ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 2 - 2026

Research on distribution network heterogeneous data evolution model based on blockchain optimization algorithm

Yipeng Liu[1,*], *Wei Dong*[1], *Wenting Wang*[1], *Jun Cao*[1] *and Yuewei Tian*[1]

[1] Guiyang Power Supply Bureau of Guizhou Power Grid Co., Ltd., Guiyang, Guizhou, 550001, China

SUMMARY: *Based on the introduction of blockchain technology, the article proposes Byzantine fault-tolerant consensus algorithm (DT-PBFT) based on dynamic credit mechanism, adopts credit score mechanism to protect the rights and interests of good nodes while penalizing evil nodes, and introduces a flexible dynamic adding and deleting mechanism to screen and eliminate the evil nodes. Subsequently, an on-chain and off-chain data collaborative algorithm model for blockchain traceability is proposed, which combines the master-slave replication principle to construct an on-chain log collaborative synchronization mechanism for off-chain data, takes the operation logs stored on the chain as the instruction to drive the update of off-chain traceability data, and combines with cryptographic algorithms to design the calculation method of off-chain data state, so as to achieve the collaborative feedback and sharing security between heterogeneous on-chain operation data and off-chain traceability data. Through the comparison experiments of consensus algorithms, it is proved that the DT-PBFT algorithm proposed in this paper is more flexible and efficient in the actual application process, and reduces the consumption of resources and improves energy saving at the same time. In the simulation simulation experiment of the cooperative mechanism, in the insertion time comparison, the overall efficiency of this paper's method has been greatly improved, and there is an improvement of about 155ms under 20,000 pieces of data.*

KEYWORDS: *blockchain; consensus algorithm; DT-PBFT; data collaboration; algorithmic modeling*

1 Introduction

Medium voltage distribution network refers to the agricultural distribution network with voltage of 6 kV-20 kV [1]. Distribution network in the actual operation process will produce a variety of massive data, these heterogeneous data with different sources, different structures, different time scales, spatial scales are also different, but also has a large amount of data, high growth efficiency, multiple types, high value, etc., and these characteristics lead to the heterogeneous data within the distribution network algorithms, analysis and application of more difficult [2, 3]. Therefore, focusing on the technical bottlenecks and optimization paths in heterogeneous data processing has important theoretical value and practical significance for enhancing the level of power network intelligence and system resilience.

The structure of data acquired within different information systems also varies, which contains both structured and semi-structured data [4]. Therefore, before analyzing the multi-source heterogeneous data in distribution networks, it is necessary to collect the multi-source

*liuyipwng060@163.com
https://doi.org/10.65102/is2026698
Published: 30 April, 2026

heterogeneous data in distribution networks. Regarding the storage of multi-source heterogeneous data, Zhou et al. proposed a novel data distribution algorithm (SUORA) designed for heterogeneous storage devices to maximize the performance and lifetime advantages of hybrid storage systems by dividing heterogeneous devices into different buckets and segments, and using a pseudo-random function to map the data distribution under the premise of balancing capacity, performance and lifetime [5]. Wu et al. proposed a general distribution network framework, TENON, to solve the problems of unclear mechanism of label information propagation across domains and lack of theoretical explanation of neural network convergence in existing domain adaptation and extra-distributed generalization algorithms [6]. For the problem of dimensional catastrophe and missing values in large-scale high-dimensional heterogeneous data, Gahar et al. proposed a new method of distributed statistical dimensionality reduction based on MapReduce paradigm. The method combines the random forest interpolation technique to deal with missing values, aiming to extract valid information and reduce the search space to optimize the data exploration process [7].

Aiming at the problem that heterogeneous data resources are difficult to be effectively utilized in intelligent power distribution system, Tan et al. proposed a new method of heterogeneous data fusion based on generative adversarial network (GAN). The method extends the complete data sample distribution by introducing the GAN theory and realizes the finite open coverage of the sample space by using the proposed peak clustering algorithm, and then repairs the incomplete samples in order to eliminate the heterogeneous features; finally, the repaired samples are calibrated with the help of the trained GAN discriminator model, and the effective integration of the heterogeneous data of the smart distribution is realized [8]. Wang et al. systematically reviewed the techniques for processing and evaluating multi-source heterogeneous data, and based on the characteristics of diverse data sources, heterogeneous structure, and complex associations, they proposed a number of methods applicable to the processing of multi-source heterogeneous sensor data in power distribution networks [9]. Li et al. designed a fusion algorithm based on improved Kalman filter for the problem of high complexity and large deviation of multi-source heterogeneous data fusion in distribution networks, which adjusts the initial data deviation through the correction process, combines the least squares method to align the timing, Lagrange interpolation to fill in the missing values, and fuses the distribution map method with the Kalman filter to enhance the algorithm performance [10]. Wu and Hu proposed a joint substation safety control system and model analysis scheme based on multi-source heterogeneous data fusion, and innovatively introduced the Attention-LSTM prediction model to realize short-term prediction and abnormal state early warning for spatio-temporal data of power equipment [11].

In addition, heterogeneous data play an important role in the process of distribution network security operation, monitoring and O&M, but the existing data management mechanisms often face problems such as data silos, information asymmetry and security risks [12]. Blockchain technology provides a new solution for these heterogeneous data management with its decentralization, non-tampering and transparency properties [13]. Aiming at the challenges in heterogeneous data management, Tseng et al. first pointed out the limitations of traditional IoT systems and the difficulties in the integration of IoT and blockchain, and then proposed a management architecture for large-scale heterogeneous data based on blockchain optimization [14]. Shen et al. use blockchain technology to build a secure and reliable data sharing platform that enables multiple data providers to collaboratively train models on encrypted data without having to centrally send raw data to potentially unreliable centralized servers [15]. In addition, blockchain, as a transparent and secure infrastructure, adopts block-chain data structure to verify and store heterogeneous data information from different sources and different structures in the embedded power grid, and utilizes the consensus mechanism to ensure the consistency

of the data information, and its tamper-evident performance effectively improves the security of storing and sharing the data information [16].

