The following publication W. Xu et al., "Blockchain-based Verifiable Decentralized Identity for Intelligent Flexible Manufacturing," in IEEE Internet of Things Journal, vol. 12, no. 16, pp. 32366-32378 is available at https://doi.org/10.1109/JIOT.2025.3576735.

Blockchain-based Verifiable Decentralized Identity for Intelligent Flexible Manufacturing

Wenjian Xu, Jiamin Deng, Jialong Yu, Shanghui Mao, Youhuizi Li, Zhe Peng, and Bin Xiao, *Fellow, IEEE*

***Abstract*—The manufacturing environment and activities with a large volume and variety of product data have put forward higher requirements for the proof and verification of identity information. Achieving decentralized digital identity management in the Industrial Internet of Things (IIoT) helps to improve the performance of relevant proofs and authentication. The Decentralized Identity (DID) system serves as a bridge between the physical and digital worlds, assigning digital identities to physical entities to facilitate their participation in online activities. However, faced with the huge number of manufacturing entities accessing the DID system, the number of DID documents in the system has proliferated. It is still a big challenge to improve the scalability of the system while ensuring the efficiency of information access and verification. In this paper, we propose a blockchain-based verifiable decentralized identity system for IIoT. First, we propose a blockchain-based system architecture with a specially designed storage structure for DID documents. Specifically, we design a structure based on Merkle Tree that visually summarises the physical associations of manufacturing entities and reduces access overhead. Second, we design a multi-block storage structure within the blockchain, which establishes inter-block jumps based on the associated DID, effectively improving the query efficiency of the system. Finally, we design a verification scheme that enables users to verify the integrity of the identity data of the proof provider. We implemented the system framework and conducted experiments to evaluate the performance of our system. The experimental results proved the effectiveness of the system.**

***Index Terms*—Blockchain; decentralized identity; identity verification; Web 3.0.**

I. INTRODUCTION

WEB3.0, envisioned as the next generation of the internet, represents a significant shift towards system decentralization, enhanced data security, and self-sovereign identity [1]. Web 3.0 is revolutionizing industrial manufacturing by enhancing supply chain transparency, enabling smart automation, improving security, and introducing decentralized financial solutions. By integrating blockchain, artificial intelligence (AI), and Industrial Internet of Things (IIoT), manufacturers can achieve higher efficiency, lower costs, and increased sustainability, marking a significant shift toward Industry 4.0 and beyond. The rapid growth of decentralized applications (DApps) [2] and decentralized finance (DeFi) [3] platforms exemplifies the development of intelligent manufacturing, reshaping the digital service landscape by providing more transparent, secure, and user-governed alternatives to traditional centralized models. These advancements emphasize the transformative power of intelligent manufacturing and Web 3.0 in building a more resilient and equitable digital ecosystem, overcoming many of the inherent limitations of existing manufacturing structures.

Although there exist a few projects that provide decentralized data storage [4]–[8], such as Filecoin, Oort, Arweave, and Storj, trusted identity management of entities in flexible manufacturing is essential for ensuring the security and governance of the Industrial Internet of Things. Existing centralized digital identity systems cannot effectively realize the validation and ownership of data. On the one hand, since identity information is stored and managed by a central authority, such a system model would be vulnerable to a single point of failure. Moreover, as data cannot be accessible among various and isolated manufacturing service providers, users are required to repeatedly register similar identity information for different online activities. This redundancy not only makes the process tedious but also creates opportunities for data falsification. Moreover, the application-oriented Industrial Internet of Things also relies on identity data to securely manage large amount of devices. During various manufacturing production procedures, the identity information collected by different sensing devices could be cluttered, scattered, and even conflicting. Inefficient data storage will also pose a significant obstacle to subsequent production-related data management, such as data integration and data validation.

The emergence and development of blockchain technology offer promising solutions to long-standing challenges in data security and collaborative processes of intelligent and flexible manufacturing. As a distributed ledger technology, blockchain decentralizes data storage by distributing it across a network of nodes, thereby eliminating the dependence on a single central authority and enhancing system resilience. This architecture effectively mitigates the risks associated with a single point of failure, such as data loss, unauthorized access, and service interruptions, which are common in traditional centralized

Manuscript received January 31, 2025. *(Corresponding author: Zhe Peng, Youhuizi Li.)*

Wenjian Xu is with the the School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China, and also with Zhejiang Key Laboratory of Biomedical Intelligent Computing Technology, Hangzhou 310023, China (e-mail: wenjian.xwj@zust.edu.cn).

Jiamin Deng is with the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: jiamin.deng@connect.polyu.hk).

Jialong Yu is with the School of Geographical and Earth Sciences, University of Glasgow, Scotland, United Kingdom (e-mail: 3068572Y@student.gla.ac.uk).

Shanghui Mao and Youhuizi Li are with the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: 20051119@hdu.edu.cn, huizi@hdu.edu.cn).

Zhe Peng is with the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, and The Hong Kong Polytechnic University Shenzhen Research Institute (e-mail: jeffrey-zhe.peng@polyu.edu.hk).

Bin Xiao is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong (e-mail: b.xiao@polyu.edu.hk).

systems. Furthermore, blockchain's inherent transparency and immutability ensure that all data recorded on the ledger is accessible to network participants with the appropriate permissions, fostering trust and accountability. This transparency enables various manufacturing service providers to collaboratively access, validate, and authenticate data in a decentralized manner, thereby streamlining the data verification process and significantly improving overall system efficiency. By facilitating secure, transparent, and decentralized data management, blockchain technology not only addresses critical data integrity issues but also paves the way for more robust and reliable intelligent manufacturing.

In the era of Industry 4.0, intelligent manufacturing relies heavily on interconnected systems, autonomous decision-making, and real-time data exchange. A major challenge in this ecosystem is ensuring secure and verifiable digital identities for a large number of manufacturing entities, including machines, sensors, robots, and human operators. Decentralized identity (DID) underpinned by blockchain technology is the key to intelligent, flexible manufacturing. The DID concept was first organized and proposed for specification by the World Wide Web Consortium (W3C) in 2019. In a blockchain-based DID system, the identity data of a manufacturing entity (e.g., a user or a device) is stored and managed by a decentralized blockchain network instead of a single authoritative organization. For a specific identity, a DID document will be generated, consisting of a DID identifier, timestamp, entity information, and verification information. Specifically, the DID identifier is the unique identifier of a DID document and the corresponding entity. Through its DID identifier, a DID document can be retrieved and identified efficiently in the DID system. Moreover, the concrete entity-related data will be recorded in the entity information portion, while verification information will be used to verify the data integrity of the DID. As such, the digitalized identity of a physical manufacturing entity would be created and stored in the information system, enhancing the interaction and coordination between the physical world and the computational world. Benefiting from the underlying blockchain, the DID of a manufacturing entity can help to improve both data security and data sovereignty for the Industrial Internet of Things. By integrating verifiable decentralized identity into intelligent manufacturing, industries can achieve secure, autonomous, and scalable identity management for their IIoT environments.

Implementing a blockchain-enabled DID system may pose distinct challenges. First, *data integrity* ensures that identity data is correct and has not been tampered with throughout an authentication process. For an identity verifier, incomplete or illegally modified data should be identified and cannot be accepted. Second, the data structure of the DID document determines the *efficiency* of data storage and verification in the blockchain-based system. In a complex manufacturing environment, a large number of manufacturing entities will be continuously registered and accessed in the DID system. Due to the inherent constraints in the manufacturing process, the relationships between different entities and their corresponding digital identities become very complicated, which would seriously affect system performance. Therefore, it is necessary to improve system efficiency so that DID documents have less storage overhead and less time overhead for identity verification.

To address the above challenges, we propose a blockchain-based verifiable decentralized identity (DID) system for intelligent flexible manufacturing. Different from existing designs, our initiative not only adopts decentralized patterns in its design, but also addresses the issues of data integrity and system efficiency, making it a comprehensive design. Specifically, we design a storage structure and data validation scheme for DID documents in the blockchain to ensure data integrity within the system and effectively reduce data storage and validation overheads to improve system performance. Concretely, we make the following contributions in this paper.

- We introduce a blockchain-based verifiable DID system to facilitate verifiable identity management in a decentralized manner.
- We propose a customized DID document storage structure that efficiently integrates verifiable identity data and effectively reduces data storage overhead.
- We introduce a novel multi-block linked table structure to enhance DID search efficiency, enabling rapid traversal and retrieval of all associated DIDs from a single DID, thereby optimizing the system's overall performance.
- We design a data validation scheme that enables the verifier to verify the integrity of the received identity data.
- We implemented the proposed system architecture and evaluated it experimentally as a way to demonstrate the feasibility of the system.

