Blockchain Segmentation: An Industrial Solution for Large Scale Data

Anooja Ali[1](✉), Nisha Joseph[2], and TousifAhamed Allabksha Nadaf[3]

[1] School of CSE, REVA University, Bengaluru 560064, India
anooja.ali@reva.edu.in
[2] Department of CSE, Saintgits College of Engineering, Pathamuttam, Kottayam 686532, India
nisha.joseph@saintgits.org
[3] Architect, Wipro Arabia Ltd., Dhahran, Saudi Arabia
ahamedpapers@gmail.com

Abstract. Blockchain technology finds diverse applications, encompassing the security and managing the vast amounts of data generated by IoT devices, with secure and transparent communication between them. In Supply chain management, blockchain can track products and goods from their origin to destination. A blockchain is an autonomous digital ledger employing Distributed Ledger Technology (DLT) to securely and transparently record transactions in a decentralized manner. It accomplishes this by using a network of computers (nodes) and these components collaborate to verify and register transactions within a communal database. This research proposes a new approach called Segment Blockchain that divides a blockchain into smaller segments and enables nodes to retain only one segment in place of the entire blockchain. This approach can potentially reduce the storage requirements for participating nodes, facilitate the incorporation of addional nodes into the network and maintain a copy of the blockchain. Our proposed methodology aims to address the concern of the risk of a singular vulnerable point, wherein a malicious entity keeps all copies of a specific segment and leaves the system, causing in the irreversible deletion of that segment. The proposed blockchain system can handle big data while reducing storage space demands while ensuring heightened security for user data. Theoretical evidence shows that this is achieved by limiting the number of blocks a malicious entity has the capability to both store and distribute every segment over a cluster of cloud-based blocks, the storage burden is significantly reduced compared to conventional designs. The system was successful in reducing storage space by 33% for large scale data. This makes the proposed segmentation approach more practical for processing and managing large volumes of data.

Keywords: Blockchain · Decentralization · Distributed Ledger Technology · Network · Segmentation · Shared Databases

1 Introduction

Blockchain systems are renowned for its anonymity and independence, hence, it is essential to ensure that each transaction can be authenticated to build trust and ensure security. Therefore, it is crucial to keep a full and accurate record of all transactions.

K. K. Patel et al. (Eds.): icSoftComp 2023, CCIS 2030, pp. 180–192, 2024.
https://doi.org/10.1007/978-3-031-53731-8_15

However, as the blockchain size has increased over time, the cost of storing all the blocks of the mainchain has also risen sharply, making it more expensive to sustain a complete node within the Bitcoin network. Blockchain can be used to securely verify and authenticate identities, which can help prevent fraud and protect sensitive information [1]. Blockchain-based finance allows decentralized financial transactions and services. It is essential to have a secure and decentralized data sharing and collaboration while ensuring data privacy and ownership.

When the size of the blockchain grows, computational requirements and storage regarding the preservation of a full node also increase, making it more difficult and expensive for users to engage in the network. This potentially causes centralization of the network as only large and well-funded players can afford to maintain a full node. In order to tackle this, various proposals have been put forward to reduce the storage and computational requirements of maintaining a full node while still ensuring the security and trustworthiness of the blockchain. Even a small transaction in Bitcoin or other Distributed Ledgers occupies only hundred bytes, but nodes with limited resources are gradually leaving the mining game, leading to more devices operating in lightweight mode or participating in mining pools.

The common approaches to increase the blockchain performance include weighted models [2], pruning, blockchain sharding [3] and off-chain [4]. Pruning removes the old and unnecessary data from the blockchain without compromising its integrity. Sharding refers to the process of dividing the blockchain into smaller, more manageable segments, allowing different nodes to process them in parallel. Additionally, there are also efforts to develop off-chain solutions, such as the Lightning Network, which enable faster and cheaper transactions by conducting them off the main blockchain.

Decentralization becomes a major issue for large data. Blockchain holds the potential to distribute complex and data-intensive tasks through leveraging smart contracts, to unidentified nodes across the network. The expansion of blockchain size and associated storage costs poses a challenge for maintaining a decentralized and secure network. Segments, or also known as blocks, are a fundamental component of blockchain technology. The use of segments is crucial for several reasons including security, scalability and consensus [5]. This guarantees the integrity and immutability of transaction data by maintaining efficiency and speed. A distributed ledger is a database variant that is spread out across multiple locations, institutions, or nodes in a network. This decentralization allows for greater transparency, security, and resilience than traditional centralized databases.