The article first introduces the classification and principles of blockchain technology. Subsequently, it proposes the Practical Byzantine Fault-Tolerant Consensus Algorithm (DT-PBFT) based on the credit score mechanism-dynamic addition and removal of nodes. The algorithm introduces a dynamic join/withdrawal mechanism so that nodes in the cluster can join/withdraw freely on demand, and introduces a credit point system, which selects the optimal master node by dividing the nodes into an alternate master node layer, an intermediate layer, a warning layer, and a cleanup layer by the layering mechanism in accordance with the degree of trustworthiness. At the same time, the Byzantine nodes in the network cleanup layer are eliminated. And the consensus process is optimized by optimizing the consistency protocol based on the two mechanisms. Then, an on-chain and off-chain data collaboration mechanism for blockchain traceability is proposed, which combines the principle of master-slave replication, divides data into operation data and traceability data, designs the underlying data structure and data collaboration mechanism, and realizes the replication collaboration of on-chain operation data to update off-chain traceability data. We also propose an under-chain integrity protection calculation method to ensure the data integrity during data synergy, and design the query method for the under-chain traceability data in synergy with the on-chain operational data to provide users with complete and detailed traceability data of the whole life cycle. In the experiments, DT-PBFT algorithm, PBFT algorithm and CPBFT algorithm are experimented and compared in terms of consensus latency, throughput and security, so as to verify the feasibility of DT-PBFT algorithm. Finally, the performance test of the on-chain-off-chain data collaboration method for blockchain traceability proposed in this paper is conducted.

2 Blockchain algorithm-based heterogeneous data evolution model for distribution networks

2.1 Blockchain technology

2.1.1 Blockchain classification

There are three types of blockchain, namely, coalition chain, private chain and public chain. Different types of blockchain correspond to different usage scenarios. Enterprises of different sizes choose different types of blockchain. Public chain can be completely decentralized, but the transaction speed is slow. Although the transaction speed is fast, but it can't be completely decentralized. Private chain has fewer nodes and faster transaction speed, but it is less centralized. Therefore, when we choose blockchain, we need to choose the type of blockchain according to different application scenarios and actual needs.

2.1.2 Blockchain fundamentals

Blockchain is a shared distributed ledger, just like we all have a ledger in our hands, and the content recorded in the ledger is consistent. A block in the blockchain mainly contains two parts: the block header and the block body. The block header is mainly used to store the hash value and timestamp data, the block body records the data generated by the transaction, the hash value is combined to produce the Merkle root. Each new block contains the hash value of the previous block, so if someone tampered with the previous block, his hash value will be different from the hash value in the next block. It is like a big chain table, which is the principle of blockchain tampering. The block structure is shown in Figure 1.

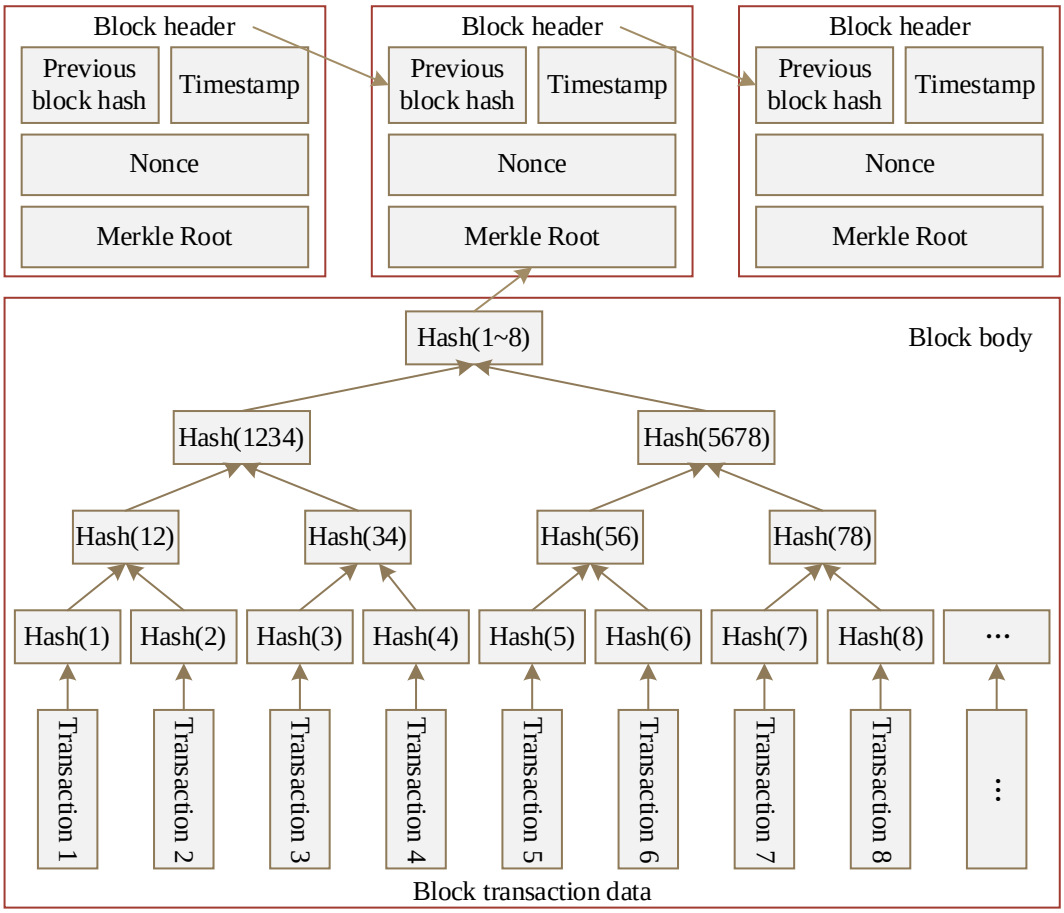

Figure 1: Block structure

2.2 Byzantine fault-tolerant consensus algorithm based on dynamic credit mechanism

2.2.1 Dynamic mechanisms

The dynamic mechanism contains nodes dynamically entering the consensus network and nodes dynamically exiting the consensus network, and the entire blockchain network does not need to be dynamically re-enabled in the process of nodes dynamically entering and exiting, which effectively enhances the flexibility of the consensus algorithm.

(1) Dynamic node addition

Dynamic addition of nodes is shown in Figure 2 in the following steps:

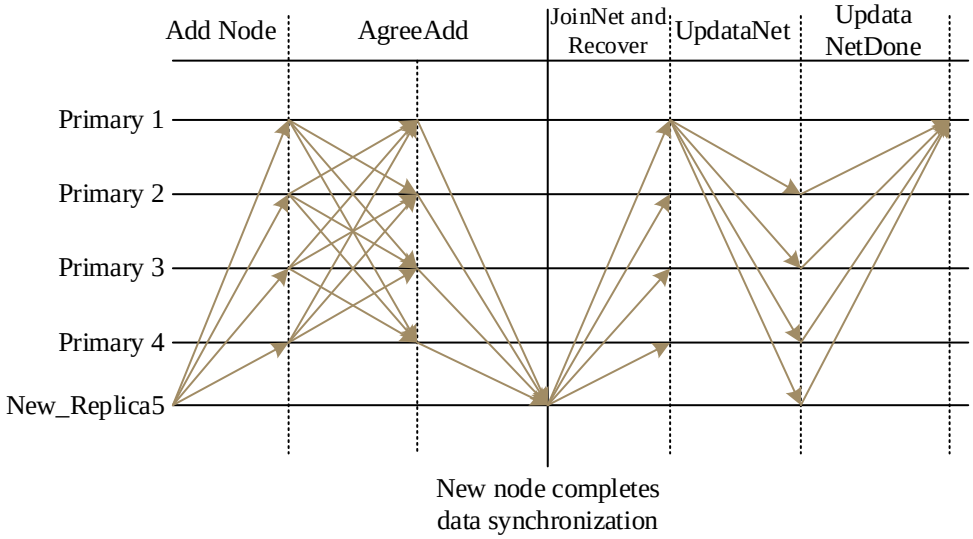

Figure 2: Dynamically add nodes

(a) Application phase: when a new node existing in the network is activated, if the node wants to join the cluster to participate in the subsequent consensus phase, it should first send a message S requesting to join the cluster to all the nodes in the cluster with a timestamp, and initiate an AddNode request connection.

(b) Authentication of new nodes phase: when all existing nodes receive an AddNode request from a new node, they broadcast and collect AgreeAdd messages from other nodes. When a node collects $2f+1$ AgreeAdd messages, it sends an authentication message to the added node agreeing to join the cluster. When the added node receives $2f+1$ authentication messages, the request is agreed and the added node is allowed to join the cluster.

(c) Data synchronization phase: the new node starts to enter the active recovery process. The new node sends a request for data synchronization and broadcasts it to the other nodes, while the other nodes send all the currently stored information to the new node to achieve data synchronization of the new node.

(d) JoinNet phase: after data synchronization, broadcast the JoinNet request to all nodes in the whole blockchain network, requesting to participate in the consensus of the network. When all existing nodes receive the request to notify all existing nodes that the new node starts to enter the network, at this time the number of nodes in the network and the new view v is recalculated.

(e) Update the network: the master node releases the UpdataNet information to all nodes in the cluster, after all consensus nodes receive it, update the total number of nodes in the block cluster N and the view v, and complete the process of adding new nodes.

(f) Reply phase: feedback to the master node after completing the update of the view and the total number of nodes, when the master node receives $2f+1$ information that the network has completed the incorporation of new nodes, then complete a dynamic increase in nodes of the consensus behavior.

(2) Node Dynamic Exit

Dynamic deletion of nodes is shown in Figure 3, which is divided into four steps: application phase, authentication message phase, network withdrawal phase and updating network:

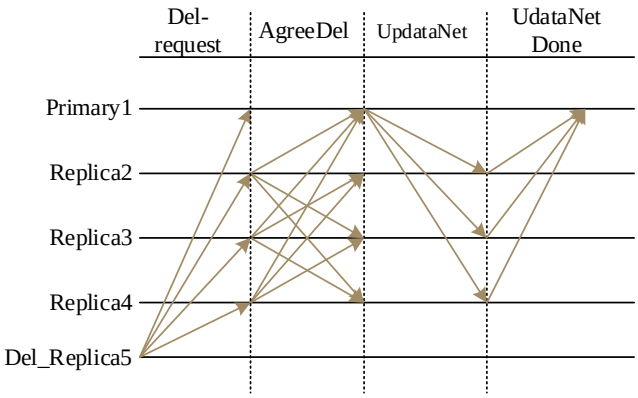

Figure 3: Dynamically delete nodes

2.2.2 Credit score mechanism

In order to improve the mechanism of PBFT, the nodes in the cluster randomly select the master node and introduce the credit score mechanism. In the Byzantine fault-tolerant algorithm, the block generation verification is mainly done by the master node, and DT-PBFT, in order to improve the original algorithm and make the nodes better monitored, introduces the credit score mechanism, which is mainly to stratify all the nodes, and the nodes are divided into four layers: alternative master node layer, intermediate layer, warning layer, and clean-up layer. In the layering model, the score value is set as n and the upper limit of the score value is set as 100, and the nodes are divided into alternative master node layer (80n<100), intermediate layer (40n<80), warning layer (20n<40), and cleanup layer (n<20) depending on the different scores to differentiate between the intervals.

Malicious nodes are removed from the network as shown in Fig. 4. When there is a malicious node kicked out of the existing blockchain network, firstly the view v,total number of nodes after removing the malicious node is calculated N. Secondly, the master node broadcasts a Del-message message to the other nodes, and the other nodes receive the Del-message message, and then they broadcast their own agreement to remove the Del_Replica to the other nodes in the blockchain network.5 Then, if f AgreeDel message, then all nodes agree and delete the data synchronization request initiated by the node and encapsulate the view v, total number of nodes N and other messages after deleting the node. Finally after the Del-Node5 node exits, the master node sends the UpdataNet message, and all the nodes in the whole network receive the UpdataNet message and update the total number of nodes N and view v in the blockchain network to complete the process of deleting the node.

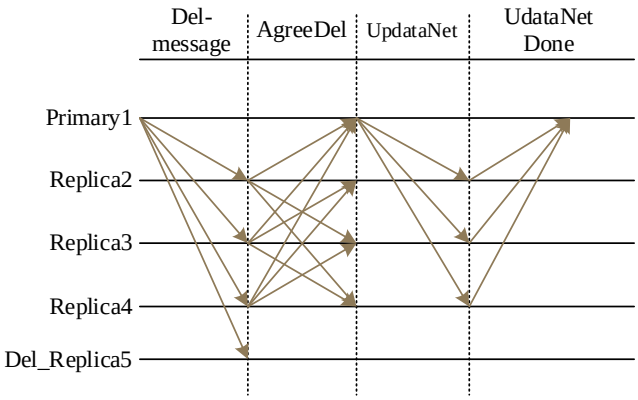

Figure 4: Malicious nodes are removed from the network

2.2.3 Consistency protocol optimization

The DT-PBFT algorithm proposed in this paper employs an optimized consistency protocol based on increasing credit score screening and adopts a dynamic mechanism to ensure the reliability of the overall cluster in the entire network of nodes. Among them, the increasing credit score mechanism screens the nodes with higher creditworthiness as alternative master nodes thereby enhancing the trustworthiness of the master nodes. The dynamic mechanism reduces the probability of problematic nodes to be selected as master nodes, and reduces the interactive information of one round of network-wide consensus verification in the consensus phase, so that the complexity of the algorithm is reduced to half of the original algorithm. The flowchart of the consistency protocol is shown in Fig. 5.