The rest of this paper is organized as follows. Section II introduces some related techniques. Then, Section III describes an overview of the verifiable decentralized identity system, followed by the details of the system design and evaluation in Sections IV and V, respectively. Finally, we conclude our paper in Section VI.

II. RELATED WORK

In this section, we construct a brief survey on the technologies behind our system, namely decentralized identity, blockchain data verification, and blockchain aided manufacturing.

A. Decentralized Identity (DID)

Identity is the basis of inter-entity interaction for all entities, including humans. As a new type of identity management system, DID allows individuals to manage their identities on their own without relying on third-party authentication organizations. It makes the authentication process safe and secure by providing a more convenient and efficient way of authentication [9]. At the same time, more researchers begin to devote themselves to DID-related research and present a series of research results [10]–[13].

In terms of practical applications, DID has already made its debut with its advantages. There exist some common issues in banking transactions, including high costs for cross-border payments, long transaction delays, and a lack of transparency. To address these issues, Md Mainul Islam et al. proposed a

cryptocurrency system [14]. This system applies decentralized identifiers (DID) to enable self-administered authentication and achieve auditability. Eranga Bandara et al. developed Connect [15], a digital contact tracking platform inspired by DID to track and control susceptible populations. With features like autonomous identity management, virus propagation information can be tracked and synchronized with public disclosures in a timely manner. This helps reduce the speed of infection spread and supports effective control of virus development. However, few studies have paid attention to the problem of document scalability in DID systems. Irregular storage also makes it more difficult to manage related entities.

B. Blockchain Data Verification

Blockchain is a new type of distributed data storage system born with the emergence and rise of cryptocurrencies. With its decentralization, traceability, and transparency, blockchain has seen widespread adoption in academic and industrial applications [16]–[18]. As a necessary part of consuming data, data validation ensures the reliability of the data when consumers use it for subsequent activities. More research and applications tend to consider the blockchain as a tool for data verification [19]–[21]. Its tamper-proof characteristics ensure that the original data it stores remains unchanged and protected from any modifications or damage. This provides a crucial guarantee for verifying the integrity and accuracy of the information.

The DID system, established based on the blockchain, relies on the unique features of the blockchain to achieve the concept of decentralization. Under that, users' privacy and the safety of identity verification data are guaranteed. IoT interacts closely with identity information. Authentication is the crucial functionality in traditional identity management systems, and it faces challenges such as high authentication overheads, device tracking, and a single point of system failure. Solutions based on DID systems, such as BDIM [22], SmartDID [23], and the Access Control Architecture [24], have proven effective in addressing these challenges, ensuring the privacy, security, and efficiency of the systems.

Traditional DID systems struggle with scalability due to the rapid growth of DID documents. The proposed system introduces a specially designed storage structure based on Merkle Trees, which efficiently organizes and visually summarizes the physical associations of manufacturing entities. Existing DID systems often face performance issues when handling a massive number of manufacturing entities. Our system implements a multi-block storage structure within the blockchain, allowing for inter-block jumps based on associated DID records.

C. Blockchain-Aided Manufacturing

Blockchain technology, as a decentralized distributed ledger system, has been widely adopted in Industrial Internet of Things (IIoT) and smart manufacturing [25]. It enhances security, collaboration, operational efficiency, product traceability, and quality control within enterprises and supply chains [26]–[29]. With its immutable, transparent, and decentralized features, the blockchain effectively addresses information asymmetry and trust issues in supply chain management, facilitating smarter industrial transformations. The application of blockchain technology in smart manufacturing has demonstrated significant value. For example, Wen et al. [30] proposed a blockchain-based supply chain information distribution system that connects IIoT nodes, ensuring data transparency and consistency. By integrating smart contracts, the system optimizes data exchange, significantly enhancing collaboration efficiency among supply chain nodes. Additionally, blockchain enables the secure sharing of information resources from private domains and provides product provenance information to consumers through smart contracts [31], [32]. This not only strengthens supply chain security, but also effectively addresses issues such as product recalls caused by security vulnerabilities [33], [34]. Blockchain provides a transparent and adaptable solution for product traceability and supply chain management in the manufacturing industry.

Blockchain also optimizes the operation of smart manufacturing systems. Leng et al. [35] introduced the ManuChain dual-layer intelligent model, combining blockchain with global optimization to address conflicts between overall planning and local execution. Through smart contracts and digital twins, blockchain enhances manufacturing flexibility, efficiency, and dynamic optimization. In supply chain management, Chang et al. [36] proposed a blockchain-based reconfiguration framework that uses smart contracts to enable real-time tracking of cash flow and logistics, thus reducing intermediaries and improving cost efficiency. Geiger et al. [37] suggested blockchain solutions for transparent production tracking and dynamic task optimization in the machine-sharing economy, ensuring data authenticity and adaptability to market changes.

Blockchain also plays a critical role in enhancing trust management in supply chains. Zhang et al. [38] proposed a framework that combines blockchain and smart contracts to reduce the "trust tax", improving transaction transparency and efficiency. This framework benefits SMEs by lowering costs, enhancing product quality, and boosting competitiveness. Wu et al. [39] optimized trust mechanisms with an improved EigenTrust algorithm to enhance supply chain reliability, while Zhang et al. [40] emphasized blockchain's role in fostering secure, transparent stakeholder collaboration. Leng et al. [41] surveyed the integration of blockchain technology in securing smart manufacturing systems within Industry 4.0, identifying eight key cybersecurity issues and proposing ten metrics for implementing blockchain solutions. Most existing blockchain-based manufacturing systems focus on the issues of data transparency and supply chain security, but few of them talk about the storage of verifiable data. In this paper, we propose a customized DID document storage structure that efficiently integrates verifiable identity data and effectively reduces data storage overhead.

III. SYSTEM OVERVIEW

As mentioned above, in this section, we propose a verifiable decentralized identity system model with blockchain imple-

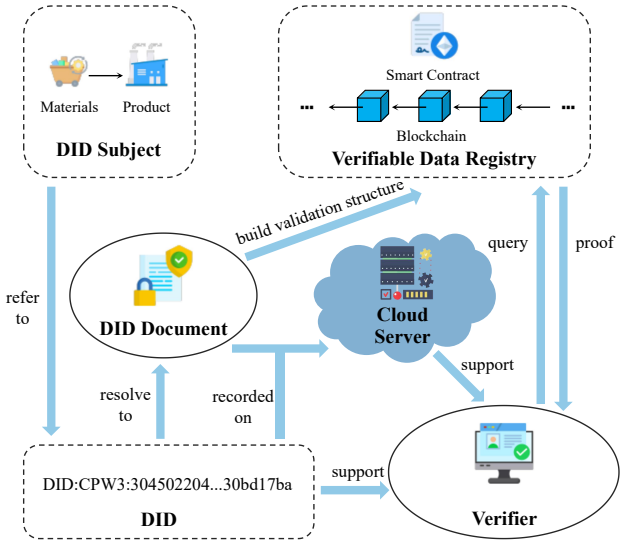


Fig. 1. System Model

mentation. Then, we present the threat model for our proposed system.

A. System Model

The five main components of the scheme are shown in Figure 1: (i) *DID subject*, (ii) *DID and DID document*, (iii) *Cloud Server*, (iv) *Verifiable Data Registry*, and (v) *Verifier*. In this scheme, the supplier provides materials to a superior supplier or manufacturer, who obtains materials and processes them to produce a composite material or product. Whether it is a raw material or a product, it exists as a DID subject at this time. The data relating to the subject is automatically collected and sorted out using IOT devices configured in advance. It is processed by a DID Method to get the unique identifier belonging to the subject. This identifier could be resolved to a generated DID document, which is uploaded and stored in the cloud server. At the same time, the DID document uploaded to the cloud server is used to construct a special structure stored in the blockchain. It will be used subsequently to apply the DID for verification.

The structure corresponding to DID documents and other DID-related data saved by the blockchain cannot be tampered with once it is constructed in the block on the chain, which is an important retention of the original data. Depending on the requirements, the holder of the identity data sends Verifiable Credentials (VC) or Verifiable Presentation (VP) to the verifier. It contains the data to be verified and a DID identifier pointing to the relevant DID document. The verifier can use the identifier to find and access the corresponding DID document stored on the cloud server to check the entity information. And complete the validation of the data reliability by checking the backups saved on the blockchain. For example, manufacturers can confirm the conformity of materials by accessing the DID document of recorded materials. By checking the DID document of products, distributors and customers can confirm the product selection specifications and production channels. On the one hand, using DID documents to complete data verification and other interactive aspects can ensure the authenticity of the data. On the other hand, the designed storage structure can effectively reduce the data access overhead, improve the system scalability, and help the verifier verify the data reliability more efficiently.

