electronics

Review

Hyperledger for IoT: A Review of Reconstruction Diagrams Perspective

Zeqi Leng [1], Kunhao Wang [1,*], Yuefeng Zheng [1], Xiangyu Yin [2] and Tingting Ding [3]

[1] College of Computer Science, Jilin Normal University, Siping 136000, China; lenglinyuan27@outlook.com (Z.L.); honest_zyf@hotmail.com (Y.Z.)
[2] College of Foreign Languages, Jilin University, Changchun 130000, China; 15734445860@163.com
[3] Siping Vocational and Technical Education Center, Siping 136000, China; dingtingting0822@outlook.com
* Correspondence: wkhyoyo@jlnu.edu.cn

Abstract: The convergence of blockchain with the internet of things (IoT) attracted widespread attention. Blockchain mainly solved the problem of secure storage and trusted transactions. The convergence of these two emerging technologies enhanced the security of smart services. However, there were some technical barriers to the deployment of practical IoT systems. In order to further promote the popularity and application of blockchain in the IoT, Hyperledger became the ideal technology to overcome these obstacles. In recent years, the mainstream application fields of IoT tried to carry out integration with Hyperledger to achieve high security, fine-grained privacy protection, real-time data flow, robustness, and other business requirements. However, there was a lack of literature review on this topic. This study obtained the latest related literature of Hyperledger in IoT from Web of Science, Wordlib, and EBSCO databases. To demonstrate more intuitive differences and provide a technology convergence process, this study proposes a reconstruction diagram analysis method. Reconstruction is the process of fusing the core design and the original architecture diagram in the literature and reconstructing the diagram so that it can show the core ideas of the literature. This approach aims to visualize the core ideas of the literature. Finally, this paper prospected and concluded by proposing four directions for future work, including a low-energy consensus algorithm, intelligent transaction validation, mixed on-chain and off-chain storage, and customized incentives.

Keywords: internet of things (IoT); hyperledger; blockchain

Citation: Leng, Z.; Wang, K.; Zheng, Y.; Yin, X.; Ding, T. Hyperledger for IoT: A Review of Reconstruction Diagrams Perspective. *Electronics* **2022**, *11*, 2200. https://doi.org/10.3390/electronics11142200

Received: 16 May 2022
Accepted: 9 July 2022
Published: 13 July 2022

1. Introduction

With the rapid development of smart devices and high-speed networks, IoT gained wide acceptance and popularity as a major standard for resource-constrained low-power networks [1]. However some technical vulnerabilities and flaws hinder the development of IoT, and these hindrances are mainly focused on data security issues and network congestion caused by centralized servers. Blockchain breaks through the traditional sense of trust and makes every decision in the network transform from a few nodes to a joint decision of all nodes, increasing the transparency of transactions and storing trust. As a result, blockchain-based IoT research is widely explored. This previous research solved the problems of secure storage of IoT data and data integrity verification to some extent. However, the shortcomings of blockchain technology in privacy protection, load balancing, and network latency led to a state of the art that cannot meet the standards of commercial applications, especially for IoT projects. According to the current stage of technology development, these landing obstacles are mainly focused on the lack of privacy protection, inefficient transactions, high latency, and lack of flexibility.

In recent years, the emergence of Hyperledger provided the possibility of solving the above problems. In addition to having the general characteristics of blockchain, Hyperledger achieves new empowerment in four aspects: security, interoperability, consensus,

and performance. In terms of security, Hyperledger is designed with fine-grained access control and private data management, giving privacy protection mechanisms and ledger isolation among enterprises, providing infrastructure for enterprise-level solutions. To protect the interests of consumers, Hyperledger supports autonomy of access control, which enables the design and implementation of a number of consumer-centric applications. The highly modular structure [2] allows systems developed based on Hyperledger to have a single point of failure of a module or component without affecting the overall operation of the system. This feature also allows Hyperledger to become integrated with various systems faster. These pluggable components enable Hyperledger to meet rich business logic in the same distributed network. In terms of interoperability: Hyperledger develops a peer-to-peer authentication system that provides the infrastructure for consumers or enterprises to perform cross-chain, cross-layer, or cross-system operations through a portable electronic identity. In terms of consensus, pluggable consensus mechanisms enable different businesses to reach consensus in the same distributed network. The consensus algorithms supported by the current Hyperledger cover energy saving, reputation, proof of work, proof of time, authority, fault tolerance, and authorized interests [3]. Thus, the applicability of Hyperledger is significantly enhanced. In terms of performance, decisions in the network are decided by a certain number of nodes instead of being decided by all nodes together, which significantly reduces arithmetic costs and ensures trust. The Hyperledger network can be dynamic, offering the possibility of connecting a large number of portable devices. Hyperledger subdivides the nodes that process transactions into four roles [4], each with a different function, and allows developers to adjust the deployment of nodes based on the network load. In addition, decoupled sequencing processing can address a certain level of network congestion and latency.

Hyperledger-based IoT research is growing year by year, yet there is no review on Hyperledger in the IoT domain. In addition, most of the blockchain reviews are text-based and supplemented by diagrams. The concise language shows the latest application progress and trends for researchers, but it cannot visually show researchers the design ideas of these studies, and it is difficult for researchers to get enough guidelines and core ideas for technology integration from the condensed language.