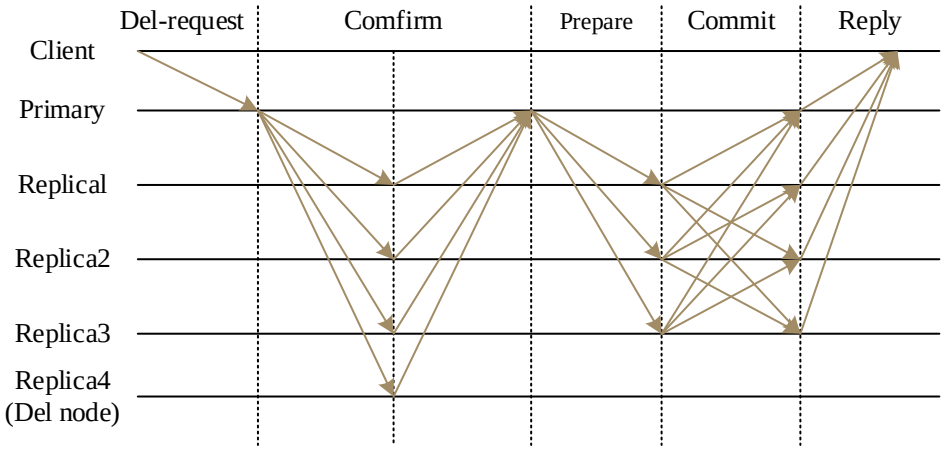

Figure 5: Consistency Protocol Flow Chart

The optimized consistency protocol execution process is mainly divided into the following five steps:

(1) Request phase: the client C sends a request message m through the master node P.

(2) Confirmation phase: the confirmation phase will be divided into two parts, the first half by the master node P to all the consensus nodes in the network to send the message number $t1$, plus the timestamp for the full sorting of the allocation of the number of q, and then the master node will be issued to all the nodes in the network to confirm the phase of the message $<confirm, v, N, t1>$, in which v on behalf of the view of the number of the category. The second half of the message is the return of the confirmation message to the master node, and the nodes participating in the consensus return the confirmation execution message $t2$ to the master node P within the time delay allowed. If a node sends a request to exit the network at this point $t3$ it can jump to the view of the dynamic exit network, until the node completes the exit network phase by the master node to integrate the current number of nodes N and update the view given the number h.

(3) Preparation phase: The master node integrates the request message and the updated number of nodes N in the acknowledgement phase and releases a message to the network as m, plus a timestamp to assign a number k in full order, and then the master node releases a preparation message $<< Prepare, h, N, d > m >$ to all the nodes in the network, where d denotes a summary part for the message m.

(4) Acknowledgement phase: If a node is in the acknowledgement phase, each node needs to send out a message to confirm that it has entered the acknowledgement segment. Each node needs to verify the authenticity of the message after receiving the confirmation packet from other nodes, checking the correctness of the signature, whether the view number is consistent

with the current view number h, and whether the serial number of the message meets the requirements of the waterline, which serves to prevent faulty nodes from consuming sequence space by using large serial numbers. The acknowledgement phase is completed when node i has checked that the $2f+1$ acknowledgement messages including itself are consistent with the pre-prepared messages. If there are more than or equal to $f+1$ nodes failing to verify the message, it is necessary to replace the master node and halve the credit score of the master node to complete the downgrading, and then select the master node from the updated alternative master nodes and start again.

(5) Feedback phase: after the node i reaches the determination phase, it gives feedback to the client C and returns the message $<REPLY,h,c,i,r>$, where r is the final execution result of the request. If the client C obtains $f+1$ feedback messages, it is recognized that the outgoing request has been fulfilled and successfully agreed upon. If the feedback information is less than $f+1$, the consensus request fails then the Byzantine nodes need to be penalized according to the credit score mechanism, and the nodes that need to be removed out of the network will complete the culling before generating a new view waiting for the client to re-send the request.

2.3 On-chain and off-chain data collaboration mechanism for blockchain traceability

2.3.1 System architecture overview

The on-chain and off-chain parts of the blockchain have different characteristics respectively. The on-chain part can ensure the security of on-chain data by utilizing the tamper-proof characteristics of the blockchain, and the off-chain part can provide efficient data query while improving scalability. The on-chain and off-chain data collaboration mechanism collaborates and synchronizes the on-chain and off-chain, which can give full play to the advantages of the on-chain and off-chain layers and make the flow of data in the blockchain more secure and efficient. The architecture of the on-chain and off-chain data collaboration system is shown in Figure 6.

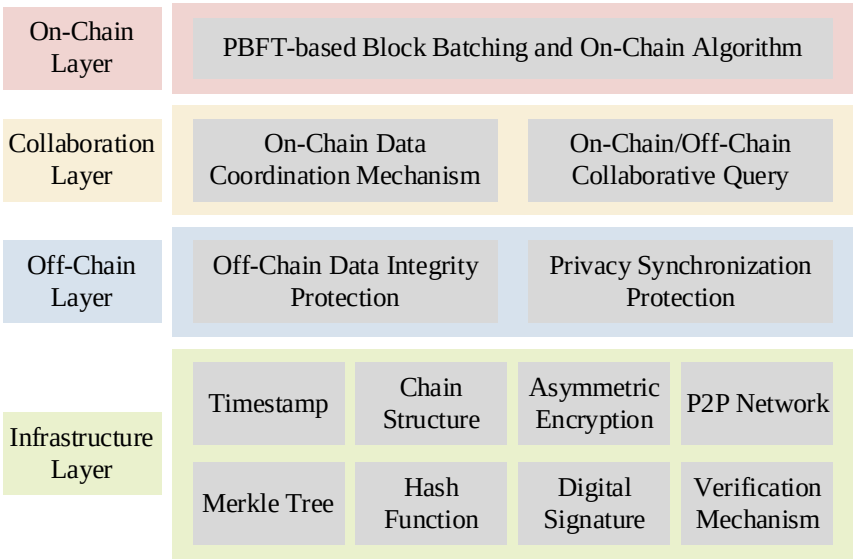

Figure 6: Architecture of on-chain and off-chain data collaboration system

2.3.2 On-chain Data Collaboration Mechanisms

The purpose of data synergy is to synchronize data in different locations or of different types to maximize the advantages of different data. The different characteristics of on-chain and off-chain data determine the way of data collaboration, on-chain data has high security but limited storage resources, off-chain data has large storage space but security is difficult to ensure. Combining the characteristics of the two, the on-chain data as the core of the on-chain and off-chain synergy, through the implementation of the on-chain security data to ensure the consistency of the off-chain data. The business operation logs, which are written more and read less, are stored on the chain, and the new block data structure is designed by combining the log attributes with the traditional blockchain data structure, based on the principle of master-slave replication, to realize the collaborative mechanism of updating the information of product flow by executing the business operation logs.

