of Things," *International Journal of Intelligent Engineering & Systems*, vol. 14, no. 5, 2021.

[9] I. S. Rao, M. M. Kiah, M. M. Hameed, and Z. A. Memon, "Scalability of blockchain: a comprehensive review and future research direction," *Cluster Computing*, pp. 1-24, 2024.

[10] J. He, G. Wang, G. Zhang, and J. Zhang, "Consensus mechanism design based on structured directed acyclic graphs," *Blockchain: Research and Applications*, vol. 2, no. 1, p. 100011, 2021.

[11] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 17-30.

[12] D. Yang, C. Long, H. Xu, and S. Peng, "A review on scalability of blockchain," in *Proceedings of the 2020 2nd International Conference on Blockchain Technology*, 2020, pp. 1-6.

[13] M. Du, Q. Chen, and X. Ma, "MBFT: A new consensus algorithm for consortium blockchain," *IEEE Access*, vol. 8, pp. 87665-87675, 2020.

[14] J. Kwon and E. Buchman, "Cosmos whitepaper," *A Netw. Distrib. Ledgers*, vol. 27, pp. 1-32, 2019.

[15] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," 2016.

[16] C. Xu, C. Zhang, J. Xu, and J. Pei, "SlimChain: Scaling blockchain transactions through off-chain storage and parallel processing," *Proceedings of the VLDB Endowment*, vol. 14, no. 11, pp. 2314-2326, 2021.

[17] S. Shahriar Hazari and Q. H. Mahmoud, "Improving transaction speed and scalability of blockchain systems via parallel proof of work," *Future internet*, vol. 12, no. 8, p. 125, 2020.

[18] N. Sohrabi and Z. Tari, "ZyConChain: A scalable blockchain for general applications," *IEEE Access*, vol. 8, pp. 158893-158910, 2020.
[19] A. Tokhmetov, V. Lee, and L. Tanchenko, "DEVELOPMENT OF DAG BLOCKCHAIN MODEL," *Scientific Journal of Astana IT University*, 2023.
[20] P. Chorey and N. Sahu, "Enhancing efficiency and scalability in Blockchain Consensus algorithms: The role of Checkpoint approach," *Journal of Integrated Science and Technology*, vol. 12, no. 1, pp. 706-706, 2024.
[21] S. Popov and Q. Lu, "IOTA: feeless and free," *IEEE Blockchain Technical Briefs*, 2019.
[22] J. L. Romero Ugarte, "Distributed ledger technology (DLT): introduction," *Banco de Espana Article*, vol. 19, p. 18, 2018.
[23] M. Gorbunova, P. Masek, M. Komarov, and A. Ometov, "Distributed ledger technology: State-of-the-art and current challenges," *Computer Science and Information Systems*, vol. 19, no. 1, pp. 65-85, 2022.
[24] Y. Zhou *et al.*, "Application of distributed ledger technology in distribution networks," *Proceedings of the IEEE*, vol. 110, no. 12, pp. 1963-1975, 2022.
[25] S. Barj, A. Ouaddah, and A. Mezrioui, "Cryptography in distributed ledger technologies from a layered perspective: A state of the art," in *International Conference on Digital Technologies and Applications*, 2023, pp. 210-220: Springer.
[26] C. Pop, T. Cioara, I. Anghel, M. Antal, and I. Salomie, "Blockchain based decentralized applications: Technology review and development guidelines," *arXiv preprint arXiv:2003.07131*, 2020.
[27] M. Habiban, F. R. Hamade, and N. A. Mohsin, "Hybrid Edge Detection Methods in Image Steganography for High Embedding Capacity," *Cybernetics and Information Technologies*, vol. 24, no. 1, 2024.
[28] M. J. M. Chowdhury *et al.*, "A comparative analysis of distributed ledger technology platforms," *IEEE Access*, vol. 7, pp. 167930-167943, 2019.
[29] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*, 2017, pp. 557-564: Ieee.
[30] F. Masood and A. R. Faridi, "An overview of distributed ledger technology and its applications," *Int. J. Comput. Sci. Eng*, vol. 6, no. 10, pp. 422-427, 2018.
[31] S. Gaba *et al.*, "Holochain: An agent-centric distributed hash table security in smart IoT applications," *IEEE Access*, 2023.
[32] N. El Ioini and C. Pahl, "A review of distributed ledger technologies," in *On the Move to Meaningful Internet Systems. OTM 2018 Conferences: Confederated International Conferences: CoopIS, C&TC, and ODBASE 2018, Valletta, Malta, October 22-26, 2018, Proceedings, Part II*, 2018, pp. 277-288: Springer.
[33] A. Haque and M. Rahman, "Blockchain technology: Methodology, application and security issues," *arXiv preprint arXiv:2012.13366*, 2020.
[34] H. Liu, X. Luo, H. Liu, and X. Xia, "Merkle tree: A fundamental component of blockchains," in *2021 International Conference on Electronic Information Engineering and Computer Science (EIECS)*, 2021, pp. 556-561: IEEE.
[35] H. Kim and D. Kim, "A taxonomic hierarchy of blockchain consensus algorithms: an evolutionary phylogeny approach," *Sensors*, vol. 23, no. 5, p. 2739, 2023.
[36] A. F. Mahdi and F. Rabee, "A blockchain mining proof of work approach based on fog computing virtualization for mobile crowdsensing," in *2024 Third International Conference on Distributed Computing and High Performance Computing (DCHPC)*, 2024, pp. 1-9: IEEE.
[37] M. S. K.K. (2019). *Mempool - bitcoin explorer*. Available: https://mempool.space/
[38] A. Mikhaylov, H. Dinçer, S. Yüksel, G. Pinter, and Z. A. Shaikh, "Bitcoin mempool growth and trading volumes: Integrated approach based on QROF multi-SWARA and aggregation operators," *Journal of Innovation & Knowledge*, vol. 8, no. 3, p. 100378, 2023.
[39] S. Banupriya and P. Sharmila, "An Optimization of Blockchain Parameters for Improving Consensus and Security in eHealthChain," *International Journal of Intelligent Engineering & Systems*, vol. 17, no. 2, 2024.
[40] U. Anıl and M. A. Akcayol, "Deep learning based prediction model for the next purchase," *Advances in Electrical and Computer Engineering*, vol. 20, no. 2, 2020.
[41] O. Aljumaiah, W. Jiang, S. R. Addula, and M. A. Almaiah, "Analyzing cybersecurity risks and threats in IT infrastructure based on NIST framework," *Journal of Cyber Security and Risk Auditing*, vol. 2025, no. 2, pp. 12–26, 2025.