To better solve the above problems, this study proposes a new analysis method: the reconstruction diagrams method. Reconstruction refers to the process of reconstructing a diagram that can show the core ideas of the literature by fusing the core design and the original architecture diagram in the literature with the core content of Hyperledger and the literature as the main research object. Therefore, the main advantage of the reconstruction diagram analysis method is twofold: first, it shows the research progress and core idea in the most intuitive way; second, it adds more design details, visualizes the core design in the literature, and uses the most intuitive way to highlight the differences between different studies. This study summarizes the progress of Hyperledger applications in the IoT based on the reconstruction diagrams perspective, and provides researchers with the main design flow and guidelines for Hyperledger technology in the IoT. The main contributions of this study are as follows.

(1) An analytical approach to reconstruction diagrams is proposed. This study considers two cases: The first one is to construct reconstruction diagrams by extracting the core design in the literature for the literature without architecture diagrams, and to restore the core design ideas of the authors to the maximum extent. The second one is for literature with existing architecture diagrams. The original architecture diagram is reconstructed on the basis of the core design of the literature, adding more design details.

(2) It shows the latest progress of Hyperledger application in the IoT. Since the research on Hyperledger in IoT is scattered at this stage, this study summarizes 52 pieces of literature in terms of application domains. These application areas are related to IoT data security, smart fisheries and agriculture, smart city monitoring, smart toys and IoT games, smart fitness, smart traffic, smart grid, smart building projects, and smart energy (energy saving direction). The rest of this paper is organized as follows: Section 2 presents an overview

of Hyperledger and its enabling technologies. Section 3 introduces the concept of the IoT, and the point of convergence between Hyperledger and IoT applications. Section 4 provides a comprehensive review of existing applications of Hyperledger in the IoT, and visualizes the differences between the scenarios through the perspective of reconstruction diagrams. Section 5 provides four potential research directions. Finally, the paper concludes in Section 6.

2. Introduction to Hyperledger

Hyperledger is committed to developing an enterprise-grade standard blockchain. At the official conceptual level, Hyperledger is a "greenhouse" system, where all technology is developed by the community. It provides an open source and secure collaborative environment for users, developers, and vendors across all domains. As a result, Hyperledger encourages interoperability between participants in similar domains, each of whom communicates to obtain the necessary information. This effective collaboration greatly reduces the duplication of work for each participant, allowing participants to have more energy to incubate new ideas. To improve code quality, the Technical Steering Committee (TSC) regularly checks the community's code and projects, and substandard code and projects are discarded. In addition, Hyperledger encourages the achievement of specialization [5], i.e., more people focus on fewer tasks and increase the level of expertise of the participants. Developing the specialization of the participants also helps to promote uniformity of intellectual property rights, and any participant contributing to the Hyperledger community does not have to worry about hidden legal issues.

A generic architecture for Hyperledger should have nine components: a consensus layer, a contract layer, a communication layer, a data storage module, an encryption module, an identity services module, a policy services module, application programming interfaces (APIs), and an interoperability module [6]. These components form a highly modular structure in which the failure of any one component does not affect the overall operation.

As one of the largest open source projects, Hyperledger currently has 18 top projects (including one that was phased out). These top projects provide key technologies for Hyperledger and enable Hyperledger to be widely used in various fields. On the technology side, Hyperledger covers areas such as cross-system authentication, permission control, multi-channel (multi-chain) platform, visualization interface, mobile application, benchmarking, encryption library, Ethernet client, and its business logic development. In terms of applications, Hyperledger is used in the mainstream fields of the internet of things, digital healthcare, supply chain traceability, finance, digital evidence, artificial intelligence, etc.

Hyperledger divides the current top projects into four categories, including distributed ledgers, domain specific, libraries, and tools. Each project contributed to Hyperledger requires regular maintenance by the developers, which means that in addition to maintaining the normal operation of the project, the developers also have to solve problems for members who want to participate in the project in a timely manner. TSC regularly reviews the maintenance status of each project and decides whether the project will move to the next stage. When a project is no longer recommended for use, it will be abandoned by the community after 6 months. However, the abandoned project information and part of the code remain in the community. Each project in Hyperledger must possess the five features of being modular, highly secure, interoperable, cryptocurrencyagnostic, and complete with APIs. Modular components are suitable for developing distributed solutions with different requirements, and high security ensures enterprise-grade blockchain implementation. Interoperability and rich APIs give large enterprise distributed networks easy information interaction.

In general, all projects in Hyperledger go through six phases (status): proposal, incubation, gradated (active), dormant, deprecated, and end of life. The status of each project is dynamic and is jointly determined by the maintainer of the project and the TSC with multiple reviews. At this stage, the top projects in Hyperledger have only two statuses, graduated and incubation. The projects in graduated status are the most active projects with the most members and the most contributed code. Due to constant updates, active

projects provide a more mature technology and infrastructure for Hyperledger. Based on the information provided on the official website, this paper dissected the core architecture and innovative design of the project in graduated status.