2.3.3 Off-chain data integrity protection

As the main data source of product traceability in this study, the integrity and security of off-chain data is an important prerequisite for the authenticity and effectiveness of traceability data. In the architecture of this study that separates on-chain data from off-chain data, the characteristics of the blockchain chain structure can ensure that the data on the blockchain are difficult to be tampered with. Although all the nodes update the off-chain traceability data by executing consistent and safe on-chain operations, there is still a possibility that the nodes may tamper with the local traceability data through the database operations privately, which threatens the security and integrity of the off-chain traceability data. For this reason, we combine the secure hash algorithm1 with a sensitive computational method for relational data to ensure the security and integrity of the data in the under-chain part. This section firstly analyzes the characteristics of the under-chain data, and then introduces the characteristics and computational principles of Secure Hash Algorithm 1. Finally, it describes how to combine the characteristics of off-chain data and the principle of safe hashing algorithm 1 to design an off-chain data state calculation algorithm to protect the integrity and security of off-chain data.

2.3.4 Collaborative on-chain and off-chain queries

The traceability information of the product is the record of the product throughout the blockchain process. In the blockchain scenario, the most frequently queried off-chain traceability information is mainly for consumers, and this part of the data is stored in the node's local off-chain database, so it is not necessary to access the blockchain to obtain the data, and it has a high query efficiency. For the participants and regulators in the blockchain, under certain circumstances, they not only need to obtain the product parameters and flow information of each link in the off-chain database, but also need to combine with the operation behavior of each node stored in the chain to help the participants and regulators to quickly investigate the problematic links of the accidents, locate the operator, time, location and other information involved in the problematic links, and accurately carry out the recall actions.

2.3.5 Synchronized on-chain and off-chain privacy protection

The security issue in data sharing has always been a key obstacle for blockchain to further break down the information silos. In a business environment full of competition and intermingling of reality and reality, blockchain enterprises, on the one hand, want to understand the situation of the whole industry in order to better adjust their own strategies, and on the other hand, they do not want to disclose their own operation to avoid competitors' understanding of their own situation. Therefore, if this kind of commercially sensitive private data is not properly protected

and uploaded onto the chain inadvertently, it will bring significant privacy risks, and may lead to various commercial risks, resulting in actual economic losses. Therefore, a privacy synchronization protection scheme combined with RSA is proposed under the mechanism of on-chain and off-chain data collaboration to ensure the security of sharing sensitive private data.

2.3.6 On-chain and Off-chain Data Collaboration Processes

First of all, the blockchain node initiates the transaction according to the business requirements, and the sensitive private data in the transaction is encrypted with point-to-point double asymmetric encryption by selecting the public key of the authorized node through the privacy synchronization protection module, and the encrypted cipher text is embedded into the transaction and directly executed to the local off-chain database. At the same time, the executed transaction is written to the local logging system, and the transaction in the logging system analyzes the four basic attributes of the SQL statement through the on-chain data synergy mechanism and attaches the auditable attributes and security attributes. The off-chain data integrity protection module calculates the current off-chain data status, writes the hash value generated by the calculation into the block header, constructs the block and pushes it into the block queue to be uploaded. After the completion of the last round of consensus, the header block in the block queue to be uploaded is out of the queue, and the attribute is updated to take the block with the latest consensus as the parent block of the current out-of-queue block, and consensus is carried out on the block through the block batch uploading algorithm with the common block screening strategy and the block sorting linking strategy.

Secondly, after the block consensus is completed, the block header and the log structure in the block body are verified to verify whether the security attributes such as the data state hash value under the local chain are compliant, and to obtain the permissions involved in the log structure in the dynamic permissions table to verify whether the node to which the log belongs to has the operation permissions. After passing the verification, the log structure is constructed into SQL statement one by one and executed to the local database under the chain. At the same time, through the on-chain and off-chain collaborative query module to obtain the traceability code in the log structure, the serial number of the log and the block hash value of the block where it is located, and construct it into a new SQL statement written into the index relationship table to construct the index relationship.

Finally, when product traceability is carried out in the blockchain node, the off-chain data can be accessed directly to obtain the flow data of the product in each link of the blockchain. At the same time, it can obtain the flow data of the target product and the operation records in each link through the on-chain and off-chain collaborative query interface to track the flow path of the product and locate the source of the product problem.

3 Experiments and analysis of results

3.1 DT-PBFT Algorithm Performance Test Experiments

3.1.1 Consensus delay comparison

Firstly, the consensus latency of DT-PBFT algorithm, PBFT algorithm and CPBFT algorithm is tested for the number of nodes 4, 7, 10, 13, 19, 25, 33, 45 under the setting that the data block is 0.5M size. The consensus latency of DT-PBFT algorithm is compared with PBFT algorithm and CPBFT algorithm by testing the consensus latency for the number of nodes in the whole network as 4, 7, 10, 13, 19, 25, 33, 45 with the setting of 1M and 2M size of data blocks. The average delay data and analysis of the three algorithms at 0.5M, 1M and 2M block sizes are

shown in Table 1. As can be seen from the table, in the case of 0.5M data block size, the consensus latency of the DT-PBFT algorithm is reduced by 21.1% and 16.95% compared with the PBFT algorithm and CPBFT algorithm, respectively. In the case of 1M data block size, the consensus delay of DT-PBFT algorithm is reduced by 18.08% and 6.88% compared with PBFT and CPBFT algorithms respectively. In the case of 2M data block size, the consensus delay of DT-PBFT algorithm is reduced by 14.74% and 6.2% compared with PBFT algorithm and CPBFT algorithm respectively. To summarize, when the data block size is 1M, the reduction of consensus delay of DT-PBFT algorithm is larger than 0.5M and 2M, and it has lower consensus delay.

Table 1: Average delay comparison

		4	7	10	13	19	25	33	45	Mean node delay	Reduce/PBFT	Reduce/CPBFT
0.5M	PBFT	110	103	110	117	118	129	146	148	122.625	-	-
	CPBFT	86	120	104	113	125	122	109	153	116.5	4.99	-
	DT-PBFT	76	81	88	75	87	108	129	130	96.75	21.1	16.95
1M	PBFT	114	135	144	148	155	153	178	212	154.875	-	-
	CPBFT	104	115	119	138	126	139	167	182	136.25	12.03	-
	DT-PBFT	114	99	121	110	130	132	148	161	126.875	18.08	6.88
2M	PBFT	135	118	168	163	170	190	203	237	173	-	-
	CPBFT	112	130	135	158	154	178	176	215	157.25	9.10	-
	DT-PBFT	123	127	129	131	144	147	180	199	147.5	14.74	6.2

3.1.2 Throughput comparison

The variation of system throughput is also influenced by the number of consensus nodes and data block size. In order to test more in line with the real system throughput, this experiment is set to test under different consensus node numbers and data block sizes. The system throughput of 4, 7, 10, 13, 19, 25, 33 and 45 nodes in the whole network is tested under different data block sizes (0.5M, 1M and 2M).