This paper discusses segmentation of blocks within their assigned segment without the need to access the entire blockchain. Each segment would have its own set of validators that are responsible for validating transactions and blocks within that specific segment. This approach can improve the scalability of blockchain networks by reducing the time required to propagate blocks across the network [6]. Since nodes only need to transmit blocks to other nodes within their segment, the network can potentially handle more transactions per second without overwhelming individual nodes.

The proposed research, organizes a defined quantity of blocks into segment, which is subsequently stored by multiple nodes. The approach put forth addresses the concern of a sole vulnerability point, wherein a malicious entity could obtain all instances of a blockchain segment and exit the system, causing irrevocable loss. The theoretical proof states by restricting the storage capacity for malicious entities and distributing each segment across a group of cloud-based blocks, the storage requirements are significantly decreased in comparison to standard designs. Thus, the proposed system establishes a blockchain capable of managing big data by minimizing storage demands while ensuring data protection for users. The system achieved a 33% reduction in storage space during the experiment. This renders the suggested approach capable of effectively handling substantial volumes of data and its processing.

The next section of the paper is literature survey, later methodology is explained and is followed by implementation and results. Later the paper is concluded.

2 Literature Survey

Blockchain is a type of decentralized ledger technology that was first described in literature. It was initially used primarily in the field of cryptocurrency, with Bitcoin and Litecoin [7], Monero [8], and Zcash [9] being the most prominent examples. As blockchain technology has developed rapidly, it has become an effective means of ensuring the genuineness, safety, and dependability of data. It has been applied in a wide range of areas, including medical data, safeguarding personal information, and strategies for distributing data. Blocks serve as the fundamental components of a blockchain, comprising a segment header with primary data and transactional information as a block body.

Block data is utilized for establishing a connection to the preceding block and for indexing data using the hash value of the current block. Transactions within the blockchain involve engagement with a hash function, thereby ensuring security. The security of the blockchain system is of utmost importance and encompasses various aspects such as data security, security of smart contracts, safeguarding privacy, and mitigating application risks. In order to guarantee the integrity of data immutability, constant enhancements are required in the foundational data structure, cryptographic techniques, and communication networks of the blockchain. These improvements are essential to foster the robust advancement of blockchain applications.

The decentralized and unidentified characteristics of blockchain guarantee that correct results are recognized by the majority as far as the security threshold is upheld [10]. The issue of excessive growth raises the storage demands for participants, creating challenges for handling data-intensive tasks such as training AI models within a decentralized system, particularly when the system maintains a universally open membership approach. Few researchers aim to alleviate individual load and address the predicament of balancing the capacity to handle all aspects, sustaining a decentralized, and high-performance architecture.

Weighted models distribute the responsibilities of a node based on their weights [11]. An instance of a weighted model is the lightweight node system, where a lightweight node refrains from storing blocks; instead, it operates as a client to certain full nodes. These nodes verify a new transaction through Simple Payment Verification (SPV) queries. Lightweight nodes consume a maximum of 4.2 megabytes, irrespective of the overall blockchain dimensions, they cannot verify new blocks. Delegated Proof of Stake (DPoS) exhibits superior performance since representative nodes often possess advanced computational capacity, storage capabilities, and network bandwidth [12, 13].

A co-signed contract involving both parties is often made in off-chain approaches and this marks the beginning of the deal [14]. Off-chain channels perform secure trading, without publishing transactions through blockchain. Transactions are published to the blockchain only when there is a violation in off-chain transactions., There are few non-financial applications for these approaches. Instead of broadcasting the assignment and outcome to the network, entities employing off-chain techniques need to ensure confidential communication, by undermining the inherent anonymity of the blockchain.

Methods for blockchain sharding involve the allocation of nodes across distinct shards, dividing storage and assigning tasks to various shards that run in parallel, guaranteeing that the workload for individual nodes remains manageable with the global increase in transactions [15]. The primary focus of sharding design is to decrease the likelihood of an adversary exerting control over a majority of the inside shard positions, even though they haven't acquired the majority of nodes across the entire network. If an adversary controls a shard but does not meet security thresholds globally, the integrity of the entire system's security is jeopardized. Therefore, to maintain system security, there are stringent requirements on the quantity of shards and nodes within each shard.