B. Threat Model

In this scheme, the DID is exposed to potential security threats with the data records in the document. On the one hand, when collecting and organizing the relevant data, the recorded values may deviate from the actual values due to unexpected malfunctions of the monitoring equipment. On the other hand, there may be human intervention to tamper with recorded data for certain commercial interests. Nodes participating in the blockchain network are responsible for performing computations; however, some may be compromised by adversaries and produce incorrect results. In line with the standard threat model adopted by many blockchain systems [42], we assume that at least ζ nodes remain honest and consistently provide correct computation results, where ζ is a constant greater than 2/3. Furthermore, without loss of generality, we assume that all nodes possess equal computational resources. Specifically, we define the following security criteria:

- **Soundness.** All data recorded in the document meets actual expectations and has not been tampered with or corrupted.
- **Completeness.** The documentation is complete with all of the entity's data information, with no omissions.
- **No single point of failure.** A blockchain system must ensure security even if a single node fails to return results or is compromised by attackers.
- **Efficient verifiability.** Users should be able to efficiently verify the correctness of the retrieved DID and easily access verifiable information for a specific version.

IV. SYSTEM DESIGN

In this section, we introduce our system, a blockchain-based verifiable decentralized Identity solution. We begin by outlining the fundamental elements of the DID system, i.e., Decentralized Identifier and DID Document. We then proposed the Authenticated Data Structure (ADS) of the DID Document and its storage within the block. Following this, we introduced an efficient structure that spans multiple blocks and connects various related DIDs, significantly enhancing the system's query efficiency. After that, we proposed a decentralized verification scheme that ensures efficient verification of query results within the system. We then presented an optimized verification scheme for authenticating DID entities in the scenario of multi-supply chain system. Finally, we discussed the user privacy issue in our system.

A. DID and DID Document

The DID identifier is an identity indicator for all registered entities in the DID system and points directly to each specific entity. As mentioned earlier, in our proposed scheme, during the process of generating DID subjects, production and data

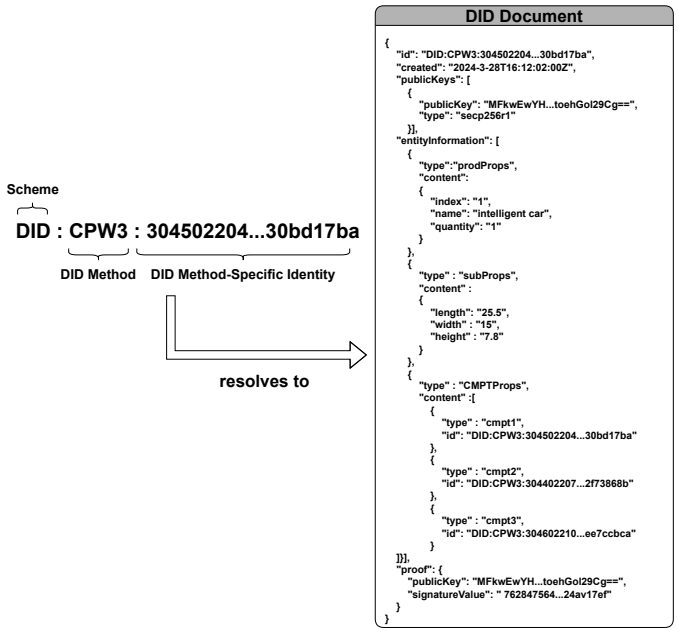


Fig. 2. DID and DID Document used in this system

related to the subject are collected to generate this unique DID identifier via a DID method named DID: CPW3. As shown in Figure 2 for the specific structure of a DID identifier, it can be seen that it consists of three parts, respectively, "fixed string (scheme name)", "did method", and "did method-specific identity", connected by ":". "Fixed string" specifies that the identifier is a DID identifier, "did method" indicates the specific scheme for defining and manipulating the identifier (in this system, that is, DID: CPW3), and "did method-specific identity" is an identifier string that is a unique value in the DID system.

When each DID identifier is generated, its corresponding DID document is also created. In these documents, public information such as the subject's attributes, public key, and verification method are recorded. The specific structure of the DID document and its saved data content in this system is shown in Figure 2. The DID and its corresponding DID document will be uploaded together by the registrant to the cloud server for storage. At the same time, the DID document will be stored specially in the blockchain (it will be explained in the next subsection), which will serve as an important guarantee for the reliability of the data.

B. Verifiable DID (vDID)

We constructed a Merkle Hash Tree (MHT) based in-block storage structure for DID documents to support data reliability verification. As illustrated in Figure 3, our vDID structure represents a DID document storage framework for a product stored on the blockchain. This vDID structure consists of a two-layer tree configuration, comprising an outer-layer ADS and an inner-layer ADS. The inner-layer Merkle tree is constructed from the data entries of the DID document, while the outer-layer Merkle tree is built upon the component hierarchy.

For example, a manufacturer uploads the DID documentation of a smart electric vehicle that consists of three components. Each of the three components is obtained by processing one or more raw materials, including primary raw materials or multistage raw materials, which may come from different raw material suppliers. such as "publicKeys", "SubProps", "ProdProps", "CMPTProps", and "proof". The inner-layer ADS is organized as a complete binary tree, where each leaf node stores a digest computed from the corresponding data fields of the DID document, such as "publicKeys", "SubProps", "ProdProps", "CMPTProps", and "proof". Meanwhile, each non-leaf node contains a digest derived from its two child nodes. For example, $h_{d0} = H(d_0)$ and $h_{d4} = H(h_{d0}|h_{d1})$, where "|" denotes the string concatenation operator. The root digests of inner-layer trees are utilized as part of the input for calculating the digests of the outer-layer ADS nodes.

All nodes in the outer-layer tree correspond one-to-one with the car and its component ingredients. To simplify the illustration process, the three components in this scenario are the controller, the sensor, and the body. Specifically, the car N_n consists of the components N_7(controller), N_8(sensor), and N_9(body), which are each in turn composed of other components or raw materials(N_1 to N_6, for leaf nodes). In other words, the outer-layer tree provides a complete overview of the DID subject corresponding to the product and all the raw materials associated with it. For leaf nodes of the outer-layer ADS, as primary material, the "ProdProps" and "SubProps" property values are contained within the node. Take node N_1 as an example, $h_1 = H(N_1|h_{d0})$, where N_1 is the node identifier and h_{d0} is the root digest of the corresponding inner-layer ADS. For non-leaf nodes, since they are composite components(products), an attribute called "CMPTProps" is attached to the node in addition to the two attribute values contained in the leaf node. "CMPTProps" represents the component properties of this entity; for example, N_1(CPU) and N_2(SSD) are the property values of N_7(controller). The computed hash of a non-leaf node, take N_n as an example, is $h_n = H(N_n|h_{d0}|H(h_7|h_8|h_9)$, where N_n is the node identifier, h_{d0} is the root digest of the corresponding inner-layer ADS, $H(h_7|h_8|h_9)$ is the digest of child nodes. Each node gets its hash value by the above calculation and recursively from bottom to top to get the final unique hash value (i.e., $hash_{ADS}$) saved in the block header. When a query is made for the data item d_0, it will return results along with the proof=$\{N_n, h_7, h_8, N_9, h_4, h_5, N_6, h_{d1}, h_{d5}\}$.

C. Efficient Model

In this section, we introduce a novel multi-block correlation structure, namely the Tier Skip Chain (TSC). This structure is built upon the foundational DID system framework and emphasizes the relational characteristics between DIDs, enabling the system to efficiently fulfill query requests and thereby enhance overall system performance.

Tier Skip Chain (TSC): Tier Skip Chain (*TSC*) is a multi-block structure that establishes skip links based on hierarchical relationships between blocks associated with linked DIDs. During the query process, *TSC* allows the blockchain to

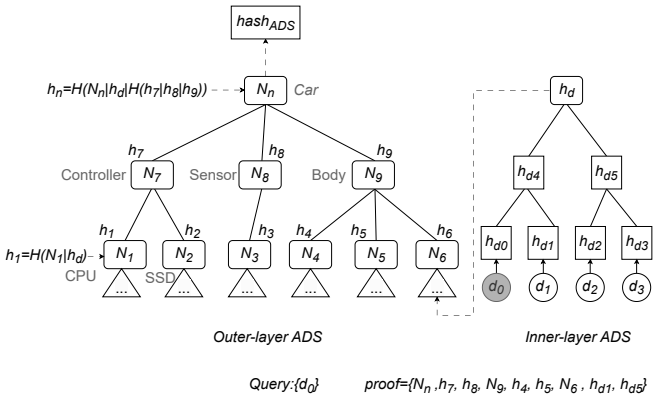


Fig. 3. vDID Structure

efficiently traverse any given DID and all its related DIDs through multiple associated skip links.