The average throughput data and analysis of the three algorithms at 0.5M, 1M and 2M block sizes are shown in Table 2. As can be seen from the table, in the case of 0.5M data block size, the throughput of DT-PBFT algorithm is improved by 25.81% and 11.58% as compared to PBFT algorithm and CPBFT algorithm. In the case of 1M data block size, the throughput of DT-PBFT algorithm is improved by 21.5% and 11.45% compared to PBFT and CPBFT algorithms. In the case of 2M data block size,the throughput of DT-PBFT algorithm is improved by 17.28% and 7.87% compared to PBFT algorithm and CPBFT algorithm. To summarize, the throughput of DT-PBFT algorithm is higher than that of block size 0.5M and 2M when the block size is 1M, which is consistent with the analysis of consensus delay.

Table 2: Average throughput comparison

		4	7	10	13	19	25	33	45	Mean TPS	Reduce/PBFT	Reduce/CPBFT
0.5M	PBFT	305	277	268	257	241	237	192	191	246	-	-
	CPBFT	351	337	295	290	255	252	224	215	277.375	12.75	-
	DT-PBFT	388	368	343	307	279	277	267	247	309.5	25.81	11.58
1M	PBFT	553	488	391	345	304	275	255	208	352.375	-	-
	CPBFT	583	514	433	354	361	312	296	220	384.125	9.01	-
	DT-PBFT	636	540	462	440	402	361	325	259	428.125	21.5	11.45
2M	PBFT	806	698	637	584	566	499	433	407	578.75	-	-
	CPBFT	877	782	692	618	609	552	461	443	629.25	8.73	-
	DT-PBFT	939	821	774	690	648	583	514	461	678.75	17.28	7.87

3.1.3 Security comparisons

In general, the node can normally carry out the consensus of the message without downtime as well as maliciously breaking the consensus rules, etc., the trust degree of the node will usually be at a high level. If a node is occasionally down, but is not detected by the system as a node with malicious behavior, the trust of the node will decrease according to the corresponding rules, and as long as it participates normally in the consensus process afterwards, the trust of the node will gradually increase. If the system detects the malicious behavior of the node during the consensus process, then the node is severely punished according to the dynamic trustworthiness rules. The ratio of malicious behavior is shown in Figure 7. As can be seen from the figure, after 20 rounds of consensus, the proportion of malicious behavior in the consensus nodes of DT-PBFT algorithm is greatly reduced, about 2%. Although CPBFT has a tendency to decrease, but due to its own margin mechanism, it leads to maintain the proportion of malicious behavior at 7%, while the proportion of malicious behavior in PBFT has been 20%. This proves that the dynamic trust degree model of DT-PBFT algorithm can effectively limit the malicious nodes to participate in the system consensus.

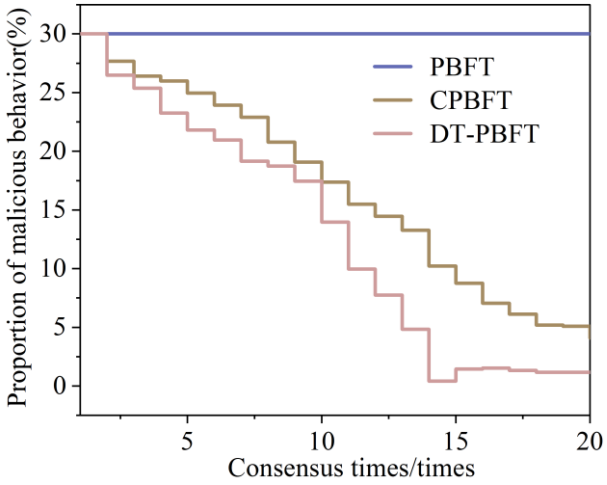

Figure 7: Proportion of malicious behavior

Meanwhile, this section compares and analyzes PBFT algorithm,CPBFT algorithm and DT-PBFT algorithm.The comparison between different algorithms is shown in Table 3. As can be seen from the table, DT-PBFT algorithm not only has a greater advantage in security, and has a higher consensus efficiency, which is especially suitable for blockchain.

Table 3: Comparison among different algorithms

Consensus algorithm	Consensus efficiency	Resource energy consumption	Degree of centralization	controllability	Safety level	There is no punishment mechanism	Do you need a token	Application scenarios
PBFT	Low	High	Decentralization	Stronger	General	No	No	Alliance Chain
CPBFT	Higher	Low	Weak centralization	Stronger	high	Yes	Yes	Alliance Chain
DT-PBFT	High	Low	Weak centralization	strong	Extremely high	Yes	No	Alliance Chain

3.2 Simulation of distribution network heterogeneous data algorithm simulation

3.2.1 Blockchain System Throughput and Latency

In this section, 20,000 pieces of data are processed and stored in trusted nodes. A third-party tool, Caliper, is used to evaluate the efficiency of the three operations of accessing nodes, querying node information, and interacting and communicating among nodes in the overall blockchain network. The transmission rate and throughput comparisons of the two methods for access (left), query (center), and node interaction (right) are shown in Table 4. The latency comparison of access, query and node interaction is shown in Table 5. In this section, 6000 repetitive experiments are conducted on all 20000 data in the form of accesses, queries and interactions to verify some performance metrics of the blockchain system. The transfer rate of this paper's method is slightly lower than the original method using relational database, while the throughput of the blockchain system is slightly reduced. Compared with the original method, the proposed new method is advanced because it takes into account the fact that a large amount of data collaboration does not require frequent access to nodes and node interactions, and then focuses on the existence of frequent and repetitive node query requirements. The higher average latency within the blockchain is exchanged for the improvement of query efficiency in specific scenarios. Under the comparison of only 20000 experimental data, the average delay and the native method is basically the same at 0.27s, which brings the double improvement of sending efficiency and throughput, respectively, about 5.5%, in the real environment of a larger amount of data under the new method to adapt to the background of the needs of this paper is bound to bring a greater breakthrough in the benefits.

