Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Research on blockchain-enabled consistency enhancement techniques for on-chain and off-chain interactions of privacy data

Ke Zhao[1,✉], Wenyu Zhang[1], Lianchao Su[1], Xiaoliang Wang[1], Chenguan Li[1]

[1] *STATE GRID WEIFANG POWER SUPPLY COMPANY, Weifang, Shandong, 261041, China*

ABSTRACT

In order to improve the consistency of on-chain-off-chain interaction of private data supported by blockchain and reduce the redundancy of data storage performance, this paper applies an efficient data interaction method of prefix hashing with improved red-black tree index to store public indexes and improve the efficiency of retrieval and interaction of blockchain data. Under the idea of generalization, anonymous region (AR) is used to hide the real location of participating nodes and protect the privacy of realized nodes. To reduce the computational overhead of the selection process, a cooperative sensing location privacy preserving optimization mechanism, LPPOM, is proposed. The scheme in this paper has a slow growth of data size on the chain with higher storage efficiency, larger throughput, and shorter query time (0.1899ms). The time cost consumed when the number of privacy chains is 15, 30, and 60 only increases by 0.2309-0.4855ms compared to the single chain system, indicating that the scheme scales well. When the file size is within 200 and the number of encrypted attributes is less than 4, its total encryption time meets the user's privacy data encryption needs (between 66.1765-236.7081ms). The IPFS read/write module is able to satisfy the people's daily use needs under the public network conditions, and its read/write speed is between 0.1568 and 0.2639MB/ms (file <100M).

Keywords: Blockchain, Data on-chain off-chain interaction, Prefix hashing, Improved red-black tree indexing, Co-awareness

1. Introduction

In today's digital age, data consistency is critical to operations in all areas. However, data consistency is often a challenge due to complex information systems and distributed environments. To solve this problem, blockchain technology can provide an effective solution [1-4]. Blockchain technology is a

✉ Corresponding author.
E-mail address: qukuailianlunwen@163.com (K. Zhao).
Received 01 March 2024; Revised 25 May 2024; Accepted 10 November 2024; Published Online 16 April 2025.
DOI: 10.61091/jcmcc127b-493

distributed ledger technology that stores data on different nodes through decentralization. Data consistency is one of the core principles of blockchain [5-7].

Blockchain can be used to ensure data consistency by means of cryptographic hash functions, consensus mechanisms, smart contracts, timestamps and decentralization. It provides a secure and reliable way to store and transmit data, making it impossible for data to be tampered with, deleted or lost [8-11]. In today's era of information explosion, the accuracy and reliability of data is crucial. Blockchain technology provides a powerful tool for various fields to ensure data consistency, which in turn enhances the efficiency and trustworthiness of business. However, blockchain technology also has some challenges and limitations [12-15]. First of all, the performance problem of blockchain technology is an urgent need to be solved nowadays. Since the nodes need to agree and store the data of the whole blockchain, the performance of blockchain network will be limited in the case of large data volume and frequent transactions. Secondly, due to the decentralized feature of blockchain technology, it will become difficult to recover data once there is an error or data loss [16-19]. Therefore, for some application scenarios that do not require high data integrity, the traditional centralized database is more suitable.

Literature [20] systematically discusses the security and privacy properties of blockchain. By introducing the concept and utility of blockchain and reviewing the security and privacy techniques for implementing security attributes in blockchain-based systems, it is concluded that the results of the study are beneficial for the readers to understand the security and privacy of blockchain in terms of concepts, attributes, and techniques. Literature [21] emphasizes the wide application of blockchain technology and the impact it has on the healthcare industry. It is pointed out that blockchain technology plays an important role in securing patient data and healthcare supply management. Literature [22] proposed a personal data privacy protection scheme based on federated blockchain, which ensures the user's privacy and security by synergizing off-chain storage and on-chain transmission with lighter encryption and segmentation, and realizes the personal data privacy protection. The effectiveness of the proposed method is demonstrated through experimental results. Literature [23] proposes a blockchain-based data collection and processing architecture. The architecture ensures the security of loT data through data consistency. The DPA-PBFT algorithm with self-optimization capability is designed and the performance improvement of this algorithm under the proposed architecture is evaluated through many experiments. Literature [24] employs blockchain technology in order to address data security and trustworthiness issues. The mobile agent technology is adopted to deploy a distributed virtual machine agent model, through which the integrity protection framework based on blockchain is constructed. The research results verify the effectiveness of blockchain technology. Literature [25] introduces the blockchain-based AMI security framework, and reshapes the AMI data management system architecture by combining the characteristics of blockchain. And the blockchain's consensus algorithm and other technologies are utilized to solve the security problem of AMI data. The feasibility of blockchain technology for AMI is verified through analysis and experiments, demonstrating the reliability of the framework. Literature [26] examines the terminology and current status of auditable data structures, and proposes a generalized framework based on blockchain capable of privacy-preserving auditing. A detailed description of the framework implementation and experimental results are provided and the effectiveness of the framework in proof generation and evaluation is emphasized. Literature [27] designed a blockchain-based decentralized model that consists of collaborative verification nodes to avoid malicious tampering. And proposed data integrity verification algorithm. The experiments reveal the superior performance of the proposed method. Literature [28] proposes a blockchain privacy protection scheme based on zero-knowledge proof for secure data sharing. The theoretical analysis of this scheme shows that this its ability to satisfy the confidentiality requirements of security, integrity and validity. Literature [29] proposed the SOC approach to achieve data security in government. It is pointed out that the method is able to realize the trustworthiness of data content and the controllability of data

ownership. By applying it in practice, it is confirmed that the SOC method provides a feasible solution for sharing data in government. Literature [30] proposed an efficient and secure data consistency verification scheme based on blockchain technology. A Merkle hash tree is constructed by using cryptographic tags to generate unique and lightweight validation. Theoretical and experimental analyses emphasize that the scheme has excellent performance in terms of security and verification speed.

In order to solve the problem of low effectiveness of blockchain data interaction in industrial Internet, it is proposed to utilize the combination of blockchain technology and off-chain database, supplemented with prefix hash indexing method and red-black tree construction method to store industrial Internet data information, to realize the key data keywords on-chaining and with a large number of data off-chain storage mechanism to achieve efficient and safe search. Considering the location privacy and data consistency of side-chain nodes in the process of cooperative sensing, this paper proposes the location privacy protection optimization algorithm of anonymous region and the corresponding data consistency optimization method to improve the consistency of the interaction between on-chain and off-chain of private data supported by blockchain.

2. Blockchain-enabled privacy data on-chain and off-chain interaction technology

2.1. Blockchain technology

2.1.1. Overview of Blockchain. Blockchain combines blocks of data into specific data structures in chronological order in the form of chains, and multiple blockchain nodes store block data in the form of a shared ledger, the distributed shared ledger [31] network is shown in Figure 1. The operations of querying and updating the data on the blockchain are constructed in the form of specific messages, often called transactions. Transactions are packaged into blocks and linked to the previous block via the block's cryptographic hash. Operations on shared ledger data are required to be broadcast into the blockchain network by constructing a new block, which undergoes a consensus mechanism to achieve consistency among nodes and is stored.