As depicted in the Figure 4, the *TSC* structure comprises five primary elements: (i) *entity node* N_i, (ii) *index node* I_i, (iii) *pruned node* D_i, (iv) *Tier Level* T_i, and (v) *Tier Link* L_i. The entity node stores complete data related to the corresponding entity, including the DID. The index node records the hash value of the block linked to the current block and block id. The pruned node only retains the DID of the corresponding entity. The Tier Level indicates the hierarchical relationship of the node within the entity or between entities. The Tier Link refers to the link between nodes of different blocks corresponding to their tier level. The relationship between Tier Level and Tier Link will be elaborated upon later in this section. These various nodes are organized into a tree structure within each block, ultimately forming an MHT. By leveraging this structure along with Tier Links, *TSC* enables logarithmic-cost traversal of the blockchain, allowing efficient queries for a specified entity and all its related entities. The intuitions behind the improvement are two-fold. Firstly, instead of duplicating entity data in multiple blocks, pruned nodes refer to earlier blocks, reducing data redundancy. Secondly, using skip links allows for efficient traversal across multiple blocks related to an entity, instead of sequential scanning. Without *TSC*, a straightforward solution would be scanning each block one by one to find relevant DID records, which is computationally expensive, especially as the blockchain grows.

The figure illustrates a simple example of the *TSC* structure. For simplicity, we assume that each block records only one complete entity (including composite entities, i.e., an entity composed of multiple distinct entities). Entities belonging to different Tier Levels are structured as nodes from bottom to top, constructing a tree known as the EntityTree within the MHT. Typically, an EntityTree maps to one or more entities registered in the system. Based on the hierarchical relationship and registration time of the entities, the nodes mapped within the *TSC* can be categorized into different types, namely entity node, index node, and pruned node.

The entity node records complete data related to the corresponding entity, including the DID and DID Document. Specifically, when any entity is first registered in the system, it is recorded as an entity node. Similar to the basic ADS data structure mentioned earlier, if the registered entity is a composite product, the node will contain attributes such as ProdProps, SubProps, and CMPTProps. Conversely, for primary raw materials or unprocessed entities, the entity node does not record the CMPTProps attribute. In the system, any entity can be registered multiple times. In the physical world, a lower-tier raw material or composite product already registered in the system may be used as a component to produce a higher-tier product, which is then registered in the system. In this case, the former entity will be registered at least twice in the system. That is, the EntityTree Root of a block (the lower-tier entity) is registered in a new block with the newly registered higher-tier entity, and it is mapped as a non-root node in the new EntityTree. The physical world's Tier Level of the entity determines its position within the EntityTree, and the different registration instances result in different node types within the tree. When an entity is registered multiple times in the system, its node will be represented as a pruned node within the EntityTree. This implies that for the same entity, its entity node must be registered in the system before the pruned node. When the pruned node is created, a corresponding index node is also generated. Referring to the Tier Level, the index node constructs the Tier Link between this block and the previous block.

The pruned node records the DID corresponding to the entity. Specifically, the complete and detailed information for the pruned node's corresponding real entity is recorded in the entity node when it is first registered in the system. When it appears as a pruned node in a different block, the primary information of the block's EntityTree will focus on the newly registered entity. The system can quickly locate the complete information of the entity by navigating from the pruned node through the index node and Tier Link to the block containing its entity node. Repeating this information in the new block would be redundant; therefore, the pruned node retains only the DID as location and verification information, suppressing other specific details to maintain the simplicity of the EntityTree.

The index node contains the location information of the target block to be navigated. Specifically, for the target block to be navigated, the node records the hash value and the block ID of that block. This hash value is obtained by recursively hashing all the fields in the target block's header. Tier Level represents the component hierarchy of each member entity within a composite entity, denoted as T_i, i $\in \{0, ..., n\}$. Specifically, in an n-Tier composite product, the entity at level n-1 is synthesized from the entity at level n, and in turn, is used to synthesize the entity at level n-2, and so on. In such a composite entity, their Tier Levels correspond to T_{n-1} = n-1, T_n = n, and T_{n-2} = n-2, respectively. The Tier Link functions as a skip link between blocks, with the index node of the subsequent block pointing to the previous target block. The pointing hierarchy of a Tier Link is related to the building relationship of the current node. For an EntityTree, each node within the tree is composed of two nodes. As previously mentioned, an entity node registered for the first time consists of both a pruned node and an entity node. Conversely, for

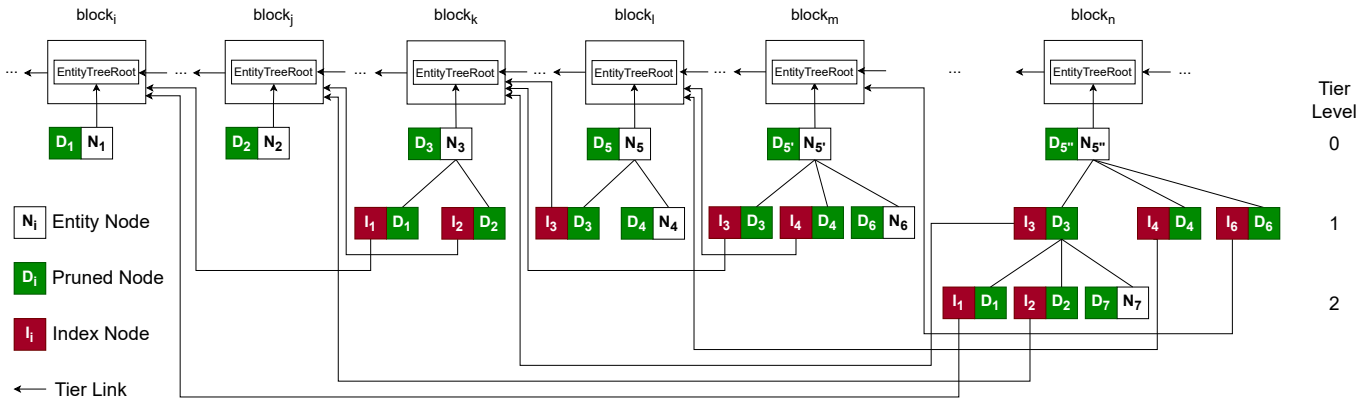


Fig. 4. Structure of a Tier Skip Chain

subsequent registrations, the entity node is composed of an index node and a pruned node. For any composite entity $E_i \in \{0, \ldots, n\}$, the root node always has a Tier Level of 0, while the corresponding leaf nodes have a Tier Level of n. For intermediate entities $E_j \in \{1, \ldots, n-1\}$, the entity at the j-Tier level and its associated entities at the same level form the entity at the j-1 level. Due to multiple registrations, the E_j node is composed of an index node and a pruned node, where the index node constructs a Tier Link pointing to the block that contains the complete information for E_j. In this direct parent-child relationship, the Tier Link constructed by the index node with a Tier Level $T_j = j$ corresponds to $T_j = 1$. In another scenario, for an indirect parent-child relationship, E_j' is registered in a new block as an intermediate component of E_i'. Compared to the previous registration of E_j, the updated component's Tier Level T_j' is 1≤t≤t_j, where t_j is the maximum Tier Level for E_j. When mapped to E_i', the updated component's Tier Level now corresponds to T_j' = t + $T_j^{i'}$, where $T_j^{i'}$ is the Tier Level of E_j' within E_i'. In this case, the corresponding Tier Link is $L_j = T_j$'. The advantage of Tier Link construction is that TSC can choose the appropriate Tier Link to jump according to different search needs, and complete the search and access to information more efficiently.

With this *TSC* structure, users can easily trace and traverse all related entities through any entity with logarithmic cost. Additionally, due to the presence of Tier Links, users can set filters to meet different levels of tracing granularity, enabling efficient retrieval and traversal based on the corresponding Tier Link.

Specifically, within the *TSC*, due to the structural changes of the EntityTree compared to the basic ADS structure discussed in the previous section, the hash values of nodes within the EntityTree are derived using different hashing methods. For nodes that do not contain the index node, if the node is an entity node, the hash calculation follows the method described in Section 2. If the node is a pruned node, its hash value is calculated as h_{pruned} = H(DID_{N_i}). For nodes that include an index node, the hash value of the node is computed as h_{node} = H(H(index node)|H(entity/pruned node)), where H(index node) = H($block_i$), and $block_i$ is the target block to which the index node points.