Clients utilizing cloud computing across diverse application domains seek assurance regarding the accuracy and reliability of their data [16]. To establish a tamper-proof cloud computing environment, blockchain, ledger, is employed in conjunction with cloud technology. The distributivity of blockchain means the absence of central governing body or single point of failure. Instead, multiple nodes in the network work together to validate and record transactions, which helps to ensure the integrity and security of the ledger.

Blockchains utilize technologies and techniques to maintain the integrity and security of the ledger. Distributed (peer-to-peer) networks allow for the validation and sharing of information across the network. Encryption and cryptography methods are also crucial components of blockchain technology, as they are used to secure the data on the blockchain and prevent unauthorized access or tampering [17]. DLT is decentralized, immutable, transparent, secure, trustless, and relies on a consensus mechanism to validate transactions. Every newly appended block becomes immutable and irreversible, with transactions being documented in a publicly accessible ledger, visible to every participant. Transactions are validated by the network participants, and trust is not required between participants [18]. Consensus among all network participants is necessary to validate a transaction before incorporating it into the Blockchain.

Segment Blockchain has a primary focus on reducing the size of the blockchain. In blockchain sharding systems, the security depends on the majority of nodes in every shard being honest. However, in terms of storage, if an adversary fails to possess segment copies, their attack will fail to achieve any objective. Hence, within the Segment Blockchain, the honest nodes don't need to constitute the majority among those storing a specific segment. The only requirement is that each segment has at least one faithful keeper. With this more lenient security, it is possible to assign a reduced count of nodes to store a segment securely, as opposed to blockchain sharding systems, which require the majority of nodes to be honest. Hence, the storage can be significantly reduced.

3 Methodology

This section discusses the basic techniques in a segmented blockchain system including the implementation of a Cryptography system with AES algorithm. It has Logical Block Addressing (LBA), and leveraging cloud storage. In distributed ledger, data is stored in multiple copies across different nodes in the network, and each node has a copy of the entire ledger. Changes to the ledger are made through a consensus mechanism, in which network nodes collaborate to collectively validate and authenticate transactions. Once a transaction is validated and appended in ledger, it is replicated across all nodes ensuring that all copies of the ledger are kept in sync [19].

Segment Blockchain is appropriate for many of the current blockchain applications, such as notation, and identity control. These applications do not necessitate high rates of transaction throughput but do advantage from decentralization. Additionally, Segment Blockchain can facilitate integrating blockchain into an IoT ecosystem where edge devices might have limited storage to store a complete record, and the systems do not demand substantial transactions per second. Blockchain is a method of recording data in a manner that prevents alteration, unauthorized access, or manipulation of the system [20]. Every block within the chain holds a group of transactions, and when a fresh transaction transpires on the blockchain, a notation of it gets appended to the ledger of each participant. In Blockchain, transactions are logged using an unalterable cryptographic signature known as a Hash.

3.1 Implementation of Cryptography with AES Algorithm

Cryptography involves converting plain text into an encoded form referred to as ciphertext which is understood by authorized individuals who possess a decryption key. Cryptography provides secure communication over insecure networks, such as the Internet. AES (Advanced Encryption Standard) stands as a symmetric-key block cipher algorithm created to supersede the outdated Data Encryption Standard (DES) algorithm [21]. The size of the block and the length of the key can vary, but AES typically uses 128-bit blocks and keys of either 128, or 256 bits. The algorithm uses a series of permutation operations to transform the plaintext into ciphertext.

Additionally, AES is a symmetric key algorithm, it is easy to implement and use, which makes it a popular choice for many security applications. AES is considered to be a very secure encryption algorithm and is resistant to attacks, including brute-force and known-plaintext attacks. The relationship between the plaintext and the ciphertext is highly complex and difficult to reverse-engineer without knowledge of the encryption key.

Hashing is a process in which an input (such as a file, message, or password) is passed through a mathematical function that produces a hash with a constant-length output. Hash functions are one-way, which means that generating a hash from an input is a straightforward process, but very difficult to generate the actual input from its hash. This property makes hash functions useful for a wide range of applications, including data integrity checks, digital signatures, and password storage. One of the primary benefits of using hashing is that it allows for efficient and secure storage and retrieval of data.