Table 4: Transmission rate and throughput contrast

		Original method	Paper method
Visit	Send Rate	112.6	110.8
	Throughput	109.5	106.4
Query	Send Rate	208.8	235.6
	Throughput	207	235.1
Node interaction	Send Rate	112.5	98.6
	Throughput	102.3	92.7

Table 5: Latency comparison of access, query and node interaction

		Original method	Paper method
Visit	Max Latency	1.88	1.98
	Min Latency	0.33	0.33
	Avg Latency	1.11	1.16
Query	Max Latency	0.5	0.68
	Min Latency	0.03	0.03
	Avg Latency	0.27	0.36
Node interaction	Max Latency	1.46	2.23
	Min Latency	0.35	0.3
	Avg Latency	0.91	1.27

3.2.2 Efficiency of on-chain and off-chain retrieval

The second experiment not only evaluates the overall retrieval efficiency, but also compares the

retrieval efficiency of the internal on-chain and off-chain separation experiments. The experimental dataset is still 20,000 pieces of source data. In the on-chain part, the smart contract is invoked to query the dataset with the data index stored in the blockchain head after separation, i.e., based on the method of this paper and the Fabric native retrieval method, respectively. In this way, we compare the querying time of the blocks in the blockchain network between the method proposed in this paper and the Fabric native retrieval method.

The experiments in the on-chain part of the blockchain are only for the retrieval time comparison between this paper's proposed method and Fabric's native retrieval couchDB, and the comparison of the native retrieval time is shown in Fig. 8. The overall view of this paper's method of time consumption are better than the native system, and by the trend can be seen with the increase in the number of retrieval, the method of this paper's retrieval is different from the native method, its time consumption is a downward trend, which is also consistent with the theory of the design of the method, the more the amount of data in this paper's method is the performance of the more friendly.

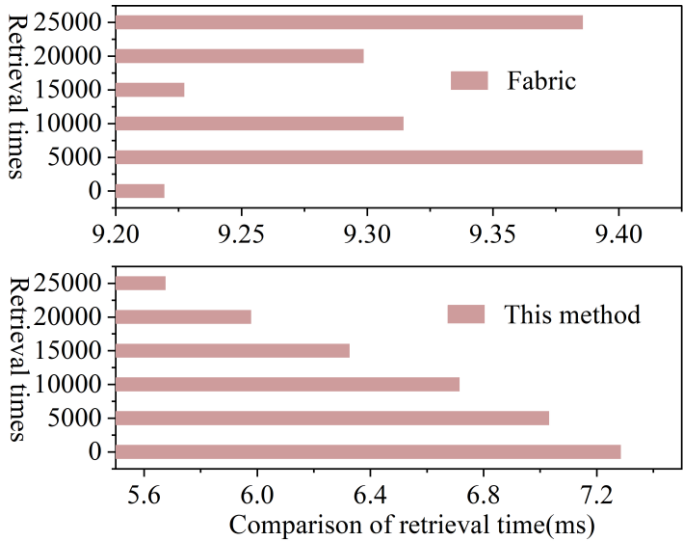

Figure 8: Comparison of native search time

Different from the on-chain comparison experiments, the off-chain as well as on-chain and off-chain retrieval experiments partially increase the repeated retrieval mechanism, and the off-chain retrieval time comparison (ms) is shown in Table 6. In the off-chain part, the red-black tree clearly shows the advantage of retrieval efficiency under multiple comparison experiments, and the addition of the Buffer stack does not have any effect, but only a slight memory loss, which is also a strategy of utilizing space for time. This is also a strategy of utilizing space for time. Repeated single-article searches can see the huge time savings with small memory loss, which is almost 10ms.

Table 6: Search time comparison

	Red-Black tree	Red-Black tree&Buffer	SkipList	MySQL
FirstSingleSearch	11.92	13.04	15.18	16.25
DuplicateSingleSearch	13.04	4.74	14.47	14.82
GlobalRetrieval	88.57	88.93	91.78	165.53

This section also needs to utilize the Improved Red-Black Tree algorithm to verify the efficiency improvement during data insertion before performing the overall data retrieval. The

insertion time comparison is shown in Table 7. In comparison with the current system built in this section for data insertion, the method in this paper under blockchain has a significant improvement in overall efficiency, about 155ms under 20,000 pieces of data, although there is an increase in the number of times of adjusting the structure of the red-black tree compared to the previous one.

Table 7: Comparison of insertion time

Data set size	Current Program	Improved
5000	93.38	39.62
10000	133.70	87.01
15000	227.79	127.33
20000	349.46	194.54
100000	1330.65	672.04
1000000	2312.54	1653.23

The ensuing clever combination of the advantages of on-chain and off-chain retrieval circumvents the possible mutual shackles, and the comparison of on-chain and off-chain retrieval times is shown in Fig. 9. The on-chain is the method of this paper and Fabric native respectively, while the off-chain is unified to use the indexing method proposed in this paper. It can be seen that within all the first retrieval experiments, the method proposed in this paper always outperforms the native method, and similarly, after the first retrieval of each group and then repeat the same condition retrieval, the figure demonstrates that the repeated retrieval has a significant efficiency enhancement compared to the first retrieval, and at the same time, with the increase of data size, the retrieval time of the two retrieval schemes also grows gradually, and the two increase lines, although both of them are positively growing, the slope of the new approach is Obviously slowing down, indicating that as the data size increases, the number of blocks constructed in the blockchain also gradually increases, and the probability of being able to match the hit in this paper's method also greatly increases, which is quantitatively reflected in the reduction of retrieval time. Based on the method of this paper and improve the red-black tree & Buffer retrieval constitute the method of application of this program, its retrieval effectiveness in the blockchain chain under the embodiment of the index itself linked. As can be seen from the figure, the logarithmic function can be well constructed out of the retrieval efficiency, and the utilization of retrieval efficiency has a strong mathematical correlation with the number of blocks in the blockchain, the number of index hits. For the specificity of data retrieval, the data prefix as the main interaction keyword can well improve the retrieval efficiency.