D. Data Verification

A distributor plans to purchase the product wholesale from the manufacturer and sell it to the market. In the process, the distributor needs to access the DID document of the product to confirm that there is no fraudulent use of specifications and materials.

Any entity that registers for access to the DID system must obtain a corresponding pair of public and private keys for identification. Similarly, a manufacturer registers a product by generating its own unique public and private key pair. After completing the construction of the MHT corresponding to the DID document in the block, the manufacturer uses the private key to sign its Merkle Root. The distributor uses the decentralized identifier to request and get the DID document from the cloud server. Then, the distributor verifies that the document was generated and uploaded by the manufacturer by checking the block ID and public key data stored in the document. Next, after confirming the authenticity of the document, the distributor would like to confirm that its product N_5" is manufactured as described by the manufacturer, using a composite material that contains the N_7 and meets its described specifications.

At this point, the distributor sends a verification request to the blockchain. For MHT stored on the corresponding block, the blockchain will start from the root node, search down the tree to verify the data, and return a proof (i.e., Merkle's path) that can rebuild the tree upon it. Specifically, after submitting a data validation request for N_5", the validator will receive a validation proof returned by the blockchain, which contains $h_{D_{N1}}$, $h_{D_{N2}}$, $h_{S_{N3}}$, $h_{node_{N4}}$, and $h_{node_{N6}}$. For distributors, h_7 can be calculated from local data ("ProdProps" and "SubProps" provided by the manufacturer). The MerkleRoot obtained through the N_7 public key appended to the document is used to compare with h_7 to complete its identification and determine the legitimacy of its source. Combined with the returned proof data, according to the calculation formula proposed in the previous section, the remaining local data is used to calculate $h_{node_{N3}}$ and MerkleRoot in turn, i.e., the reconstruction of the product's MHT is realized. Verification of the reliability of the data is accomplished by comparing the MerkleRoot obtained from the derivation of the reconstructed

MHT with the accessed original MerkleRoot stored in the block header.

Next, we analyze the computational complexity of the verification model. The verification process primarily involves retrieving a proof and reconstructing the hash path. Specifically, the blockchain provides a Merkle proof for a product, returning a set of intermediate hashes. To verify, the distributor reconstructs the Merkle path from the leaf node to the root by hashing pairs of nodes iteratively. Given a Merkle tree with n leaf nodes, the height is $O(\log n)$, so verifying a Merkle proof requires $O(\log n)$ hash computations. To this end, the overall computational complexity of the verification model is $O(\log n)$, making it efficient for large datasets.

E. Optimizations for Multiple Supply Chains

In this section, we present an optimized verification scheme for authenticating DID entities within a blockchain-based multi-supply chain system. To enhance verification efficiency, we incorporate an integrative locating layer into our framework.

In a multi-supply chain setting, a basic verification approach would involve establishing a dedicated *TSC* for each supply chain to track updates to verifiable objects. However, verifying newly updated global DID documents would require users to frequently retrieve multiple update logs and comply with the configurations of their respective *TSC*s. This method not only demands significant bandwidth and time from users but also increases the operational burden on *TSC* maintainers, who must manage a freshness service. To address these challenges, we further refine and optimize our system to facilitate efficient DID document verification within a blockchain-powered multi-supply chain framework.

A newly introduced *locating layer* has been designed and incorporated into our framework to support multiple supply chains. This layer collects and verifies authentication information for all DID documents stored on the blockchain, ultimately serving as a lookup service for users. A committee at the supply chain level is established to verify and integrate multiple independent *TSC*s, creating a unified supply-chain-level *TSC* (s-*TSC*). Within the s-*TSC*, each entry represents a snapshot of the supply chain's status. In order to generate a snapshot, the supply-chain-level committee gathers the most recent data from individual entities. Such data may contain verified signatures and hashes. Next, it verifies the authenticity of these signatures against the respective entity *TSC*. Finally, the committee signs the root, and the collected hashes are used to construct a Merkle tree, such that all entity versions within the snapshot are encapsulated.

This architecture facilitates the gradual upgrading of multiple supply chains within the blockchain. Entities without their own *TSC*s can still be incorporated into the integrative locating layer using their latest version's hash values. In addition, the supply-chain-level committee provides an aggregation timestamp service to prevent replay attacks, ensuring that the latest version of entities and the corresponding consistent state of the supply chain are sent to the users. After that, the committee could send the most recent signed supply-chain snapshot to the users. With this information, along with the Merkle proofs, outdated entities could be verified. If outdated entities are identified, users can retrieve the latest blocks from the individual *TSC*s based on their hash values.

Furthermore, a multiple-supply-chain blockchain can support multiple aggregation layers, each potentially representing different distribution categories, such as the development stage of entities. Within the *TSC*, the update history of each entity is preserved. For example, by assigning a new release tag to an entity update, users are notified of the latest distribution release. Simultaneously, participants can publish alternative versions of an entity (e.g., a test version) within the *TSC*. To ensure timeliness, a dedicated timestamp service is maintained for each distribution.

F. User Privacy

Protecting user privacy in blockchain-based verifiable decentralized identity (DID) systems is a crucial challenge. Since blockchain is immutable and transparent, implementing privacy-preserving techniques is essential. We have incorporated several mechanisms in our system to address this issue. For example, users should share only necessary identity attributes instead of full identity details, and zero-knowledge proofs (ZKPs) techniques are employed to allow users to prove possession of certain credentials without revealing the actual data. Moreover, we store identity-related data off-chain (e.g., IPFS, cloud storage) while keeping only cryptographic proofs on-chain. Other techniques, such as group signatures or blind signatures, are also possible [43].

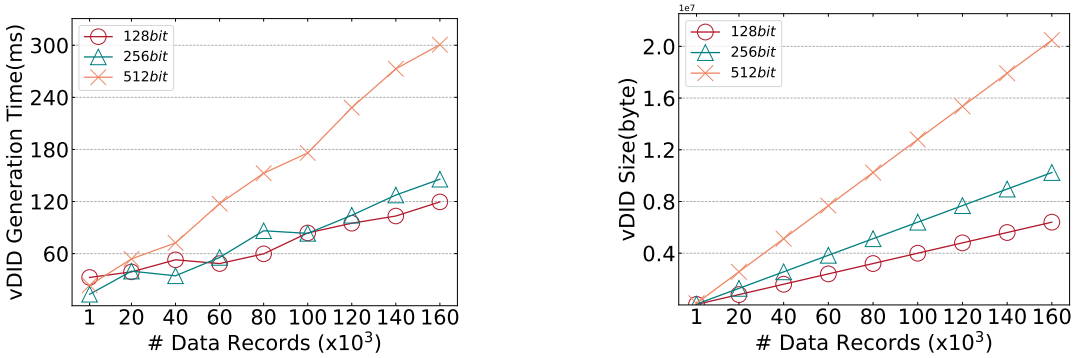
V. IMPLEMENTATION AND EVALUATION

In this section, we extensively evaluated our framework with real data. Specifically, we tested the vDID construction performance, proof performance, as well as the effectiveness of Tier Skip Chain and the multiple chain optimization.

A. Experimental Settings and Implementation

We construct a blockchain for simulation purposes using Docker (version 18.09.0) and Hyperledger Fabric (version 1.4). Go is used to develop the smart contract. We choose Hyperledger Fabric because it is commonly deployed in real-world manufacturing environments. In order to simulate a real-world node, a Docker container is used to deploy each node in the Fabric network. We set up our experimental environment on a commercial laptop with an Intel Core i7 processor and 16GB of operating memory. The experiments included evaluating the time cost of constructing the block structure (MHT), the cost of generating the proof for verifying data integrity (i.e., the size of the structure and the generation time), and the time cost of completing the verification using evidence. In the experiments, we used three types of secure hash algorithms (SHA), namely 128-bit, 256-bit, and 512-bit hashing [44]. Moreover, we tested the efficiency of our Tier Skip Chain.

The dataset used for the experiment is an open-source collection known as DataCo [45]. This dataset encompasses



(a) vDID Generation Time

(b) vDID Size

Fig. 5. vDID Construction Performance

comprehensive supply chain information from large commercial enterprises in real-world manufacturing environments. It covers the entire lifecycle of a product, from the procurement of raw materials through production processes to the final distribution stages, which contains specific data of some products and their materials, including their inter-compositional relationships.