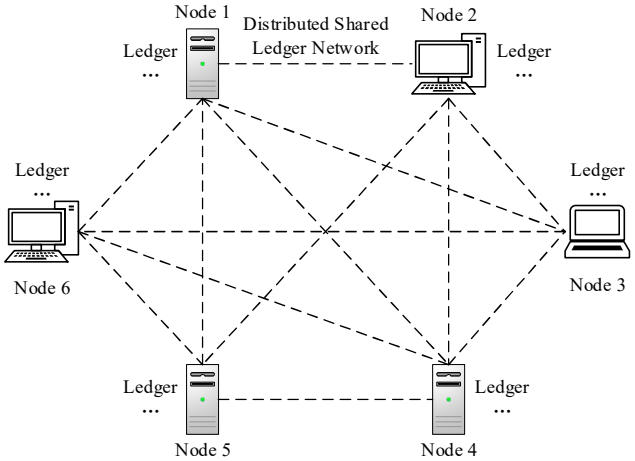


Fig. 1. Distributed Shared ledger network diagram

Blockchain technology has the following characteristics:

Distributed architecture: blockchain networks have a distributed architecture, where block data is not stored and managed centrally by a centralized authority, but is stored dispersed across nodes in the blockchain network. Each node has a copy of that shared ledger, and the copies are updated synchronously and consistently during the consensus process. Decentralized Trust Mechanism:

Blockchain builds a decentralized trust mechanism through distributed architecture and open and transparent bookkeeping. Transactions between nodes do not need to be endorsed by a trusted third party, and all blockchain nodes keep accounts together, ensuring that transactions between any two parties can be carried out credibly. Open and transparent data: blockchain data is stored synchronously among nodes, and account and transaction information in the blockchain network is recorded on the blockchain. Without special encryption, block data can be retrieved from any blockchain node. The data cannot be tampered with: Blockchain adopts chained block structure with timestamps to store data, which is highly traceable and verifiable, and at the same time with cryptographic algorithms and consensus mechanisms to ensure the blockchain's tamperability. Blockchain protocol can be divided into public chain and coalition chain based on the different ways of node access control. In public chain networks such as Bitcoin, any organization or individual is able to build nodes to join the network or withdraw at any time. In coalition chain networks, due to better network conditions and node authentication mechanisms than public chains, the consensus algorithm can achieve higher throughput and lower latency based on more trust between nodes than public chains. In the scenario considered in this study, the number of enterprises or departments related to a particular IoT data is limited, and they need to coordinate and agree with each other in advance before they can interact with the data. Therefore, the federation chain is more suitable as the underlying architecture of the IoT data sharing system.

2.1.2. Consensus Algorithms. Blockchain ensures the consistency of block data between blockchain nodes through a consensus process. The consensus algorithm [32-33] involves how to add new blocks to the blockchain network while ensuring that each active working node is able to add consistent blocks. The process in which permissions to submit new blocks are assigned and new blocks are published into the network usually varies depending on the consensus algorithm. In public chain systems, since nodes are able to join or leave the network at any time, consensus algorithms need to provide incentives for nodes to be online and participate in the consensus to ensure that the network is active.Algorithms such as POW and POS provide incentives by giving rewards to nodes submitting new blocks.The POW algorithm employs the arithmetic competition for the right to make a block and the longest chain mechanism to ensure the consistency of the data and the fairness of the incentives. The nodes participating in the consensus can only find a random number that satisfies certain conditions by traversing, and the node that finds the random number obtains the block right of the current block. When adding a new block to the blockchain, other nodes are able to verify that the random number satisfies the conditions. The blockchain with the maximum block height is a valid blockchain, so nodes that receive longer chains will switch from the short chain to the long chain, and eventually all nodes in the blockchain network will reach agreement on the block data within a certain period of time, then it is guaranteed that the data will gradually converge to be consistent after a sufficient amount of time.

In a coalition chain network, the nodes in the blockchain network have agreed in advance on the power of their respective obligations, so no additional consensus incentives are required. The main purpose of the consensus process in a coalition chain network is to achieve distributed consistency under conditions that can withstand a Byzantine attack. The Byzantine attack consists of any one of the node's error methods and possible attacks, and its consistency is needed to ensure system security and activity. Threshold conditions for consensus algorithms capable of resisting Byzantine attacks to maintain security and activity in asynchronous networks have been demonstrated.

2.1.3. Smart Contracts. A smart contract [34] is a collection of code and data that can be deployed for execution on a blockchain network. Similar to blockchain data, smart contracts also ensure the consistency of the computing process through a consensus mechanism. Smart contracts are able to be written and deployed based on rules agreed upon in advance, and when trigger conditions are met, operations such as adding, modifying, etc. are performed on some state variables stored on the

blockchain. Smart contracts are executed by all nodes participating in the consensus at the same time, and the input and output variables of their arithmetic logic are data on the blockchain, so the execution results of smart contracts are also consistent and tamper-proof.

Before a smart contract is deployed, the algorithmic logic needs to be encoded into a specific form, and the algorithmic logic is submitted to the blockchain by initiating a transaction or creating a contract account. It provides a certain indexing method that allows the user to invoke the execution of the smart contract, and the user invokes the smart contract by submitting the transaction containing the index. Based on the algorithmic logic and transaction inputs, the smart contract reads and manipulates the state variables of the contract or invokes other deeper contracts, and ultimately returns the results to the user. Therefore, in a public chain network, the transaction that invokes a smart contract is usually billed based on the complexity of the smart contract's algorithmic logic. Similarly, in federated chain networks, the design of smart contracts should try to optimize the complexity of algorithms, so as to avoid the execution of overly complex algorithms in smart contracts.

2.2. Blockchain-enabled on-chain and off-chain interaction methods for privacy data

2.2.1. Problem Description and Requirements Analysis. The first of the existing problems is that industrial data storage services face complex and diverse security risks, and the second aspect is that due to the large-scale and heterogeneous nature of industrial data, it makes the data redundant in storage resources in the whole process flow links such as transmission, storage, utilization, and processing of private information, which in turn leads to a decline in the ability of data retrieval. In this paper, this paper makes full use of the characteristics of the blockchain, decentralized design to make up for the existing tree structure is vulnerable to a single point of attack resulting in system paralysis shortcomings. The design architecture of prefix hashing, improved red-black tree indexing, and stack proposed in this paper separates data information for processing on demand. Not only does the off-chain separation of large amounts of data storage to reduce the redundancy of the blockchain structure, but also allows sensitive private information to be protected and not easily publicized.

2.2.2. Prefix Hashing and Improved Red-Black Tree Interaction Methods for Up and Down Chaining:

1) Prefix Hashing and Improved Indexing. Most of the existing on-chain and off-chain storage schemes belong to Diffie-Hellman or traditional mapping scheme (TBS), and the indexing mode also utilizes the cloud service or common database own index form. This scheme proposes a prefix hash blockchain construction model with improved red-black tree indexing to improve the efficiency of on-chain + off-chain indexing and optimization of storage resources.