3.2 Hashing with Message Digest Algorithm (MDA-5)

MD5 algorithm is a hash function generating digest, 128-bit for any input, irrespective of its length. The resultant digest is commonly displayed as a hexadecimal sequence with 32 digits. MD5 is extensively employed as a one-way function, implying it as simple to compute hash value from the message, but intricate to reverse-engineer the initial message from the hash. MD5 is commonly used for data integrity checks, changing the input message results in variant hash value, making it useful for verifying that the original data has not been tampered with. It is also used for storing passwords by hashing them with MD5 and storing the new hash instead of password. Hence, if the stored hash value is modified, the original details cannot be directly obtained, since reversing the hash function is very difficult.

3.3 Map Reduce

MapReduce is a programming paradigm and implementation that enables parallel and distributed processing of large data. The key idea behind MapReduce is to divide a large input data set into smaller chunks, process each chunk independently in parallel, and then combine the results of these independent processes. The map function applies a transformation to each data element in the input data set and produces key-value pairs. Subsequently, the reduce function merges the values linked with each intermediate key to generate a conclusive output.

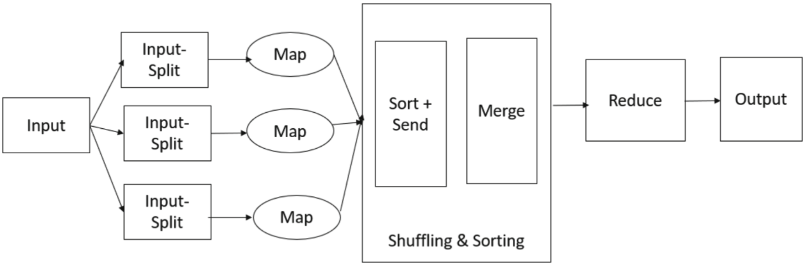

Fig. 1. MapReduce process flow

MapReduce is a flexible programming model and can be used for a wide range of applications, including data analysis and warehousing and implemented on a distributed system, making it a versatile tool for large-scale data processing [22]. MapReduce can distribute data processing across a large number of nodes, enabling efficient processing of large amounts of data and improving scalability in blockchain. MapReduce process flow is in Fig. 1. On the other hand, a reducer class handles the reducing phase, which involves aggregating and reducing the output of various data nodes to produce the final output. The data to be processed using MapReduce is stored in HDFS, and it can be a single or multiple files. The input determines the specification and specifies how files will be separated and read. During reducing phase, reducer processes the map-generated data by applying operations through a reducer function. The final output, is a smaller set of tuples, is stored in HDFS.

3.4 LBA

LBA is used in MapReduce to allow the operating system to address blocks of data on the disk using logical block addresses, rather than physical block addresses. This helps to simplify the process of accessing and managing data on the disk. In MapReduce, data is typically stored in a distributed file system across multiple nodes, such as HDFS. Data is typically partitioned into blocks of a fixed size, such as 64 or 128 MB. In HDFS, each block is assigned block ID, used to track the location of the block across the distributed file system.

When data is processed using MapReduce, the MapReduce framework assigns a task to a node to process a specific block of data. The node retrieves the block from the distributed file system, processes it using the MapReduce task, and then writes the output back to the distributed file system. In MapReduce, the processing of data is typically handled by a large number of worker nodes, while a separate set of master nodes are responsible for coordinating and managing the overall job. The worker processes data by executing Map and Reduce functions on the data blocks assigned to them.

3.5 Cloud Storage

In the proposed solution, involving cloud storage, the platform utilized is Drive HQ, which happens to be the pioneer in cloud IT solutions. Its inception in 2003 was aimed at providing a one-stop-shop for all Cloud IT solutions. Over time, DriveHQ has grown to become a leader in essential areas, with a minimalist approach that simplifies services through drive mappings. The free basic Plan, offer 5 GB, while family packages come for $4 per month. DriveHQ secures sensitive data through 256-bit SSL, with HTTPS or FTPS being the file transfer methods available.

4 Implementation and Results

The proposed system employs blockchain technology to encrypt data blocks, enabling a distributed computing method known as MapReduce. This strategy involves generating blocks prior to uploading them, while the database stores the metadata for these blocks. Moreover, Segment Blockchain includes a secure deduplication feature that utilizes an ownership protocol. This prevents any side-channel information exposure in deduplication.