B. vDID Construction Performance

Figure 5(a) and Figure 5(b) show the construction performance of MHT. Specifically, we set the number of DID documents (corresponding to products and materials) at 1K, 20K, 40K, 60K, 80K, 100K, 120K, 140K and 160K to evaluate the time cost of vDID construction under different conditions. As shown in Figure 5, the construction time of vDID increases linearly as the number of documents increases. We also conduct comparative experiments using different encryption methods to process the data. Specifically, we study the effect on the performance of the vDID construction with three different lengths of hash functions (Figure 5a). The experimental results show that the time cost spent on constructing the vDID remains almost unchanged for the 128-bit and 256-bit hash functions. However, if the encryption method with 512-bit hash function is used, the construction time for vDID almost doubles for large number of data records. With respect to this issue, we claim that stronger encryption provides higher security but requires more computational resources, increasing processing time and energy consumption. In contrast, weaker encryption improves efficiency but may be vulnerable to brute-force attacks as computing power increases over time.

Figure 5(b) shows the relationship between vDID size and the number of data records. Across all three hash lengths (128-bit, 256-bit, and 512-bit), the vDID size increases linearly with the number of data records. Specifically, when the number of data records is 4×10^4, the vDID size is approximately 0.07MB, 0.13MB, and 0.25MB for the 128-bit, 256-bit, and 512-bit hashes, respectively. When the number of data records reaches 16×10^4, the vDID sizes grow to about 0.65MB, 1.00MB, and 2.05MB.

C. Proof Performance

Figure 6(a), Figure 6(b) and Figure 6(c) show the the proof performance of the system. Specifically, we study the impact of the number of data records on three metrics, namely (i) proof generation time, (ii) proof verification time, and (iii) proof size.

Figure 6(a) depicts the proof generation time as the number of data records increases. For all three hash lengths, the proof generation time shows an approximately linear growth from 0 to 6×10^4 data records, with similar performance across the hash lengths during this range. However, from 6×10^4 to 16×10^4 data records, the proof generation time increases further. Specifically, the 128-bit hash sees an increase from about 3ms to 5.5ms, the 256-bit hash from around 3ms to 5ms, and the 512-bit hash from roughly 3.5ms to over 7.5ms.

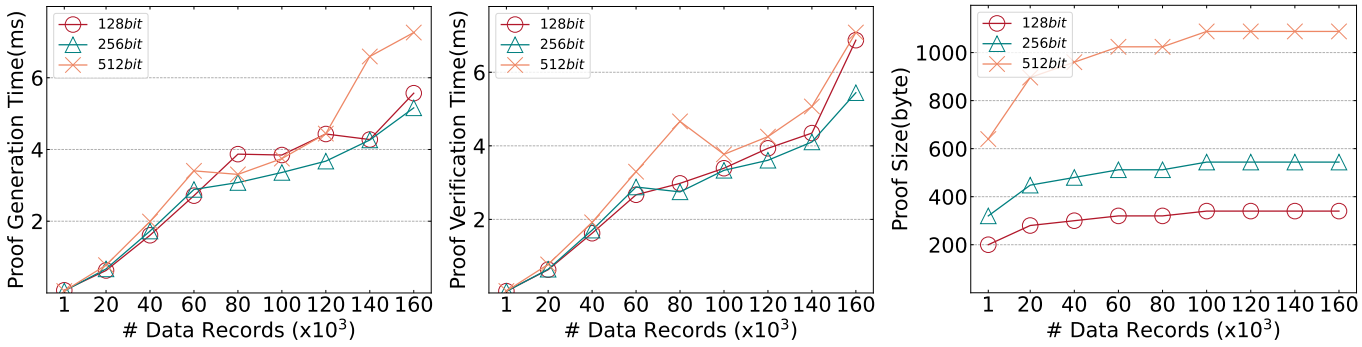
Figure 6(b) shows the proof verification time as the number of data records grows. Across all hash lengths, the verification time gradually increases with the number of data records. For instance, with the 128-bit hash, the verification time rises from approximately 1.8ms at 4×10^4 records to about 7.0ms at 16×10^4 records. Similarly, the 256-bit hash verification time increases from around 1.8ms to 5.5ms, while the 512-bit hash rises from approximately 1.9ms to 7.0ms.

Figure 6(c) illustrates the relationship between proof size and the number of data records. From 0 to 10×10^4 records, the proof size grows slowly for all three hash lengths, and it stabilizes beyond 10×10^4 records. Specifically, the proof size stabilizes at around 200 bytes for the 128-bit hash, 400 bytes for the 256-bit hash, and 1000 bytes for the 512-bit hash.

D. Query Performance of Tier Skip Chain

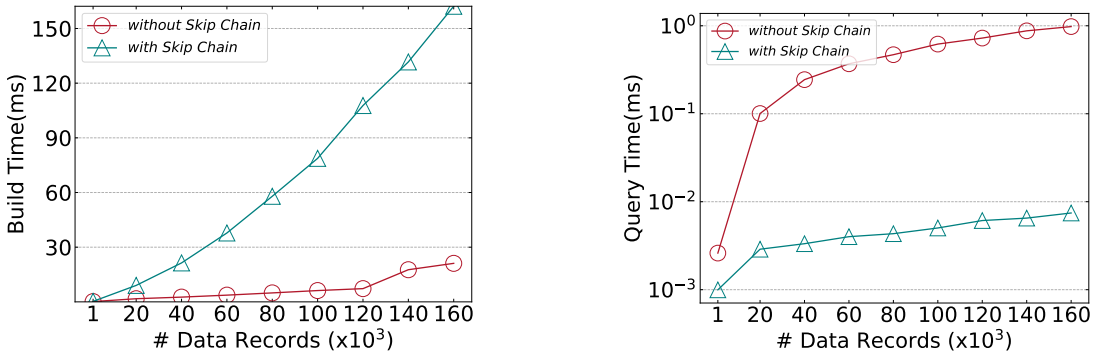
Figure 7(a) compares the construction time with and without the hierarchical skip chain as the number of data records increases. Without the skip chain, the construction time keeps almost unchanged with different number of documents. In contrast, the build time of Tier Skip Chain becomes larger as the number of data records increases. The reason is that, the building process requires to traverse all the DIDs and construct the skip links among the entities.

Figure 7(b) compares the query time with and without the hierarchical skip chain as the number of data records grows.



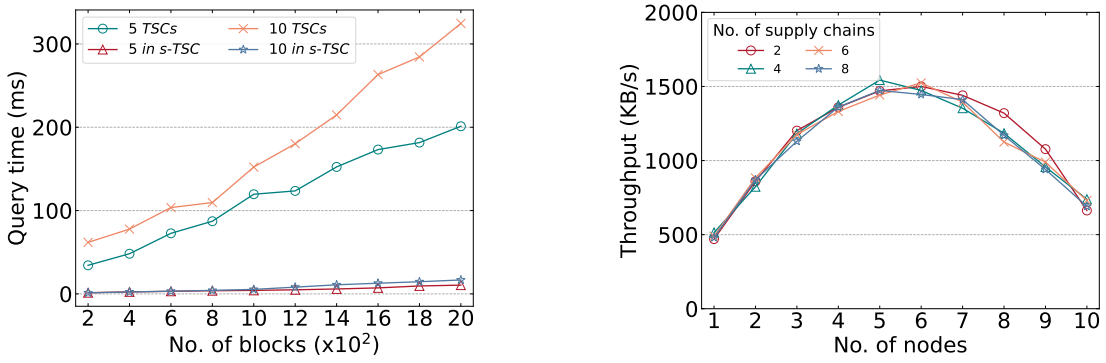
(a) Proof Generation Time (b) Proof Verification Time (c) Proof Size

Fig. 6. Verification Performance



(a) Skip Chain Build Time (b) Skip Chain Query Time

Fig. 7. Query Performance of Tier Skip Chain



(a) Optimization Query Time (b) System Throughput

Fig. 8. Optimization Query Time and System Throughput

Without the skip chain, the query time increases linearly, from approximately 0.24ms at 4×10^4 records to about 0.98ms at 16×10^4 records. However, with the skip chain, the query time remains consistently below 0.1ms, showing almost no dependence on the number of data records.

E. Query Optimization Time and System Throughput

Figure 8(a) and Figure 8(b) show the effectiveness of the optimization proposed in Section IV-E and the system throughput, respectively. Figure 8(a) illustrates the query times for multiple supply chains under varying conditions. The experiment examines four scenarios: 5 supply chains with and without optimization (*5 in s-TSC* vs. *5 in TSCs*), and 10 supply chains with and without optimization (*10 in s-TSC* vs. *10 in TSCs*). In the scenarios without optimization, the query time increases significantly as the number of blocks grows, reaching 200 ms for 5 supply chains and 320 ms for 10 supply chains when the block count hits 2,000. This indicates a clear scalability issue as the system struggles to maintain efficiency with more data. Conversely, when optimization is applied through integrative locating layers, the query times for both 5 and 10 supply chains exhibit only a slight increase

as the block number rises. Remarkably, even at 2,000 blocks, the query time remains under 20 ms for both scenarios. This demonstrates the effectiveness of the optimization strategy in maintaining low query times and suggests that the integrative locating layers significantly enhance the system's scalability and performance.