2) Chain data separation and prefix hash indexing. Considering that industrial data contains a large amount of information, such as characters representing countries, domains, enterprises, countries, domains, enterprises and other information is bound to exist part of the same code, just like the beginning of IP address 182 can represent an area network, because of its characteristics this paper innovatively proposes a prefix hash indexing mechanism, which will be the attribute data of the industrial Internet, timestamps, IP addresses, port numbers, relevant protocols and other information as data metadata stored in the off-chain database. Construct prefix hash index in the on-chain part: according to the data structure, construct the prefix hash index cluster EK=(EK1,EK2,.... ,EKn), here is based on the longest prefix to generate EK, and prefixes followed by different data information is constructed to generate FExy, and Ex prefix match to form a prefix hash index similar to hash index [35] form of prefix hash index, which is the origin of the name of this method. Finally the prefix hash index cluster is added to the blockchain header to form an extended block header.

In response to the problem of infinite expansion of block header that may be brought by massive data, this paper adopts the scheme of limiting the block size and deciding the prefix of block header

inversely by the number of data in the block body. A comparison of the three chain form mechanisms is shown in Table 1, and 6 blocks are assumed in this section to show the partial correspondence between prefixes and data. The default block size in the system is 15MB, of which the header is 3MB and the block body is 12MB, the maximum size of each piece of industrial data specified is 2KB, and 10,000 pieces of data can be stored in each block body after rounding down the calculation, which corresponds to the prefix of at least 1 (50byte), up to 1000 prefixes (0.05MB).

Table 1. Comparison of three chain forms mechanism

Block serial number	Block header - prefix	Block body - data
1	EK_1	$F_{1-1} \sim F_{1-10,000}$
2	$EK_1, EK_2 \vdots EK_{10,000}$	$F_{1-1}, F_{2-1} \cdots F_{10,000-1}$
3	EK_A, EK_C	$F_{A1} - F_{A6,000}, F_C$
4	EK_B, EK_D	$F_{B1} - F_{B5,000}, F_D$
5	EK_E, EK_F	$F_{E1} - F_{E6,000}, F_{F1} - F_{F4,000}$
6	EK_G	$F_{F4,001} - F_{F5,000}, F_G$

3) Index improvement based on red-black tree. Aiming at the characteristics of industrial data, the data is deconstructed into α-complete and β-enterprise indexes and uploaded into the customized block body, in which the arrangement of the two corresponding indexes follows the rules of the red-black tree, and the red-black tree-based data structure arrangement constructed by the two indexes will be addressed by the automated code within the smart contract to interact with the off-chain database, in which the embedded β-enterprise indexes are indexed directly into the cache stack to improve indexing efficiency. Red-black tree & Buffer stack is customized for its characteristics, experiments can be seen under the massive data only query performance, the structure of the red-black tree is better than the underlying MySQL database B-tree structure, LevelDB and jump table structure, in the experimental part of this paper, a performance comparison to be confirmed. The stack structure is used to retrieve the data from the previous retrieval pressed into the stack in a fast and sequential manner. After repeated calibration, the height of the stack is set to logn, where n is the total number of data within the system. The idea of improvement in this paper is to directly check the existence of two child nodes on the path that are both red when finding the insertion position and solve the problem immediately to complete the task in a specified time.

2.2.3. On-chain and off-chain smart contract design:

1) Overall smart contract process. The overall smart contract process is shown in Figure 2. In the user side can be man-made, IoT devices query industrial data, it also needs to send a query request through a query smart contract, through the prefix hash index in the blockchain nodes in the way of anchoring nodes, and by its block header information points to the index of all the corresponding blocks, to get a complete index and enterprise-level index returned to the query smart contract to continue to carry out step (7), the red-black tree & Buffer's indexing The way to get information from the DB. Finally, under the operation of the feedback smart contract in step 8, the retrieved information is fed back to the user to complete a complete retrieval process.

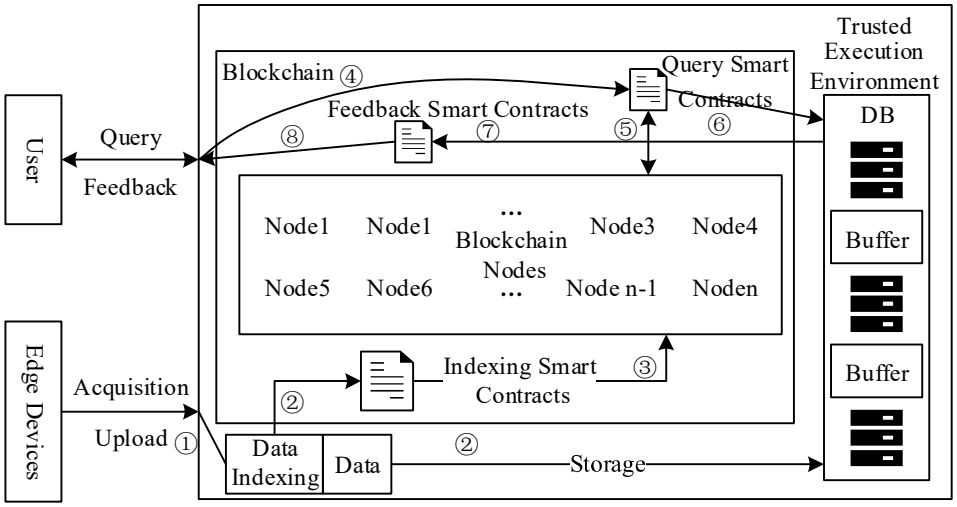


Fig. 2. Flow chart of intelligent contracts

2) Smart Contract Interaction. Query smart contract: This smart contract contains four functions, their role is to obtain the end of the query request and then interact with the blockchain nodes, database. Among them, GetQuery() immediately executes FindNode() after obtaining the request to find the specified node according to the prefix hash index method proposed in this paper, and then GetIndex() will input the industrial Internet data to be queried as a parameter to execute, and then according to the node's block header information to get the complete, enterprise-level index stored in the block body pre red-black tree & Buffer retrieval, and finally QueryDB() will be used to retrieve the data from the node, and finally QueryDB() is used to retrieve the data from the node, and the database. Buffer retrieval, and finally QueryDB() interacts the index with the external database to complete the smart contract function.

Feedback smart contract: This contract is relatively simple, only contains two functions GetDataFromDB() and PushDataToUser(), in the Database query to the Data as an input to the smart contract, the role of the database query to obtain the number and push feedback to the demand initiator. Considering the efficiency of third-party external components, this paper utilizes smart contracts for data feedback without using a predicate machine.

Index on the chain smart contract: This contract is designed to obtain the data index after the separation of the complete and enterprise-level index, respectively, by GetOnlyIndex () and SplitIndex (), which is an internal loop to determine and maintain the two stacks EntireArr [] and EnterpriseArr [] used to build the two different sets of indexes, and then pushed to the blockchain nodes for subsequent storage of the indexes by the The PushNode() function pushes to the blockchain node for subsequent index storage.