The proposed system aims to offer a secure and efficient means for users to upload their files to the cloud. This is accomplished by employing both blockchain technology and sophisticated data deduplication techniques that ensure the confidentiality and integrity of data. User picks up the file from the local device. Upon selection, the file is divided utilizing LBA. This ensures that every block has a predetermined size with data alignment in a specific boundary. To guarantee distinctiveness, the system generates an MD5 hash per block as a block-level, 128-bit fingerprint, produced by encoding the input.

Users choose a file from their local system and divide into smaller blocks using LBA later, MD5 hash is generated for the block. Figure 2 is file upload method. System validates the distinctiveness of block by contrasting their MD5 values with those of already present blocks in the database (DB). Upon detecting distinct block, metadata is uploaded in DB, followed by encrypting the block and depositing it in the cloud using a blockchain structure. Thus, the system provides immutability to the stored data.

Duplicate blocks identified in the verification process, will not be allocated to the private cloud, only, the block instance is updated. The system eliminates the necessity for storing multiple duplicates of identical data, resulting in lowered storage expenses and efficiency. The proposed file upload method has advantages, including capability to create blocks before data upload, enhancing both efficiency and security. The Proof of Ownership protocol has the potential to facilitate secure deduplication, leading to additional reductions in storage expenses and enhanced efficiency. Overall, these attributes render the suggested system well-suited to contemporary requirements for data storage and security. Through the eradication of the necessity to retain multiple replicas of identical data, the system concurrently lessens the potential for data loss. It remains crucial to rigorously assess any novel system, prior to its deployment to guarantee alignment with essential criteria and intended functionality.

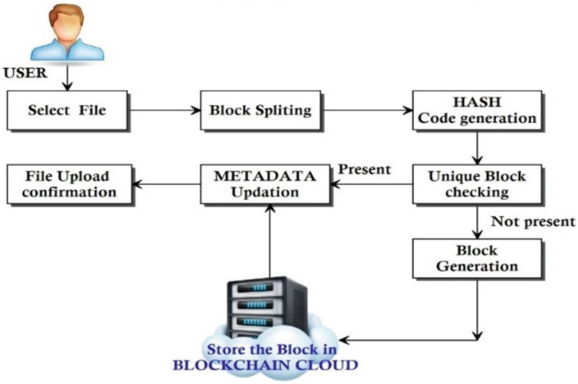

Fig. 2. File upload process

Figure 3 describes the process for downloading a file and partitioning it into smaller blocks with LBA. The user selects the file, the file is transferred from the server to the user's device in chunks. The file is divided into smaller blocks using LBA. Each block is assigned a unique address that indicates its location within the file. Overall, this process is designed to make it easier and more efficient to download and manage large files, by breaking them down into smaller, more manageable pieces that can be accessed and transferred more quickly and reliably.

➢ Algorithm 1: Method of Block Development

➢ Input: File Data Block

➢ Output: Block of Block Chain

```
for N blocks in upload queue
    Create the Root-Hash-code
    Fetch Hash code for the Previous Block (PBC)
    Generate Header PBC-TimeStamp-Nonce
    Data encryption and the creation of the Block Body
    Combine Body and Header for block formation
otherwise
    wait and check
```

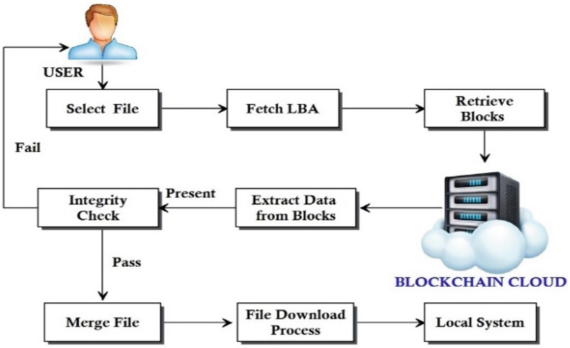

Fig. 3. The proposed file download process

To ensure that each block retains its uniqueness and integrity, a unique MD5 hash per block is created, serving as a digital signature that verifies the block's data. This hash is produced by applying the MD5 algorithm to the block's input data, resulting in a 128-bit output that remains the same for identical input data. The merge file feature combines multiple blocks into a single file, which users can download to their local systems. The auditor verifies whether the blocks are intact or not. For verification, Algorithm 2 is used.