We conducted experiments to evaluate the performance of our system by measuring throughput with varying numbers of blockchain nodes and supply chains. As shown in Figure 8(b), four lines represent the throughput for 2, 4, 6, and 8 supply chains. These lines overlap significantly, indicating that the number of supply chains has minimal impact on throughput within the tested range of node numbers. Our results indicate that system performance is influenced by the number of blockchain nodes. Throughput initially increases with more nodes, starting at approximately 500 KB/s with a single node and peaking at around 1500 KB/s with 5 to 6 nodes. Beyond this point, throughput declines as node numbers increase. This decline is possibly due to rising communication overhead, as each node must synchronize with others, increasing network latency and reducing throughput. An excessive number of nodes can overwhelm the system's capacity. Thus, blockchain systems must balance decentralization and efficiency. In our model, 5 to 6 nodes appear to provide an optimal balance.

VI. CONCLUSION

This paper presents a blockchain-based verifiable decentralized identity system for smart manufacturing. We first designed a decentralized architecture based on blockchain and a storage structure for DID documents, which can effectively reduce the overhead of system storage. Second, we propose a multi-block hopping structure constructed based on associative DIDs, which effectively improves the query efficiency of the system. Then, we developed a verification scheme that allows verifiers to use this DID Document storage structure for more efficient and accurate verification. Finally, we implemented the system and demonstrated through experimental evaluation that our system can achieve more efficient data validation with reduced storage overhead. Future work may include investigating more efficient indexing or hierarchical storage mechanisms to further optimize DID document retrieval in a large-scale manufacturing environment. Moreover, it is possible to incorporate machine learning or AI-driven anomaly detection to identify and prevent fraudulent identity claims in real time.

ACKNOWLEDGMENTS

This work was supported in part by Natural Science Foundation of Zhejiang University of Science and Technology (No. 2025QN023), the Hong Kong Research Grants Council General Research Fund (No. 12202922, 15238724), the Shenzhen Science and Technology Program (No. JCYJ20230807140412025), the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (2023C03195), the Graduate Course Development Project of Zhejiang University of Science and Technology (No. 2024yjskj03), the Ideological and Political Education Teaching Research Project of Zhejiang University of Science and Technology (No. 2024-ksj3), the Key R&D Program of Zhejiang Province (No. 2023C01217), the Natural Science Foundation of Zhejiang Province (No. LQ24F020040), and Yangtze River Delta Science and Technology Innovation Community Joint Research Project (No. 2022CSJGG1000/2023ZY1068).

REFERENCES

[1] F. A. Alabdulwahhab, "Web 3.0: the decentralized web blockchain networks and protocol innovation," in *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*. IEEE, 2018, pp. 1–4.

[2] J. Xu, S. Wang, A. Zhou, and F. Yang, "Edgence: A blockchain-enabled edge-computing platform for intelligent iot-based dapps," *China Communications*, vol. 17, no. 4, pp. 78–87, 2020.

[3] B. Liu, P. Szalachowski, and J. Zhou, "A first look into defi oracles," in *2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*. IEEE, 2021, pp. 39–48.

[4] S. Kumar, A. K. Bharti, and R. Amin, "Decentralized secure storage of medical records using blockchain and ipfs: A comparative analysis with future directions," *Security and Privacy*, vol. 4, no. 5, p. e162, 2021.

[5] D. Trautwein, A. Raman, G. Tyson, I. Castro, W. Scott, M. Schubotz, B. Gipp, and Y. Psaras, "Design and evaluation of ipfs: a storage layer for the decentralized web," in *Proceedings of the ACM SIGCOMM 2022 Conference*, 2022, pp. 739–752.

[6] R. Gu, S. Wang, H. Dai *et al.*, "Fluid-shuttle: Efficient cloud data transmission based on serverless computing compression," *IEEE/ACM Transactions on Networking*, 2024.

[7] R. Gu, K. Zhang, Z. Xu, Y. Che *et al.*, "Fluid: Dataset abstraction and elastic acceleration for cloud-native deep learning training jobs," in *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2022, pp. 2182–2195.

[8] R. Gu, Y. Chen, S. Liu, H. Dai, G. Chen *et al.*, "Liquid: Intelligent resource estimation and network-efficient scheduling for deep learning jobs on distributed gpu clusters," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2808–2820, 2021.

[9] X. Li, T. Jing, R. Li, H. Li, X. Wang, and D. Shen, "BDRA: Blockchain and decentralized identifiers assisted secure registration and authentication for vanets," *IEEE Internet of Things Journal*, 2022.

[10] S. Huh, M. Shim, J. Lee, S. Woo, H. Kim, and H. Lee, "Did we miss anything?: Towards privacy-preserving decentralized id architecture," *IEEE Transactions on Dependable and Secure Computing*, 2023.

[11] W. Hussain, J. M. Merigó, H. Gao, A. M. Alkalbani, and F. A. Rabhi, "Integrated ahp-iowa, powa framework for ideal cloud provider selection and optimum resource management," *IEEE Transactions on Services Computing*, vol. 16, no. 1, pp. 370–382, 2023.

[12] N. Fotiou, Y. Thomas, V. A. Siris, G. Xylomenos, and G. C. Polyzos, "Self-verifiable content using decentralized identifiers," *Computer Networks*, vol. 230, p. 109799, 2023.

[13] Z. Peng, J. Deng, S. Gao, H. Cui, and B. Xiao, "vDID: Blockchain-enabled verifiable decentralized identity management for web 3.0," in *Proc. of the IEEE/ACM International Symposium on Quality of Service (IWQoS)*, 2024.

[14] M. M. Islam, M. K. Islam, M. Shahjalal, M. Z. Chowdhury, and Y. M. Jang, "A low-cost cross-border payment system based on auditable cryptocurrency with consortium blockchain: Joint digital currency," *IEEE Transactions on Services Computing*, 2022.

[15] E. Bandara, X. Liang, P. Foytik, S. Shetty, C. Hall, D. Bowden, N. Ranasinghe, and K. De Zoysa, "A blockchain empowered and privacy preserving digital contact tracing platform," *Information processing & management*, vol. 58, no. 4, p. 102572, 2021.

[16] D. Ressi, R. Romanello, C. Piazza, and S. Rossi, "Ai-enhanced blockchain technology: A review of advancements and opportunities," *Journal of Network and Computer Applications*, p. 103858, 2024.

[17] Y. Li, Y. Lu, X. Yang *et al.*, "Blockchain-empowered multi-skilled crowdsourcing for mobile web 3.0," *Computer Communications*, vol. 232, p. 108037, 2025.

[18] P. Li, Z. Xiao, X. Wang, K. Huang, Y. Huang, and H. Gao, "Eptask: Deep reinforcement learning based energy-efficient and priority-aware task scheduling for dynamic vehicular edge computing," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 1830–1846, 2024.