3) Retrieval process. The overall retrieval process and the innovative research proposed in this paper are more in the form of internal algorithms, and the manifestation of the surface is not well concretely displayed except in the efficiency quantization, but in the interaction process can be a further understanding of its execution process. First indexed to the prefix part, and then in the chain string to find the corresponding index, retrieval is completed will get its data where the specific block number, may be in this block, may also be distributed in other blocks, this randomness ensures that a large amount of data under the retrieval efficiency will not be reduced.

2.3. Coordination-aware privacy data consistency optimization approach

Compared with the traditional single-chain structure of the perceptual system, the introduction of side chains and the construction of smart contracts help to process transactions in parallel and largely

improve the throughput and processing efficiency of the system. At the same time, the sidechain can also ensure the privacy of the records on the chain and the anonymity of the participating nodes to a certain extent. However, the sensitive information of task execution nodes in the sidechain may still be leaked to other nodes, such as location information, which may affect the enthusiasm of node participation. Therefore, in this paper, we propose a spatial location privacy preserving algorithm (LPPOM) combined with the idea of generalization, which utilizes the anonymous region instead of the actual location of the participating nodes in order to facilitate the preservation of location privacy.

2.3.1. Co-aware location privacy preserving optimization algorithm. In order to protect the location privacy of the nodes on the chain, this paper combines the idea of generalization and uses randomly constructed anonymous regions (ARs) to hide the real locations of the participating nodes to achieve location privacy protection of the nodes. However, the uncertainty of the anonymization region makes it more difficult to select the appropriate task execution node in the case of location privacy protection. Meanwhile, the anonymization regions constructed by different nodes may overlap, increasing the computational overhead of the selection process. Therefore, this section proposes a privacy-preserving query optimization mechanism, LPPOM, to reduce the computational overhead of the node selection process for collaborative sensing [36] without compromising the security and privacy of the system. In the IoT collaborative sensing process, if the participating nodes want to accept and perform tasks, they need to submit the corresponding job information, which also includes location information. Based on these location data, the blockchain system will automatically create random anonymized regions to generalize the real location using smart contracts, and construct the corresponding query anonymized regions (QARs) based on the predefined query radius r to recruit the appropriate nodes for task execution.

When a circular region is used as the shape for constructing an anonymous region, the area of the total query region S_c can be calculated as follows:

$$S_c = \pi(\tilde{r} + r)^2 \tag{1}$$

where $\tilde{r}$ represents the radius of the circular anonymized region and $\pi(\tilde{r})^2$ represents the area of the region.

When a rectangular region is used as the shape for framing the anonymous region, the area of the total query region can be calculated as follows:

$$S_r = ab + 2(a+b)r + \pi r^2 \tag{2}$$

where a and b denote the two sides of the rectangle and ab denotes the area of the region. In order to simplify the calculation, this paper mainly uses the circular region as the shape for constructing the anonymous region. The specific location privacy protection optimization process is as follows:

1) Generate initialized anonymous region: in the process of task execution node selection, combining $k-$ the anonymity idea and privacy requirements $req = (k, \tilde{r})$, on the basis of the node's real location, the system continues to find other $k-1$ task execution node locations within the neighborhood to obtain $\{L_1(x_1, y_1), \ldots, L_{k-1}(x_{k-1}, y_{k-1})\}$, and then calculates the k initialized random anonymous region $\{AR_0, AR_1, \ldots, AR_{k-1}\}$, and satisfies:

$$\begin{cases} Center(AR_i) \neq L_i(x_i, y_i) \\ Radius(AR_i) = \tilde{r} \end{cases} \tag{3}$$

where the first inequality indicates that the user's real location $L_i(x_i, y_i)$, $0 \le i \le k-1$ will not be situated in the geometric center of the randomly generated anonymous region, $L_0(x_0, y_0)$ is the actual location of the node, and $Radius(AR_i)$ represents the radius of the anonymous region AR_i.

2) Merging overlapping query anonymized regions: after generating k anonymized region $\{AR_0, AR_1, \ldots, AR_{k-1}\}$, the system will construct the corresponding query anonymized region $\{QAR_0, QAR_1, \ldots, QAR_{k-1}\}$ based on the predefined query radius r, and calculate the area $\{S(QAR_0), S(QAR_1), \ldots, S(QAR_{k-1})\}$ and determine whether there are overlapping query anonymized regions and whether it is necessary to perform the merging operation. In order to reduce the computational overhead in the sensing process, the query anonymized region should be guaranteed:

$$S_{\min} = \min \sum_{i=0}^{l} S(\overline{QAR}_i) \tag{4}$$

where, $S_{\min}$ denotes the smallest area sum of querying anonymous region $\overline{QAR}_i$ after optimization, $\overline{AR}_i$ denotes the ith anonymous region after optimization and satisfies $0 \le i \le l$, $0 \le l \le k-1$. The specific optimization process mainly consists of the following three steps:

Determine whether the anonymous regions overlap or not: in order to improve the selection efficiency of task execution nodes in the system, the location query overhead of the current node should be reduced as much as possible. If any two anonymous regions AR_i and AR_j can further reduce the overall query area by merging operation, they will be selected and merged by the system. Therefore, the conditions for determining whether two anonymized regions overlap should be satisfied:

$$QAR_{i,j} = \arg\min \left\{ S\left(QAR_{i \ne j}\right) \right\}, \forall i, j \in [0, k-1] \tag{5}$$

and are satisfied at the same time:

$$\begin{cases} Center\left(QAR_{i,j}\right) \ne L_i(x_i, y_i) \\ Center\left(QAR_{i,j}\right) \ne L_j(x_j, y_j) \end{cases} \tag{6}$$

i.e., the user position does not lie at the geometric center of the query anonymous region.

Selection of merged region: any two anonymous regions AR_i and AR_j can be merged by generating function Gen after judgment in order to generate a new anonymous region, i.e., $AR_{i,j} = Gen(AR_i, AR_j)$, and the corresponding new query region $QAR_{i,j}$ is able to be obtained based on the query radius calculation. If satisfied:

$$S(QAR_i) + S(QAR_j) \ge S(QAR_{i,j}) \tag{7}$$

Then the system will merge the anonymous regions, otherwise, no merge operation is performed.

Repeat the execution: determine whether the anonymous regions overlap and select the merged regions until there are no anonymous regions to be merged. Finally, the system will obtain the following set of anonymous regions:

$$\mathrm{Re}\, gions_{set} = \left\{ \overline{AR}_0, \overline{AR}_1, \ldots, \overline{AR}_l \right\} \tag{8}$$

Combining $k-$ the anonymity idea, the k locations of the participating nodes are used to generate anonymized regions $\{AR_0, AR_1, \ldots, AR_{k-1}\}$ associated with them, which are eventually

merged to obtain a collection of anonymized regions $\mathrm{Re}\,gions_{set}$ that can be used for the selection of nodes for the execution of tasks in the collaborative sensing process.