2.1. Fabric

Fabric is the cornerstone of Hyperledger; its innovative design enables Hyperledger to be widely used in various fields [3]. Fabric pioneered the introduction of the authority mechanism, giving the possibility of confidential transactions and ledger isolation in various industries. Fabric's architecture consists of membership services, certificate authorities (CA), nodes, peers, and four types of components. Membership services provide digital certificates for blockchain nodes, CAs provide identity certificates for all nodes in the network, which complete transactions with private and public keys, nodes consist of nodes that are allowed to join the network, and peers are roles that perform different tasks in the blockchain network.

2.2. Sawtooth

The innovative design of Sawtooth is to simplify the development process of the blockchain application by separating the central system from the application layer [7]. Each Sawtooth node consists of a fixed component validator, and a possible components transaction processor, REST (representational state transfer) API, and client [8]. In the Sawtooth network, the initial node sends broadcast packets to get nearby nodes, and neighboring nodes can join the network according to the rules and broadcast their neighbor's one-hop-away node. As long as there is a response, the node can join the network.

Sawtooth architecture has five core components, including a peer-to-peer network, distributed log, state machine/smart contract logic layer, distributed state storage, and consensus algorithm. The peer-to-peer network allows nodes to communicate via TCP, including information about blocks, peers, etc. [9]. The Sawtooth network broadcasts transactions via gossip protocol. The distributed log includes an ordered list of transactions, which is sorted by nodes according to the consensus algorithm. Sawtooth extends the functionality of smart contracts by treating them as state machines or transaction processors. In the smart contract logic layer, Sawtooth uses radix Merkle. The consensus component provides a consensus interface that allows various consensus algorithms.

2.3. Iroha

Iroha also provides a distributed framework that is designed to feature privilege management, fault tolerance, and performance efficiency [10]. Compared to other platforms, Iroha requires authorization to read and write data in addition to the authorization required for nodes to join the network. Iroha allows rich built-in commands for simpler asset management, unlike other platforms that require predefined assets [11]. It is designed with a fault-tolerant consensus algorithm, Crash, which allows Iroha to have lower latency.

Iroha architecture has 11 components, including Torii, MST processor, peer communication service (PCS), ordering gate, ordering service, verified proposal creator (VPC), block creator, block consensus (YAC), synchronizer, Ametsuchi blockstore, and world state view (WSV) [12]. In a typical Iroha transaction, client-initiated transactions are received by Torii and forwarded to the MST processor, which typically has two tasks, including forwarding transactions to the PCS and receiving transaction messages (multiple signatures) from other peers. The ordering gate verifies the stateless transactions with other peers, and the ordering service in the peer creates a transaction proposal (each node contains an ordering service) and verifies that the stateless transaction passes the first verification. The VPC performs state verification of the transaction, and the block creator creates new blocks and sends them to the YAC to perform consensus. The YAC forwards the final message to multiple peers. The synchronizer is responsible for downloading blocks from the block store and adding the missing blocks from the peers to the peers. At this point, the Iroha network updates the WSV.

2.4. Indy

The innovative design of Indy lies in a decentralized identity system [13]. The core feature of this authentication is the self-sovereign identity [14]. This means that once an identity is established, it cannot be revoked, selected, or associated by any institution or person without the permission of the identity owner. There are only two types of nodes in the Indy network, including verification nodes (which are few in number) and observer nodes (which are many in number). Among them, authentication nodes are responsible for processing write requests and participating in consensus. Observer nodes are responsible for reading requests and have the opportunity to become verifying nodes depending on their reputation level. Indy can provide users with portable proof of identity and does not require centralized authentication by a third party.

2.5. Aries

As the only technology of the six active projects that is not a distributed ledger platform, Aries is a way to provide secure communications for decentralized identity management and verifiable credentials. Aries has four core components, including agents, DID communications, protocols, and key management [15]. Agents provide trusted agents for self-sovereign identity authentication. Specifically, trusted agents help people or organizations send bytes and store data directly. The user downloads or writes the appropriate agent according to the requirements of the agent, such as IoT agents, cloud agents, protocols, scale, and privacy requirements. DID communications is meant to provide information exchange for multiple trusted agents [16]. It is based on decentralized protocols, and its main paradigms are message-based, asynchronous (request–response messages), and simplex. Key management provides a distributed key management system that uses three types of keys, including master keys, key encryption keys, and data keys. The distributed key management system allows any identity owner to perform network connectivity, key exchange, and recovery without relying on any organization, free from the central failures of third-party organizations.

The emergence of Aries facilitates the implementation of decentralized authentication, and peer-to-peer certificate authentication will eradicate the surveillance economy. This authentication method is highly portable and applicable, allowing users to store their proof of employment, or other identification, in a wallet and decide which part of the information can be publicly queried.

2.6. Besu

Besu is an enterprise class Ethereum platform [17]. Besu has seven core modules, including Ethereum virtual machine (EVM), P2P network, storage, permissioning, privacy, user-facing API, and monitoring.