In comparison to traditional blockchain systems, the proposed system has achieved optimized storage. Block storage optimization reduces the amount of storage while maintaining the confidentiality and integrity of the system. Storage optimization is important because blockchain requires a significant storage space to store data. These findings suggest that the suggested system is a very secure and effective way to store transaction in a hybrid cloud by offering faster uploading and downloading times and optimised storage space utilization.

```
➢ Algorithm 2: Block Verification

➢ Input: Block ID
➢ Output: Block Status

if Block count > 0 then
    List the details
    Retrieve block, Test and Extract
    Generate Hash for transaction
    Detect Previous Hash Code of the Block
    Compare the Hash Codes
    Print Result
else
    Display – Verification Completed
```

The suggested system has been implemented and assessed within a hybrid cloud setup, confirming the proper functionality of all four user capabilities. A comparison has been conducted regarding the process of uploading blocks and downloading times to evaluate the system's performance. Based on the results presented in Fig. 4, it appears that the system demonstrates quicker block upload and download times in contrast to conventional blockchain systems. This is a positive indication that the system is meeting the requirements and specifications that were set out for it.

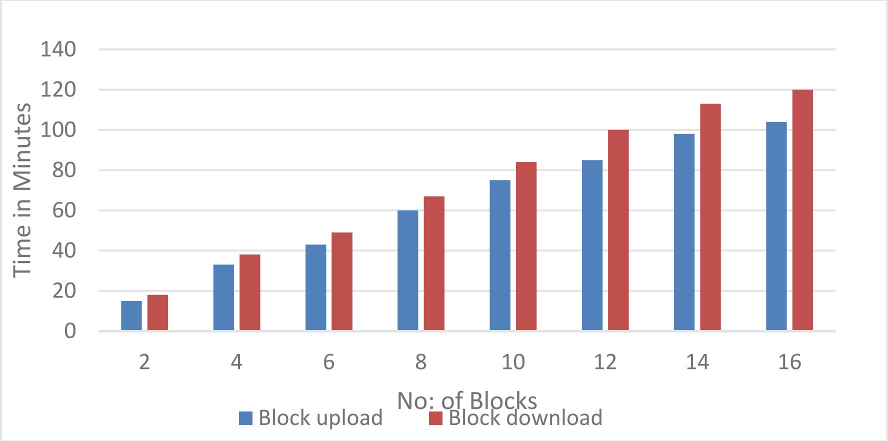

Fig. 4. Time is taken Vs No of blocks for File upload and download graph

Initially, the count of files uploaded is 30 files and later divided, and saved in cloud without the map-reduce algorithm with a total block of 850. However, the block count was decreased to 415 using the map-reduce approach. This is mentioned in Table 1. Hence, the findings indicate that the proposed method is effective and secure for hybrid cloud data storage. In comparison to existing blockchain, it is capable of offering faster uploading and downloading times and optimized storage space utilization.

Table 1. Total and Unique Blocks

File Count	Total Block	Unique Block
5	150	88
10	320	174
15	450	209
20	580	318
25	720	378
30	850	415

5 Conclusion

Blockchain creates a highly secure, transparent, and decentralized system for recording and verifying transactions. A blockchain is a digital ledger, that uses cryptography and distributed computing to record transactions transparently. Distributed ledgers are often used in applications such as cryptocurrencies, supply chain, and voting systems, with significant benefits in terms of security, transparency, and efficiency, and is likely to continue to grow in popularity in the years to come. Segmented Blockchain can improve sharding by dividing the storage of transactions from the process of transaction validation. The proposed Segmentation Blockchain methodology reduces storage requirements for the blockchain system without compromising decentralization or security. Data analysis shows that Segment Blockchain significantly decreases data storage demands, making it advantageous for data-heavy blockchain applications. Distributed Ledger based Blockchain is a powerful technology with transparency, security, consensus, decentralization and immutability as unique features. The proposed model creates a blockchain system that can effectively handle large-scale datasets and also ensures security and privacy of user data. In addition, reducing the storage space requirements can also help to lower the costs associated with running and maintaining the blockchain. By reducing the storage space needed, the proposed system could make blockchain technology more accessible and efficient. However, the effectiveness of this approach would depend on the specifics of the implementation and the characteristics of the network in question.

Funding. There is no funding for this research publication.