2.3.2. Data consistency optimization methods. In the first step, define the block confirmation time in the blockchain network as T_c, the number of sidechains as M, and the number of task execution nodes needed for a single task as N. Assuming that the validation time for a single transaction is T_t, for the agent nodes in the sidechain, the maximum number of transactions can be computed as $\frac{T_t}{T_t}$, and the number of transactions that can be carried out by all the agents as $\frac{T_c}{T_t}M$. In order to avoid signing up for a number of task execution nodes that are much higher than that needed for the actual task number, it should be satisfied:

$$\min \frac{T_c}{T_t}M - N \tag{9}$$

Define c as the number of redundant contracted nodes, i.e., $c = \frac{T_c}{T_t}M - N$. Then, $T_t = \frac{T_c M}{N + c}$ can be obtained. therefore, the value of T_t can be dynamically adjusted to reduce the system overhead due to contracting redundant task execution nodes.

In the second step, in order to achieve the final consistency of the data, control parameters P and Q are defined to filter the redundant contracted nodes and avoid the over-coverage of the hotspot region. Where P denotes the maximum number of task execution nodes required for a single work area and Q denotes a multiple of the task coverage target. To improve the task assignment success rate, P and Q should satisfy the following constraints:

$$\forall j \in n, \sum_{i \in m} x_{i,j} \le \mathrm{P}_j, \sum_{i \in m} x_{i,j} s_{i,j} \ge \mathrm{Q}gS \tag{10}$$

3. Blockchain privacy data security performance test results analysis

3.1. Analysis of the results of the privacy data on-chain and off-chain secure retrieval scheme

This section comprehensively evaluates the performance of our proposed scheme in various aspects such as storage efficiency, throughput and query time, objectively analyzes the effectiveness and feasibility of the scheme in this paper, and provides the necessary basis for deployment and adjustment in practical applications. In our designed scheme, by storing the keyword ciphertexts and the hash values of data file ciphertexts in the chain, while the actual data file ciphertexts are stored in the cloud, the limitations of storage capacity and read/write speed of blockchain are effectively solved.

3.1.1. Comparison of storage efficiency. This experiment evaluates the size of on-chain data in each scheme when the stored blockchain-supported privacy data transaction volume is different, based on the use of encrypted one-to-two lookup tables. The transaction volume is set to 2000, 4000, 6000, 8000, 10000, and 12000, respectively.The storage efficiency comparison results are shown in Fig. 3.The storage efficiency of the traditional blockchain storage scheme and the Diffie-Hellman scheme do not differ much, and the size of the data on the chain in all of them increases with the increase of the transaction volume. In contrast, this scheme possesses obvious advantages, especially when the transaction volume of blockchain-supported private data is large (12000), the on-chain data size stored by this paper's scheme (LPPOM) is approximately only 19.98% and 19.28% of the traditional blockchain scheme and the Diffie-Hellman blockchain scheme. The on-chain off-chain storage scheme

used in this paper does not need to store the original data on the blockchain, therefore, the on-chain data size grows slowly and has higher storage efficiency.

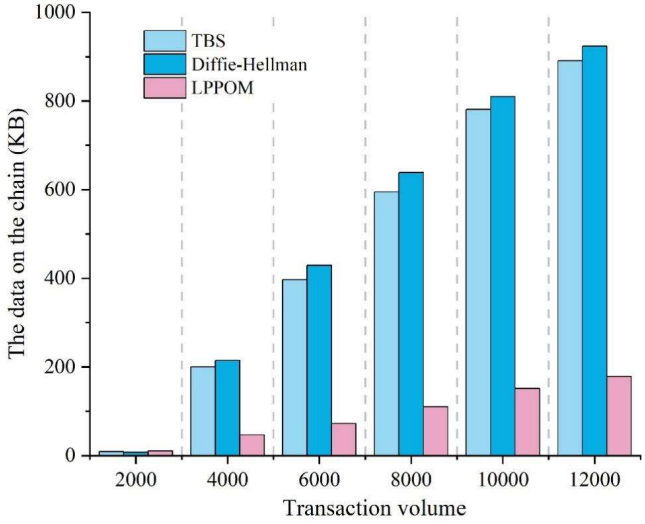
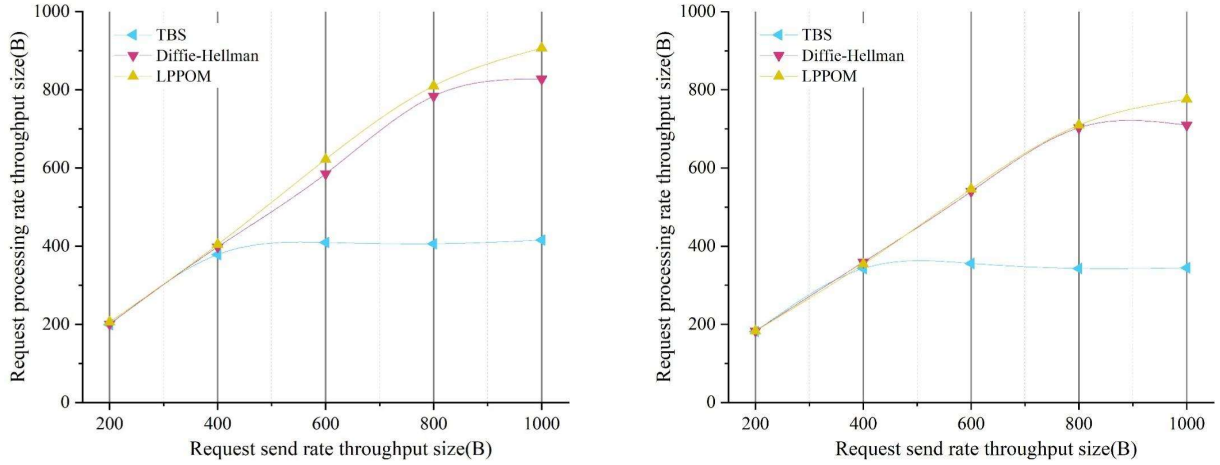
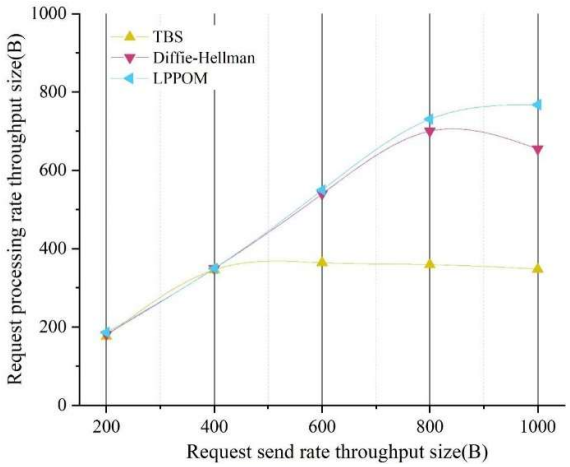


Fig. 3. Storage efficiency comparison results

3.1.2. Comparison of the throughput of the programs. Our proposed scheme achieves better results in terms of throughput when compared with existing methods. We sent requests at different rates and set the size of each request record data at 50B, 500B, and 1000B, respectively, and evaluated the speed of the system to process the request data.The TBS scheme stores all the raw data on the blockchain, and all participants use the blockchain as a shared database. The throughput of each scheme is shown in Fig. 4, (a) to (c) are the throughput of each scheme when the size of the file record data is 50B, 500B, and 1000B, respectively. When the request sending rate grows linearly, the request processing rate of TBS is limited by the blockchain full copy storage. When the request sending rate is less than 375, the request processing rate of the TBS scheme grows linearly and is not much different from the other schemes. However, as the request sending rate of the system increases, its efficiency begins to decrease, resulting in a large number of request messages not being processed in a timely manner. Experimental results show that under the same conditions, the Diffie-Hellman scheme and the scheme in this paper (LPPOM) achieve better throughput performance compared to the TBS scheme. However, our scheme also reduces the on-chain data storage requirements by utilizing an on-chain-off-chain secure storage architecture and ensures user privacy and security. Therefore, our scheme is more suitable for blockchain applications with massive blockchain-backed privacy data.