In terms of privacy, Besu ensures private interactions through Tessera nodes [18]. For example, if a private transaction is sent by Bob, this transaction must first be passed to Bob's Tessera node and complete the information exchange with Alice's Tessera node (the Tessera node involved in the transaction) before being passed to Alice. For better enterprise orientation, permissioning enables node permissions and account permissions so that only specific Storage will store the blockchain and world state, where world state includes account state, account storage, and code storage. Besu provides users with a monitoring interface to demonitor nodes and networks.

Besu supports two node types, including full nodes and archive nodes. Full nodes store only the current block state, ensuring the current up-to-date state. Archive nodes are responsible for storing all the historical states of the blocks since the creation of the world, in addition to the latest state. In addition, Besu provides three APIs for users, including JSON-RPC based on HTTP/WebSockets, RPC publish/subscribe based on WebRocket, and GraphQL based on HTTP.

Besu is compatible with the main Ethernet network and supports both public and private networks. It gives the possibility of building an enterprise class Ethernet platform.

3. Introduction to the IoT

3.1. Overview of the IoT and the IIoT

The internet of things (IoT) consists of two segments of varying difficulty. One subdivision is the human internet of things (IoT), which is a major improvement in which the dominant type of interaction is client server [19]. The IoT provides increasingly intelligent services to satisfy rich semantic requests. Another subdivision is the industrial internet of things (IIoT), which is intended for complex task collaboration, decision making based on collected data, and remote access to machinery [20].

The IoT is a new technological paradigm, a global network of machines and devices capable of interacting with each other. At the application level, the IoT can perform a variety of light tasks based on predefined consumer requirements, including operations such as automatic floor sweeping, intelligent identification, and linkage of traffic lights. The value of the IoT to the enterprise lies in the ability of connected devices to communicate with each other and integrate with vendor-managed inventory systems, customer support systems, business intelligence applications, and business analytics [21].

The industrial IoT is a technology to improve the efficiency and quality of manufacturing by software and modeling industrial knowledge and experience. At the application level, the biggest difference between the industrial IoT and the IoT is that it is designed for heavy-duty tasks, such as smart manufacturing, environmental monitoring, intelligent transportation, enemy reconnaissance, etc.

3.2. Convergence Point of Hyperledger with IoT and IIoT

According to the Global System for Mobile Communication Association (GSMA) statistics, the number of IoT devices connected worldwide is up to 14.7 billion in 2021. The IoT is already widely used in numerous areas, and as mentioned earlier, blockchain technology is widely researched but difficult to implement, especially for commercial applications. The IoT urgently needs completely new technologies to solve the challenges it faces. This paper summarizes the possible integration points of Hyperledger and the IoT based on the study of Hyperledger-enabling technologies.

(1) Distributed storage and collaboration ensures tamper resistance for large volumes of data and decisions.
(2) Fine-grained permission control enhances privacy protection between enterprises and consumers.
(3) Hyperledger supports multiple chaincode authoring languages and provides channel-oriented chaincode lifecycle management, enhancing the efficiency of system collaboration.
(4) The fine-grained state-based endorsement strategy enhances the security of enterprise transactions.
(5) The efficient consensus mechanism effectively reduces network latency for device or large appliance collaboration and enables faster node state agreement, which can provide millisecond response time for the industrial IoT.
(6) Peer-to-peer authentication ensures high portability of identities and provides great convenience for portable IoT device identification.
(7) The highly modular framework and support for diverse chaincode writing languages enables Hyperledger to be quickly integrated with any IoT and industrial IoT system.
(8) The dynamic network makes the system highly flexible and robust to meet the basic business requirements of the IoT.

This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

4. Hyperledger Applications in the IoT

In this section, the research in the direction of the IoT is subdivided into nine application areas: IoT security, smart fisheries and smart agriculture, smart toys and IoT games, smart fitness, smart city monitoring, smart transportation, smart grid, smart construction projects, and smart energy (energy saving direction).

4.1. IoT Security

In the current research, privacy, confidentiality, and integrity security of data received more attention than other security requirements. Since the large amount of data generated by IoT devices is mainly processed by centralized cloud services, it is difficult to guarantee the privacy and confidentiality of these data. Wang [22] proposed a Hyperledger (Fabric 1.1.0) based the IoT data integrity verification scheme. In Figure 1, it is shown that the IoT data are split into multiple fragments and automatic verification and processing of device metadata is achieved and records are stored through predefined smart contracts. The cloud service provider is only responsible for returning the validation results to the user. This reduces overhead and computational costs, but lacks the design to handle more complex data types. To solve the problem of transaction security between different cloud service providers, Yang [23] proposed a federated cloud system based on Hyperledger (Fabric 1.0). In Figure 2, it is shown that this system is designed to determine the trusted level mechanism through user credit value instead of centralized management, and chaincode is signed between different cloud service providers for secure transactions. To a certain extent, trust is ensured and the utilization of cloud computing resources is improved. In cloud services that store datasets, there is a risk of malicious tampering and a single point of failure of the dataset model of the data owner. Dib [24] proposed a dataset utilization system based on Hyperledger (Fabric 1.1). In Figure 3, it is shown that the cloud service stores only the data model encrypted by the data owner, and consumers pay for the service through Hyperledger when sharing the dataset. The transparency of the dataset being utilized and the security of the dataset are enhanced, but no regulatory policy is designed for high trust level users. In supply chain systems, where data security is the first concern, Cao [25] proposed a traceability system (Sawtooth) for the steel industry. In Figure 4, it is shown that the data of each link is stored through a smart contract, and the regulator obtains all the circulation data through the block. Consumers scan the RFID code to obtain the final traceability information.