Conflict of Interest. The authors declare the absence of any conflicting financial interests. They also assert that the study was conducted without any involvement of financial affiliations that could be perceived as a possible source of conflicts of interest.

Contributions. Each of the authors made an equal contribution to the preparation of the manuscript.

References

1. Biryukov, A., Tikhomirov, S.: Security and privacy of mobile wallet users in Bitcoin, Dash, Monero, and Zcash. Pervasive Mob. Comput. **59**, 101030 (2019)
2. Mahapatro, R.K., Ali, A., Ramakrishnan, N.: Blockchain segmentation: a storage optimization technique for large data. In: 2023 8th International Conference on Communication and Electronics Systems. ICCES. IEEE (2023)
3. Luu, L., et al.: A secure sharding protocol for open blockchains. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (2016)
4. Divakaruni, A., Zimmerman, P.: The lightning network: turning bitcoin into money. Finan. Res. Lett. **52**, 103480 (2023)
5. Jere, S., et al.: Recruitment graph model for hiring unique competencies using social media mining. In: Proceedings of the 9th International Conference on Machine Learning and Computing (2017)
6. Sharon Priya, S., Ali, A.: Localization of WSN using IDV and Trilateration Algorithm. Asian J. Eng. Technol. Innov. **4**(7) (2016)
7. Tu, Z., Xue, C.: Effect of bifurcation on the interaction between bitcoin and litecoin. Finan. Res. Lett. **31** (2019)
8. Dong-Her, S., et al.: Verification of cryptocurrency mining using ethereum. In: IEEE Access 8, 120351–120360 (2020). 2018 Crypto Valley Conference on Blockchain Technology, CVCBT. IEEE (2018)
9. Akcora, C.G., Gel, Y.R., Kantarcioglu, M.: Blockchain networks: data structures of bitcoin, Monero, Zcash, Ethereum, ripple, and iota. Wiley Interdisc. Rev. Data Min. Knowl. Disc. **12**(1), e1436 (2022)
10. Tschorsch, F., Scheuermann, B.: Bitcoin and beyond: a technical survey on decentralized digital currencies. IEEE Commun. Surv. Tutorials **18**(3), 2084–2123 (2016)
11. Palm, E., Olov, S., Ulf, B.: Selective blockchain transaction pruning and state derivability. In: 2018 Crypto Valley Conference on Blockchain Technology, CVCBT. IEEE (2018)
12. Patil, S.S., Ali, A., Ajil, A.: Approaches for network analysis in protein interaction network. Int. J. Hum. Comput. Intell. **2**(2), 47–54 (2023)
13. Vasin, P.: Blackcoin's proof-of-stake protocol v2, **71**. https://blackcoin.co/blackcoin-pos-protocol-v2-whitepaper.pdf (2014)
14. Burchert, C., Decker, C., Wattenhofer, R.: Scalable funding of bitcoin micropayment channel networks. Roy. Soc. Open Sci. **5**(8), 180089 (2018)
15. Ali, A., Sumalatha, D.P.: A survey on balancing the load of big data for pre-serving privacy access in Cloud. Asian J. Eng. Technol. Innov. (AJETI), 176 (2018)
16. Varshney, T., et al.: Authentication & encryption-based security services in blockchain technology. In: 2019 International Conference on Computing, Communication, and Intelligent Systems, ICCCIS. IEEE (2019)
17. Kuo, T.-T., Kim, H.-E., Ohno-Machado, L.: Blockchain distributed ledger technologies for biomedical and health care applications. J. Am. Med. Inform. Assoc. **24**(6), 1211–1220 (2017)
18. Ølnes, S., Ubacht, J., Janssen, M.: Blockchain in government: benefits and implications of distributed ledger technology for information sharing. Gov. Inf. Q. **34**(3), 355–364 (2017)
19. Morkunas, V.J., Paschen, J., Boon, E.: How blockchain technologies impact your business model. Bus. Horiz. **62**(3), 295–306 (2019)
20. Song, X.-S., et al.: A markov process theory for network growth processes of DAG-based blockchain systems. arXiv preprint arXiv:2209.01458 (2022)

21. Ali, A.: Analytical study on fast and secure authenticated key agreement protocol for low power networks. IJCST **6**(4) (2015)
22. Ali, A., Hulipalled, V.R., Patil, S.S.: Centrality measure analysis on protein interaction networks. In: 2020 IEEE International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET). IEEE (2020)