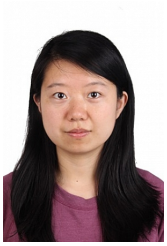
[19] H. Wu, Z. Peng, S. Guo, Y. Yang, and B. Xiao, "VQL: Efficient and verifiable cloud query services for blockchain systems," *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, vol. 33, no. 6, pp. 1393–1406, 2021.
[20] H. Wang, C. Xu, C. Zhang, J. Xu *et al.*, "vChain+: Optimizing verifiable blockchain boolean range queries," in *Proc. of the IEEE International Conference on Data Engineering (ICDE)*, 2022.
[21] H. Wu, Y. Tang, Z. Shen *et al.*, "TELEX: Two-level learned index for rich queries on enclave-based blockchain systems," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2025.
[22] R. Xiong, W. Ren, X. Hao, J. He, and K.-K. R. Choo, "BDIM: A blockchain-based decentralized identity management scheme for large scale internet of things," *IEEE Internet of Things Journal*, 2023.
[23] J. Yin, Y. Xiao, Q. Pei, Y. Ju, L. Liu, M. Xiao, and C. Wu, "SmartDID: a novel privacy-preserving identity based on blockchain for iot," *IEEE Internet of Things Journal*, vol. 10, no. 8, pp. 6718–6732, 2022.
[24] Y. Liu, Q. Lu, S. Chen, Q. Qu, H. O'Connor, K.-K. R. Choo, and H. Zhang, "Capability-based iot access control using blockchain," *Digital Communications and Networks*, vol. 7, no. 4, pp. 463–469, 2021.
[25] T. M. Fernández-Caramés and P. Fraga-Lamas, "A review on the use of blockchain for the internet of things," *Ieee Access*, vol. 6, pp. 32 979–33 001, 2018.
[26] M. Hussain, W. Javed, O. Hakeem, A. Yousafzai, A. Younas, M. J. Awan, H. Nobanee, and A. M. Zain, "Blockchain-based iot devices in supply chain management: a systematic literature review," *Sustainability*, vol. 13, no. 24, p. 13646, 2021.
[27] H. Gao, D. Fang, J. Xiao, W. Hussain, and J. Y. Kim, "Camrl: A joint method of channel attention and multidimensional regression loss for 3d object detection in automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8831–8845, 2022.
[28] A. Raja Santhi and P. Muthuswamy, "Influence of blockchain technology in manufacturing supply chain and logistics," *Logistics*, vol. 6, no. 1, p. 15, 2022.
[29] M. A. Agi and A. K. Jha, "Blockchain technology in the supply chain: An integrated theoretical perspective of organizational adoption," *International Journal of Production Economics*, vol. 247, p. 108458, 2022.
[30] Q. Wen, Y. Gao, Z. Chen, and D. Wu, "A blockchain-based data sharing scheme in the supply chain by iiot," in *2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*. IEEE, 2019, pp. 695–700.
[31] W. Nikolakis, L. John, and H. Krishnan, "How blockchain can shape sustainable global value chains: An evidence, verifiability, and enforceability (eve) framework," *Sustainability*, vol. 10, no. 11, p. 3926, 2018.
[32] N. Rožman, J. Diaci, and M. Corn, "Scalable framework for blockchain-based shared manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102139, 2021.
[33] R. Cole, M. Stevenson, and J. Aitken, "Blockchain technology: implications for operations and supply chain management," *Supply chain management: An international journal*, vol. 24, no. 4, pp. 469–483, 2019.
[34] G. Iyengar, F. Saleh, J. Sethuraman, and W. Wang, "Blockchain adoption in a supply chain with manufacturer market power," *Management Science*, vol. 70, no. 9, pp. 6158–6178, 2024.
[35] J. Leng, D. Yan, Q. Liu, K. Xu, J. L. Zhao, R. Shi, L. Wei, D. Zhang, and X. Chen, "Manuchain: Combining permissioned blockchain with a holistic optimization model as bi-level intelligence for smart manufacturing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 182–192, 2019.
[36] S. E. Chang, Y.-C. Chen, and M.-F. Lu, "Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process," *Technological Forecasting and Social Change*, vol. 144, pp. 1–11, 2019.
[37] S. Geiger, D. Schall, S. Meixner, and A. Egger, "Process traceability in distributed manufacturing using blockchains," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, 2019, pp. 417–420.
[38] Y. Zhang, P. Zhang, F. Tao, Y. Liu, and Y. Zuo, "Consensus aware manufacturing service collaboration optimization under blockchain based industrial internet platform," *Computers & Industrial Engineering*, vol. 135, pp. 1025–1035, 2019.
[39] Y. Wu and Y. Zhang, "An integrated framework for blockchain-enabled supply chain trust management towards smart manufacturing," *Advanced Engineering Informatics*, vol. 51, p. 101522, 2022.
[40] Y. Zhang, X. Xu, A. Liu, Q. Lu, L. Xu, and F. Tao, "Blockchain-based trust mechanism for iot-based smart manufacturing system," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1386–1394, 2019.
[41] J. Leng, S. Ye, M. Zhou, J. L. Zhao, Q. Liu, W. Guo, W. Cao, and L. Fu, "Blockchain-secured smart manufacturing in industry 4.0: A survey," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 237–252, 2020.
[42] C. Cai, Y. Zheng, A. Zhou, and C. Wang, "Building a secure knowledge marketplace over crowdsensed data streams," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 6, pp. 2601–2616, 2019.
[43] H. Wu, Z. Peng, J. Xiao *et al.*, "HeX: Encrypted rich queries with forward and backward privacy using trusted hardware," *IEEE Transactions on Dependable and Secure Computing (TDSC)*, 2025.
[44] M. Parmar and H. J. Kaur, "Comparative analysis of secured hash algorithms for blockchain technology and internet of things," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 3, 2021.
[45] Fabian Constante, Fernando Silva, and António Pereira. ([2019-3-13]) Dataco smart supply chain for big data analysis. [EB/OL]. [Online]. Available: https://data.mendeley.com/datasets/8gx2fvg2k6/5

Wenjian Xu received the Ph.D degree in computer science from the Hong Kong Polytechnic University in 2018. He is currently an associate professor with the School of Information and Electronic Engineering, Zhejiang University of Science and Technology. He has published more than 18 articles in toptier conferences and journals, such as SIGMOD, ICDE, TKDE, TSC. His research interests include big data analytics, cloud computing, and inference optimization of large language models. He is also a member of the China Computer Federation (CCF). He has served as a Reviewer for the IEEE Transaction on Knowledge and Data Engineering, Human-centric Computing and Information Sciences, and International Conference on Big Data.

Jiamin Deng is currently pursuing the Ph.D. degree in The Hong Kong Polytechnic University. She received the B.S. and the M.S. degrees from Chongqing University in 2019 and 2022, respectively. Her research interests include blockchain, distributed identity, and data security.

Jialong Yu is currently pursuing the M.S. degree in the University of Glasgow. He received the B.S. degree from the University of Nottingham in 2023. His research interests include distributed computing, urban big data, and smart cities.

Shanghui Mao is a graduate student pursuing a Master's degree in Computer Technology at Hangzhou Dianzi University (HDU). He completed both his undergraduate and current graduate studies at HDU, where he developed a solid foundation in computer science and technology. During his academic journey, he has cultivated an interest in research and problem-solving, particularly in areas related to computer technology. Currently, he is focused on improving his research skills and exploring new knowledge to broaden his expertise. His academic background and dedication to growth position him well for upcoming challenges in the field.



Youhuizi Li received the B.E. degree from Xidian University, Xi'an, China, in 2010, and the Ph.D. degree from Wayne State University, Detroit, MI, USA, in 2016, both in computer science. She is currently an Associate Professor with the Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education, Hangzhou, China. She is also with the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou. Her research interests include edge computing, privacy protection, and energy efficient systems. Dr. Li was an Editor for Sustainable Computing: Informatics and Systems and a Reviewer for IEEE Internet of Things Journal and Mobile Networks and Applications.

Zhe Peng is currently a research assistant professor in the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University. He received the B.S. degree from Northwestern Polytechnical University, the M.S. degree from University of Science and Technology of China, and the Ph.D. degree from The Hong Kong Polytechnic University. He was a visiting scholar in the Department of Electrical and Computer Engineering, Stony Brook University. His research interests include blockchain, distributed computing, internet of things, big data analysis and management, and data security.

Bin Xiao (Fellow IEEE) received the B.Sc. and M.Sc. degrees from Fudan University and the Ph.D. degree from The University of Texas at Dallas. He is currently a Full Professor with the Department of Computing, The Hong Kong Polytechnic University. He has published more than 250 technical papers in top-tier conferences and journals, including IEEE S&P, ACM CCS, IEEE INFOCOM, IEEE ICDCS, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING (IEEE TDSC), IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY (IEEE TIFS), IEEE/ACM TRANSACTIONS ON NETWORKING (IEEE/ACM ToN), IEEE TRANSACTIONS ON MOBILE COMPUTING (IEEE TMC), IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS (IEEE TPDS), IEEE TRANSACTIONS ON COMPUTERS (IEEE TC), IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (IEEE TNNLS), and IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (IEEE JSAC). His research interests include information security, data privacy, and blockchain systems. Currently, he serves as an Associate Editor for IEEE TRANSACTIONS ON CLOUD COMPUTING and IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING. He served as an Associate Editor for Journal of Parallel and Distributed Computing (Elsevier) from 2016 to 2021 and IEEE INTERNET OF THINGS JOURNAL from 2020 to 2023. He is the Chair of the IEEE ComSoc CISTC Committee and IEEE ComSoc Distinguished Lecturer. He has been the Track Co-Chair of IEEE ICDCS 2022, the Symposium Track Co-Chair of IEEE ICC 2020, ICC 2018, and Globecom 2017, and the General Chair of IEEE SECON 2018.