The automatic handling of compromised devices can avoid dangerous behaviors in time; Rodriguez [26] proposed a Hyperledger (Fabric)-based IoT device monitoring scheme. In Figure 5, the source and target devices are shown to verify the transaction reliability through the endorsing node in Hyperledger, and the dangerous devices are automatically isolated by chaincode. The security of device data are ensured. To solve the problems of latency and efficiency, Kim [27] proposed a lightweight scheme combining deep learning and Hyperledger (Fabric). In Figure 6, it is shown that the system, based on the node behavior, latitude and longitude of the network nodes, etc., and clustering, can generate multiple clusters using the clustering K-means algorithm. The system generates the corresponding chain verifier (consisting of four nodes screened) to verify the communication legitimacy and store the transaction records. The security of the data is improved to some extent. The configuration data of the IoT devices is an important part of the IoT data, and once tampered with will directly affect the original task direction. Helebrandt [28] proposed a Hyperledger (Composer)-based configuration file system for IoT devices. In Figure 7, it is shown that on-chain and off-chain (storing large configuration files) storage is designed to encrypt the messages that modify the configuration and load the management ID, device ID, and timestamp into a new block. However, it lacks the supervision of more configuration information, such as power, CPU utilization, and disk space. The multi-level proxy approach helps to secure the transmission of the IoT data, so Mbarekp [29] proposed a multi-level proxy-based IoT data protection system based on Hyperledger (Fabric 1.1.0).

In Figure 8, it is shown that the validity of the blocks is verified by the check of the three level agents, which ensures the security of the data.

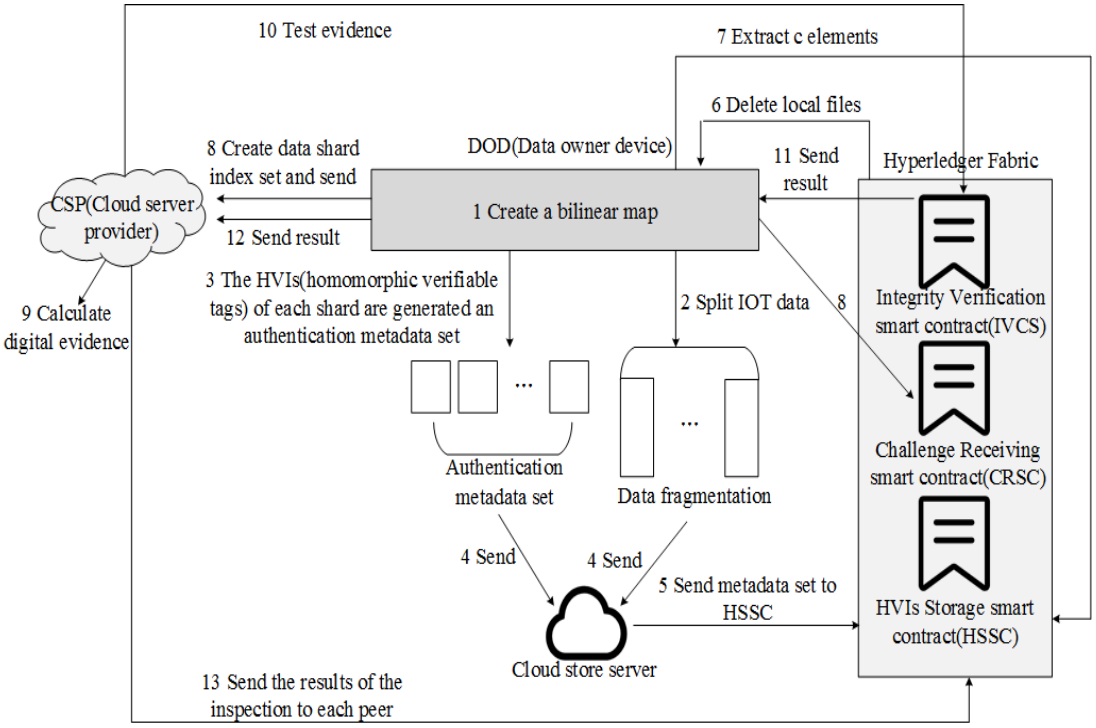

Figure 1. Reconstruction diagram of reference [22].