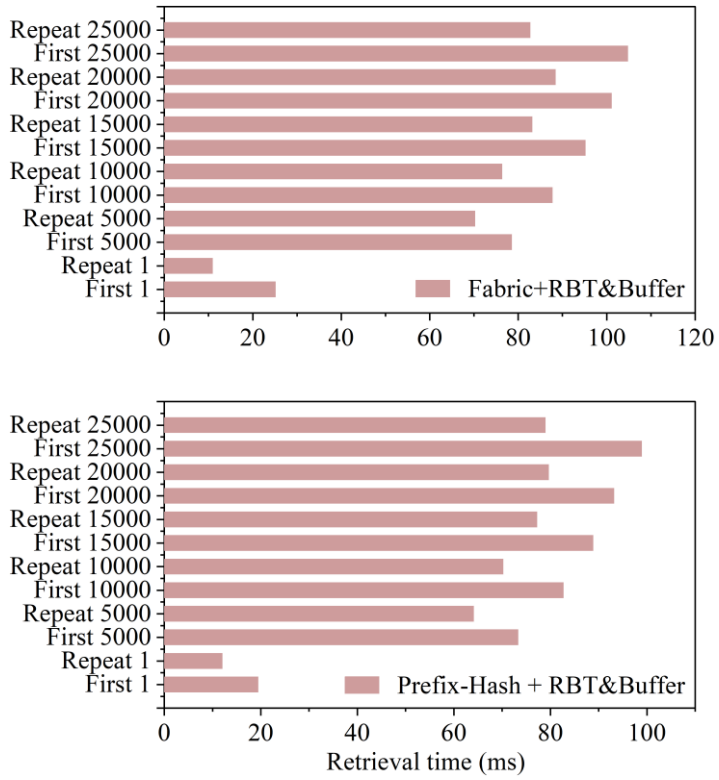

Figure 9: Comparison of search time in chain and chain

In summary, the data retrieval model based on the combination of on-chain and off-chain adopts blockchain technology to ensure data security, and provides efficient data retrieval interfaces through the combination of on-chain and off-chain indexes. The experiment verifies that the proposed scheme can significantly improve the retrieval efficiency and blockchain performance compared with the existing schemes.

4 Conclusion

In this paper, we propose a heterogeneous data evolution model for distribution network based on blockchain algorithm, and through experiments, we show that the model can ensure the authenticity and reliability of the data in supply chain traceability while having high performance to meet the requirements of practical applications. The main research conclusions are as follows:

(1) The DT-PBFT algorithm proposed in this paper is compared experimentally with the PBFT algorithm and the CPBFT algorithm, and the experimental results show that the DT-PBFT algorithm proposed in this paper is better than the above two algorithms in the four aspects of the consensus latency, system throughput and security. For example, in the case of 1M data block size, the consensus delay of DT-PBFT algorithm is reduced by 18.08% and 6.05% compared with PBFT algorithm and CPBFT algorithm respectively.

(2) In the blockchain system throughput and delay, under the comparison of 20000 experimental data, this paper's method can bring about a double improvement in the sending efficiency and throughput, which is about 5.5% respectively, under the situation that the average delay is basically the same as that of the native method at 0.27s. This proves that the method in this paper can effectively improve the retrieval efficiency and maintain a high degree of security.

About the Author

Yipeng Liu (1999-5), male, Han ethnicity, from Guiyang, Guizhou Province, holds a bachelor's degree and works as an assistant engineer specializing in ubiquitous power distribution operations.

Wei Dong (1987-12), male, Han ethnicity, from Kaiyang, Guizhou, Engineer with a bachelor's degree, specializing in ubiquitous power generation.

Wenting Wang (1991-8), female, Han ethnicity, Guiyang, Guizhou, bachelor's degree; Engineer, the research direction is ubiquitous command and control business.

Jun Cao (1983-10), male, Han ethnicity, Huanggang, Hubei, Bachelor's degree, Senior Engineer, specializing in power equipment operation and maintenance as well as production command.

Yuewei Tian (1989-04), female, Han ethnicity, from Zunyi, Guizhou, Senior Engineer with a bachelor's degree, specializing in power equipment operation and maintenance management.

References

[1] Abeysinghe, S., Abeysekera, M., Wu, J., & Sooriyabandara, M. (2020). Electrical properties of medium voltage electricity distribution networks. CSEE Journal of Power and Energy Systems, 7(3), 497-509.

[2] Ganjkhani, M., Gholami, A., Giraldo, J., Srivastava, A. K., & Parvania, M. (2023). Multi-source data aggregation and real-time anomaly classification and localization in power distribution systems. IEEE Transactions on Smart Grid, 15(2), 2191-2202.

[3] Yuan, G., Zhou, Y., Zhang, H., & Zhang, Y. (2021, December). Analysis and Application of Distribution Network Operation Based on Multi-source Data Fusion. In 2021 IEEE Sustainable Power and Energy Conference (iSPEC) (pp. 3988-3992). IEEE.

[4] Yuan, Q., Pi, Y., Kou, L., Zhang, F., Li, Y., & Zhang, Z. (2022). Multi-source data processing and fusion method for power distribution internet of things based on edge intelligence. Frontiers in Energy Research, 10, 891867.

[5] Zhou, J., Chen, Y., Zheng, M., & Wang, W. (2022). Data distribution for heterogeneous storage systems. IEEE Transactions on Computers, 72(6), 1747-1762.

[6] Wu, J., He, J., & Tong, H. (2024, August). Distributional network of networks for modeling data heterogeneity. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 3379-3390).

[7] Gahar, R. M., Arfaoui, O., Hidri, M. S., & Hadj-Alouane, N. B. (2019). A distributed approach for high-dimensionality heterogeneous data reduction. IEEE Access, 7, 151006-151022.

[8] Tan, Y., Liu, W., Su, J., & Bai, X. (2018). Generative adversarial networks based heterogeneous data integration and its application for intelligent power distribution and utilization. Applied Sciences, 8(1), 93.

[9] Wang, G., Gunasekaran, A., & Ngai, E. W. (2018). Distribution network design with big data: Model and analysis. Annals of Operations Research, 270(1), 539-551.

[10] Li, M., Tan, J., Zhang, J., Tan, X., & Luo, T. (2025). Design of Multi-Source Heterogeneous Data Fusion Algorithm for Distribution Networks Based on Improved Kalman Filter. IEEE Access.

[11] Wu, B., & Hu, Y. (2023). Analysis of substation joint safety control system and model based on multi-source heterogeneous data fusion. IEEE Access, 11, 35281-35297.

[12] Li, X., Wang, Z., Leung, V. C., Ji, H., Liu, Y., & Zhang, H. (2021). Blockchain-empowered data-driven networks: A survey and outlook. ACM Computing Surveys (CSUR), 54(3), 1-38.

[13] Li, X., Han, B., Li, G., Luo, L., Wang, K., & Jiang, X. (2021). Dynamic topology awareness in active distribution networks using blockchain-based state estimations. IEEE Transactions on Power Systems, 36(6), 5185-5197.

[14] Tseng, L., Wong, L., Otoum, S., Aloqaily, M., & Othman, J. B. (2020). Blockchain for managing heterogeneous internet of things: A perspective architecture. IEEE network, 34(1), 16-23.

[15] Shen, M., Zhu, L., & Xu, K. (2020). Secure homogeneous data sharing using blockchain. In Blockchain: Empowering Secure Data Sharing (pp. 39-59). Singapore: Springer Singapore.

[16] Bowen, H., Yi, L., Li, F., Xinhua, D., & Ping, C. (2019, December). Blockchain-based access control data distribution system. In 2019 IEEE 5th International Conference on Computer and Communications (ICCC) (pp. 1231-1236). IEEE.