(a)Handling of all cases(50B) (b)Handling of all cases(500B)



(c)Handling of all cases(100B)

Fig. 4. Handling of all cases

3.1.3. Comparison of query times. Fig. 5 shows the results of time cost consumed in the retrieval process using traditional based scheme and Red Black Tree & Buffer respectively. Comparison of the time cost consumed in the retrieval process by both schemes reveals that the system retrieval time increases as the number of requested keywords increases. Comparison with the blockchain-supported privacy data blockchain traditional lookup for information retrieval scheme shows that the traditional lookup table mechanism prolongs the local query processing time, and the average query time corresponding to a single keyword is computed to be 0.3512 ms, while the average query time of the Red-Black-Tree & Buffer scheme is 0.1899 ms, which is a result of the fact that the traditional lookup table corresponds to multiple data items, which need to be individually computation of homomorphic encryption algorithm results. Meanwhile, another important reason for the high time cost of our scheme is that the multi-chain architecture proposed in this scheme can realize cross-chain data interoperability and sharing, which may also affect the retrieval efficiency to some extent. Experimental results show that our scheme is still within a reasonable range in terms of multi-chain based query speed. It is worth mentioning that our Red-Black-Tree & Buffer scheme enhances the security of the algorithm by a pseudo-random function with a 120-bit key, which realizes the privacy protection of the indexed data.

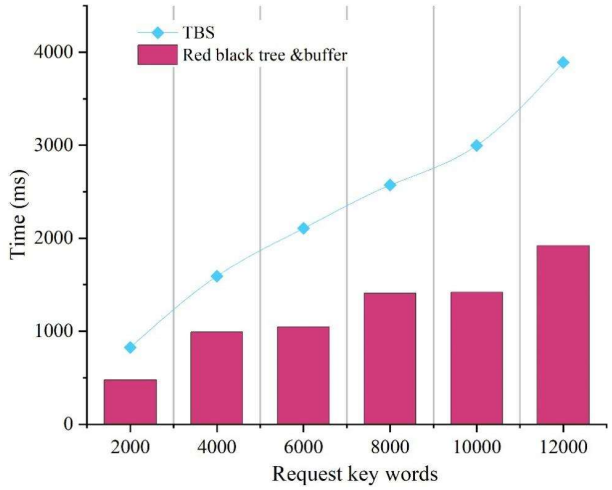


Fig. 5. The result of the cost of time in the retrieval process

3.1.4. System performance test results. Next, we designed experiments to test the scalability of the scheme in this paper. In order to evaluate the degree of impact of the proposed side-chain nodes on the system performance, we first conducted a performance test under a single-chain system, and then set the number of enterprise chains Nb to 0, 15, 30, and 60, respectively, and tested the query time of this paper's scheme under large-scale blockchain-supported privacy data scenarios. The query time test results of this paper's scheme under large-scale blockchain-supported private data scenarios are shown in Fig. 6, and the system response time is the shortest under the single-chain system, which takes only 0.6652 ms on average.This is because the main chain can initiate the retrieval directly to the cloud after receiving the user's retrieval request without the need of interacting with the enterprise private chain. With a different number of sidechain nodes, the system query time gradually increases as the main chain stores multiple types of blockchain-supported private data and needs to further request the corresponding enterprise chain to initiate retrieval to the cloud after receiving the retrieval request. In addition, each private chain needs to store, process and verify data independently, which likewise leads to an increase in storage and computational load. However, the average value of time cost consumed by the number of enterprise chains of 15, 30, and 60 is 0.8961ms, 1.0967ms, and 1.1507ms, respectively, which is only 0.2309-0.4855ms more than the single-chain system, and the trend of change is relatively stable. Therefore, this scheme has a certain degree of scalability.

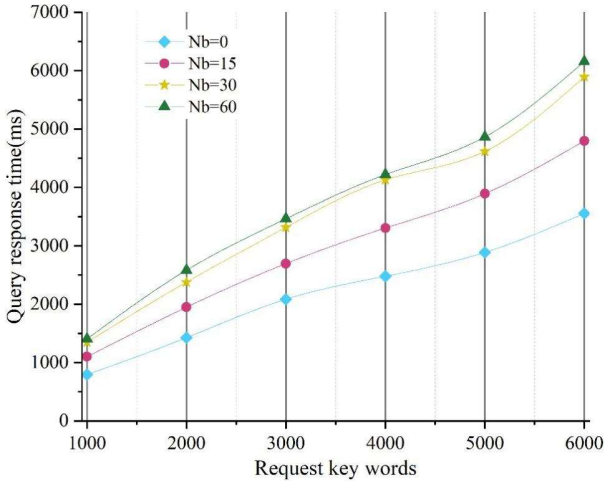
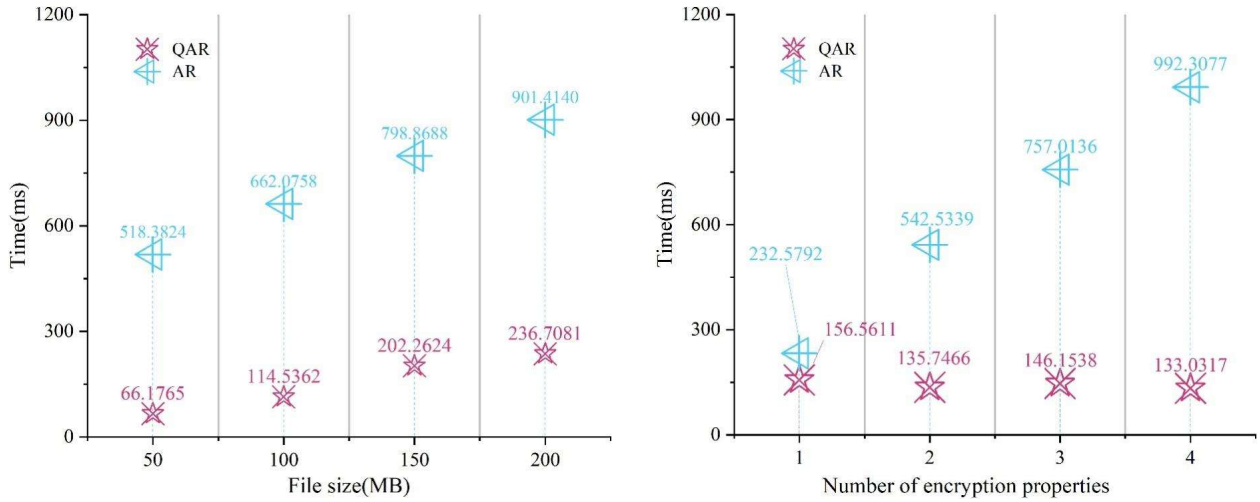