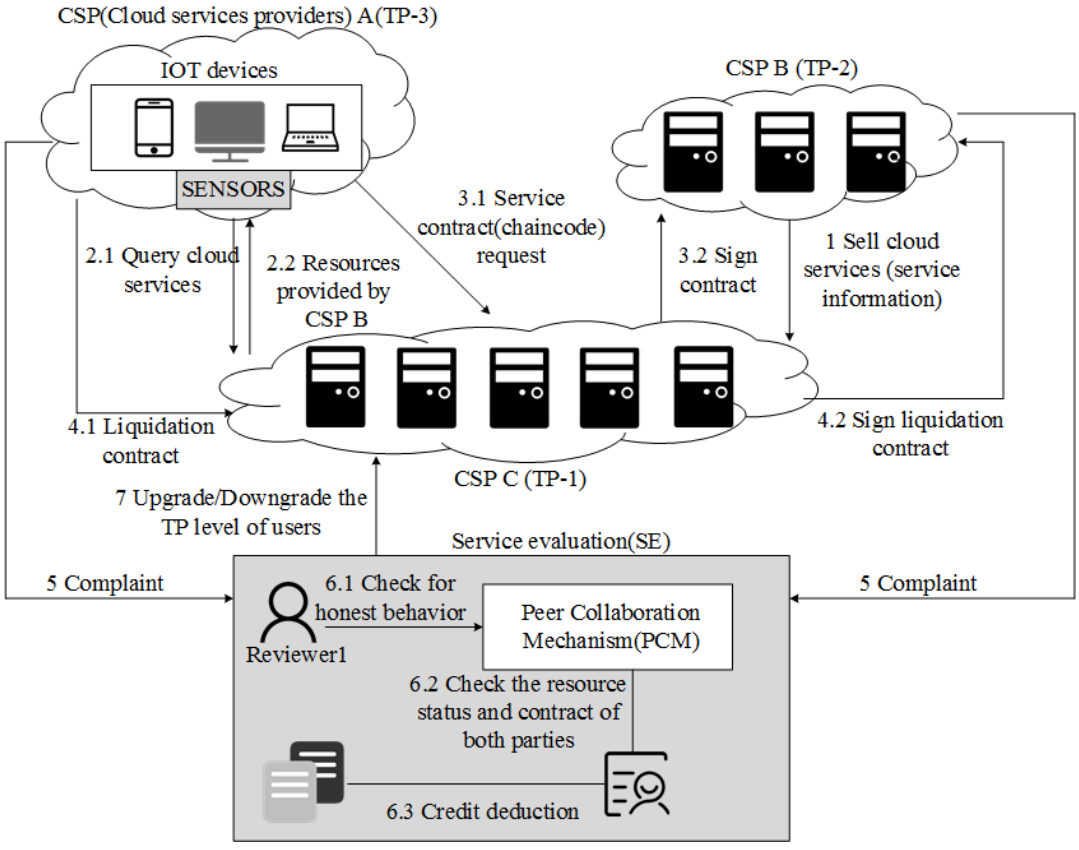

Figure 2. Reconstruction diagram of reference [23].