Fig. 6. The query time test results in large block chain support

3.2. Privacy data on-chain off-chain encryption performance test results

The data encryption function uses the AR algorithm designed in this paper that utilizes the anonymous region to hide the real location of the participating nodes and the QAR algorithm that queries the anonymous region to protect the user's private data, and it is the core functional module of the system, and the performance of the data encryption module has a direct impact on the performance of the user's performance during the depositing and uploading of certificates. The performance test of data encryption mainly includes two aspects, exploring the impact of the file size submitted by users and the complexity of the access control policy set by users on the encryption performance of the system, respectively. On this basis, the test of the data encryption function module is divided into two groups: one group keeps the access control policy unchanged (two sets of encryption attributes) and takes the file size as a variable to test the effect of file size on the data encryption performance; and the other group keeps the file size unchanged (file size 100MB) and uses a more complex access control policy to test the effect of the access control policy on the data encryption performance.

3.2.1. Data encryption performance testing. The data encryption performance test results are shown in Fig. 7, (a) and (b) represent the file data size and the number of access control policy data, respectively. From the experimental results, it can be seen that the data file size directly affects the performance of the AR algorithm module in the QAR algorithm for file encryption, while the complexity of the access control policy affects the performance of the QAR algorithm module. In common scenarios, i.e., the file is less than 200M and the number of encrypted attributes is less than 4, the total encryption time of QAR algorithm is between 66.1765-236.7081ms, which meets the user's demand for encryption of private data.



(a) The size of the file data

(b) Access control policy data number

Fig. 7. Data encryption performance test results

3.2.2. File Read and Write Performance Tests. This system uses the IPFS interstellar file system to store the electronic certificate of deposit files uploaded by users. In the electronic deposit system, the upload and download operations of the user's deposit involve a large number of file read and write operations. Therefore, the performance of reading and writing files stored in IPFS is an important performance bottleneck. In the following, we test the read and write speeds of files of different sizes in IPFS in LAN and public network environments respectively.

The IPFS data file read and write performance test results are shown in Figure 8. From the test results, we can see that when the file size is 100M, the IPFS read/write module can achieve a read/write speed of 0.1568~0.2639MB/ms under the public network conditions, which is basically up to the upper limit of the bandwidth of the test server, and can satisfy the needs of people's daily use. In the LAN environment with good network conditions, the file reading and writing speed will be greatly improved.

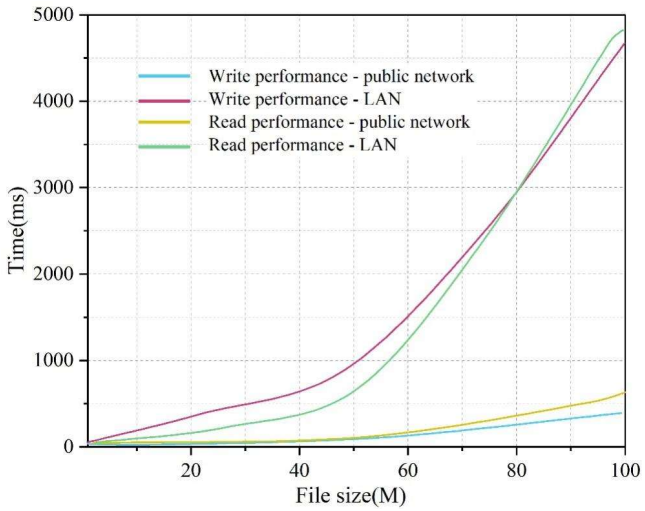


Fig. 8. IPFS data file reading and writing performance test results

3.2.3. Blockchain Read/Write Performance Testing. The read and write performance of the blockchain is also one of the bottlenecks constraining the performance of the system. Here the performance of data query and transaction submission is tested directly using the QAR-based blockchain access module in the system. The blockchain read and write performance test results are shown in Table 2. It can be seen that the average values of QAR federation chain data search throughput and transaction submission throughput in this system are 205 TPS and 127.5 TPS, respectively. And when the concurrent volume of requests is high, it will affect the successful submission of transactions to a certain extent. According to the analysis, the main performance bottleneck of the query is the server bandwidth, while the transaction submission performance has a relatively high demand on the server's CPU resources because it involves a large number of encryption and decryption calculations and hash calculations. If the server configuration is improved, the system performance will be more satisfactory.

Table 2. Block chain reading and writing performance test results

Type	Success	Failure	Send rate (TPSW)	Throughput (TPSW)
Query	1000	0	100	100
	1000	0	200	196
	1000	0	300	264
	1000	0	400	260
Transaction submission	1000	0	100	98
	988	12	200	146
	956	44	300	144
	864	136	400	122

4. Conclusion

In this article, the blockchain-supported interaction consistency enhancement technique for private data on-chain and off-chain is investigated, and an optimization algorithm for location privacy preservation in anonymous regions is proposed to investigate the secure retrieval and encryption performance of private data on-chain and off-chain. The following conclusions are drawn:

1) The scheme in this paper has higher storage efficiency when the transaction volume of privacy data supported by the blockchain is large (12000) and the data size on the LPPOM chain grows slowly. And the LPPOM scheme is to ensure the user's privacy security by reducing the storage requirements

of on-chain data, the more blockchain data, the better the application. Red Black Tree & Buffer consumes the least time cost in the retrieval process, and its average query time corresponding to a single keyword is only 0.1899ms.

2) Protecting data privacy and security by using coordination-aware technology to find the location of private data as a blockchain storage structure. This paper tests QAR's privacy data depository system from both functionality and performance aspects and finds that the functionality and performance performance of QAR's algorithmic module can meet the needs of practical application scenarios.

Funding

State Grid Shandong Electric Power Company science and technology project: Research on privacy computing technology under multi-level supervision based on blockchain (item status coding: 520604230007).

References

[1] Chen, Y., Avitabile, P., & Dodson, J. (2020). Data consistency assessment function (DCAF). Mechanical Systems and Signal Processing, 141, 106688.

[2] Shi, P., Cui, Y., Xu, K., Zhang, M., & Ding, L. (2019). Data consistency theory and case study for scientific big data. Information, 10(4), 137.

[3] Lindholm, A., Zachariah, D., Stoica, P., & Schön, T. B. (2019). Data consistency approach to model validation. IEEE Access, 7, 59788-59796.

[4] Gadde, H. (2023). AI-Based Data Consistency Models for Distributed Ledger Technologies. Revista de Inteligencia Artificial en Medicina, 14(1), 514-545.

[5] Ben Fekih, R., & Lahami, M. (2020). Application of blockchain technology in healthcare: a comprehensive study. In The Impact of Digital Technologies on Public Health in Developed and Developing Countries: 18th International Conference, ICOST 2020, Hammamet, Tunisia, June 24–26, 2020, Proceedings 18 (pp. 268-276). Springer International Publishing.