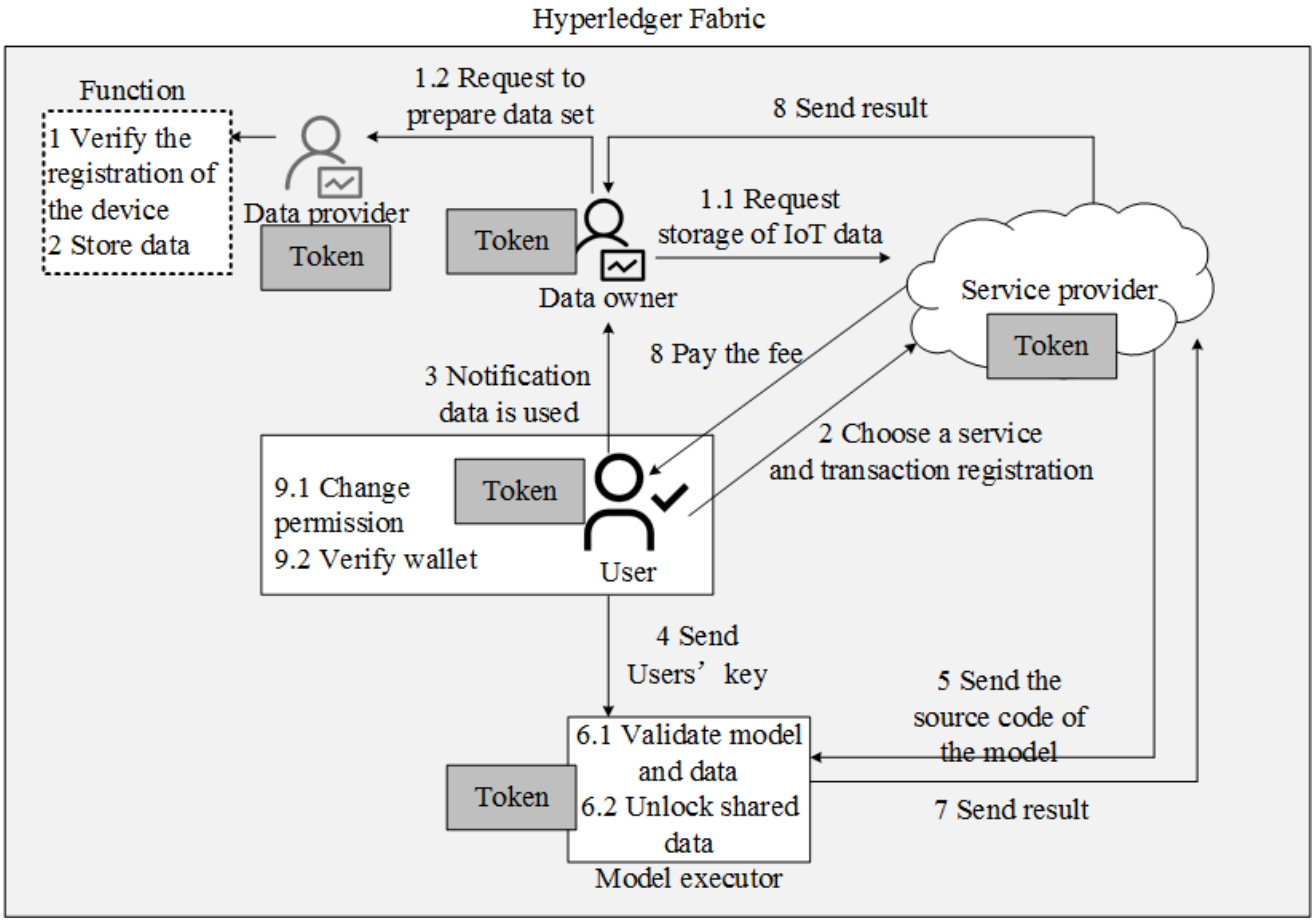

Figure 3. Reconstruction diagram of reference [24].

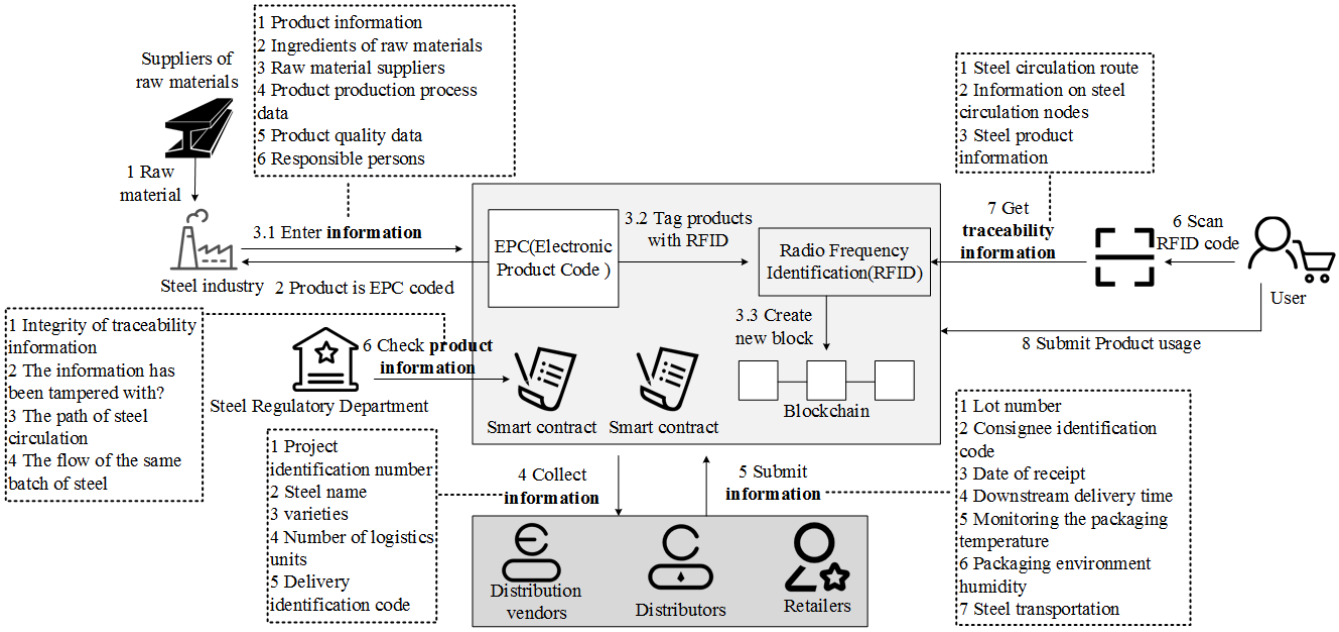

Figure 4. Reconstruction diagram of reference [25].