[6] Javaid, M., Haleem, A., Singh, R. P., Khan, S., & Suman, R. (2021). Blockchain technology applications for Industry 4.0: A literature-based review. Blockchain: Research and Applications, 2(4), 100027.

[7] Hashemi Joo, M., Nishikawa, Y., & Dandapani, K. (2020). Cryptocurrency, a successful application of blockchain technology. Managerial Finance, 46(6), 715-733.

[8] Kuznetsov, A., Oleshko, I., Tymchenko, V., Lisitsky, K., Rodinko, M., & Kolhatin, A. (2021). Performance analysis of cryptographic hash functions suitable for use in blockchain. International Journal of Computer Network & Information Security, 13(2), 1-15.

[9] Lashkari, B., & Musilek, P. (2021). A comprehensive review of blockchain consensus mechanisms. IEEE access, 9, 43620-43652.

[10] Khan, S. N., Loukil, F., Ghedira-Guegan, C., Benkhelifa, E., & Bani-Hani, A. (2021). Blockchain smart contracts: Applications, challenges, and future trends. Peer-to-peer Networking and Applications, 14, 2901-2925.

[11] Banafa, A. (2022). Blockchain technology and applications. River Publishers.

[12] Guo, L., Xie, H., & Li, Y. (2020). Data encryption based blockchain and privacy preserving mechanisms towards big data. Journal of Visual Communication and Image Representation, 70, 102741.

[13] Yang, Y., Wu, J., Long, C., Liang, W., & Lin, Y. B. (2022). Blockchain-enabled multiparty computation for privacy preserving and public audit in industrial IoT. IEEE Transactions on Industrial Informatics, 18(12), 9259-9267.

[14] Liang, X., Zhao, J., Shetty, S., & Li, D. (2017, October). Towards data assurance and resilience in IoT using blockchain. In MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM) (pp. 261-266). IEEE.

[15] Nie, Z., Long, Y., Zhang, S., & Lu, Y. (2022). A controllable privacy data transmission mechanism for internet of things system based on blockchain. International Journal of Distributed Sensor Networks, 18(3), 15501329221088450.

[16] Zheng, X., Zhu, Y., & Si, X. (2019). A survey on challenges and progresses in blockchain technologies: A performance and security perspective. Applied Sciences, 9(22), 4731.

[17] Fan, C., Ghaemi, S., Khazaei, H., & Musilek, P. (2020). Performance evaluation of blockchain systems: A systematic survey. IEEE Access, 8, 126927-126950.

[18] Ferrag, M. A., & Shu, L. (2021). The performance evaluation of blockchain-based security and privacy systems for the Internet of Things: A tutorial. IEEE Internet of Things Journal, 8(24), 17236-17260.

[19] Bamakan, S. M. H., Motavali, A., & Bondarti, A. B. (2020). A survey of blockchain consensus algorithms performance evaluation criteria. Expert Systems with Applications, 154, 113385.

[20] Zhang, R., Xue, R., & Liu, L. (2019). Security and privacy on blockchain. ACM Computing Surveys (CSUR), 52(3), 1-34.

[21] Rajendiran, K., Sridhar, A., & Cassinadane, A. V. (2022). Data Consistency, Transparency, and Privacy in Healthcare Systems Using Blockchain Technology. Blockchain Technology in Corporate Governance: Transforming Business and Industries, 125-141.

[22] Liang, W., Yang, Y., Yang, C., Hu, Y., Xie, S., Li, K. C., & Cao, J. (2022). PDPChain: A consortium blockchain-based privacy protection scheme for personal data. IEEE Transactions on Reliability, 72(2), 586-598.

[23] Liao, D., Li, H., Wang, W., Wang, X., Zhang, M., & Chen, X. (2021). Achieving IoT data security based blockchain. Peer-to-peer networking and applications, 1-14.

[24] Wei, P., Wang, D., Zhao, Y., Tyagi, S. K. S., & Kumar, N. (2020). Blockchain data-based cloud data integrity protection mechanism. Future Generation Computer Systems, 102, 902-911.

[25] Tian, H., Jian, Y., & Ge, X. (2022). Blockchain-based AMI framework for data security and privacy protection. Sustainable Energy, Grids and Networks, 32, 100807.

[26] Tortola, D., Felicioli, C., Canciani, A., & Severino, F. (2024, August). A Blockchain-Based Privacy-Preserving Auditable Data Structure Framework. In 2024 IEEE International Conference on Blockchain (Blockchain) (pp. 1-10). IEEE.

[27] Hao, K., Xin, J., Wang, Z., & Wang, G. (2020). Outsourced data integrity verification based on blockchain in untrusted environment. World Wide Web, 23, 2215-2238.

[28] Feng, T., Yang, P., Liu, C., Fang, J., & Ma, R. (2022). Blockchain Data Privacy Protection and Sharing Scheme Based on Zero-Knowledge Proof. Wireless Communications and Mobile Computing, 2022(1), 1040662.

[29] Piao, C., Hao, Y., Yan, J., & Jiang, X. (2021). Privacy preserving in blockchain-based government data sharing: A Service-On-Chain (SOC) approach. Information Processing & Management, 58(5), 102651.

[30] Wang, F., Zhou, J. T., Wang, H., & Guo, X. (2022, December). A blockchain-based multi-cloud storage data consistency verification scheme. In 2022 IEEE Intl Conf on Parallel & Distributed Processing with

Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom) (pp. 371-377). IEEE.

[31] Rathore Heena,Samant Abhay & Jadliwala Murtuza. (2021). TangleCV: A Distributed Ledger Technique for Secure Message Sharing in Connected Vehicles. ACM TRANSACTIONS ON CYBER-PHYSICAL SYSTEMS(1),

[32] Haoran Shi,Zehua Chen,Yongqiang Cheng,Xiaofeng Liu & Qianqian Wang. (2024). PB-Raft: A Byzantine fault tolerance consensus algorithm based on weighted PageRank and BLS threshold signature. Peer-to-Peer Networking and Applications(1),1-16.

[33] Ehtisham Ul Haque,Waseem Abbasi,Ahmad Almogren,Jaeyoung Choi,Ayman Altameem,Ateeq Ur Rehman & Habib Hamam. (2024). Performance enhancement in blockchain based IoT data sharing using lightweight consensus algorithm. Scientific Reports(1),26561-26561.

[34] Lei Wang,Hao Cheng,Zihao Sun,Aolin Tian & Zhonglian Yang. (2025). PSPL: A Ponzi scheme smart contracts detection approach via compressed sensing oversampling-based peephole LSTM. Future Generation Computer Systems107655-107655.

[35] Chaudhry Natalia & Yousaf Muhammad Murtaza. (2023). A hash-based index for processing frequent updates and continuous location-based range queries. Knowledge and Information Systems(10),4233-4271.

[36] Jiaxi Liu,Bolin Gao,Wei Zhong,Yanbo Lu & Shuo Han. (2024). Adaptive optimization strategy and evaluation of vehicle-road collaborative perception algorithm in real-time settings. Computers and Electrical Engineering(PC),109785-109785.