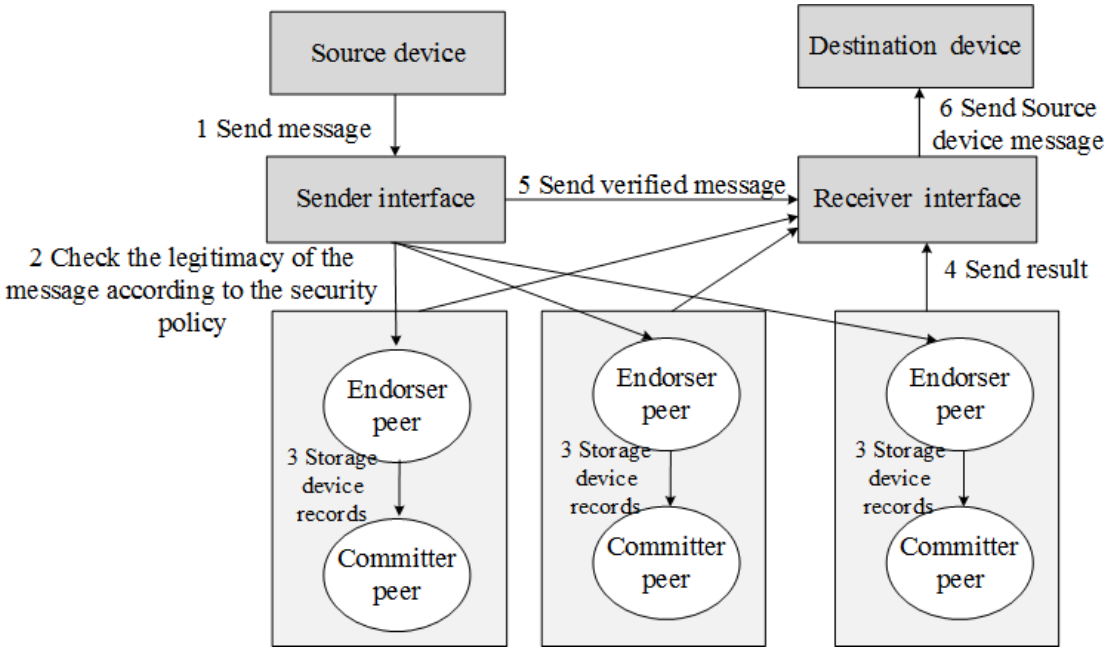

Figure 5. Reconstruction diagram of reference [26].

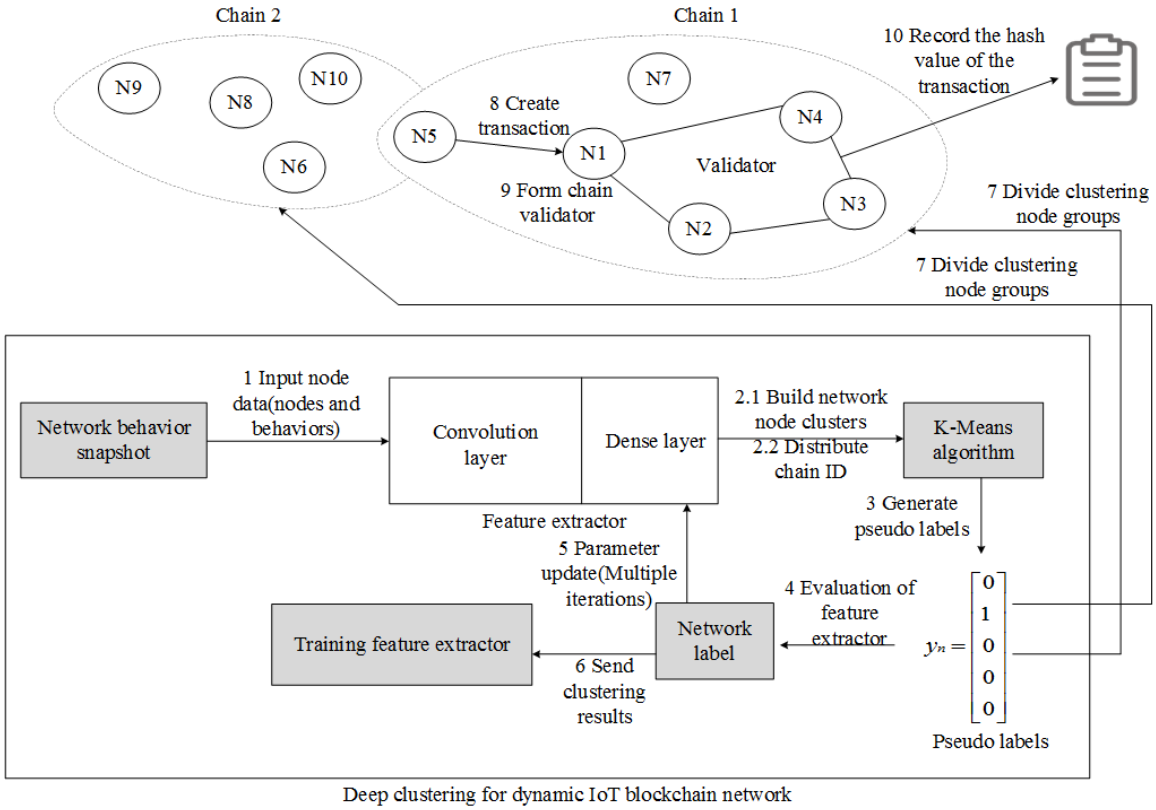

Figure 6. Reconstruction diagram of reference [27].

Since most of the key management in Hyperledger is issued and managed by government nodes, there are still security problems such as key tampering and forgery, so Ribeiro [30] proposed a distributed key management scheme (Fabric 1.4.0). In Figure 9, it is shown that by signing a smart contract between the device and the connection server, the system establishes a temporary session key to safeguard the device privacy, and this scheme solves the security problem of the device key to some extent.

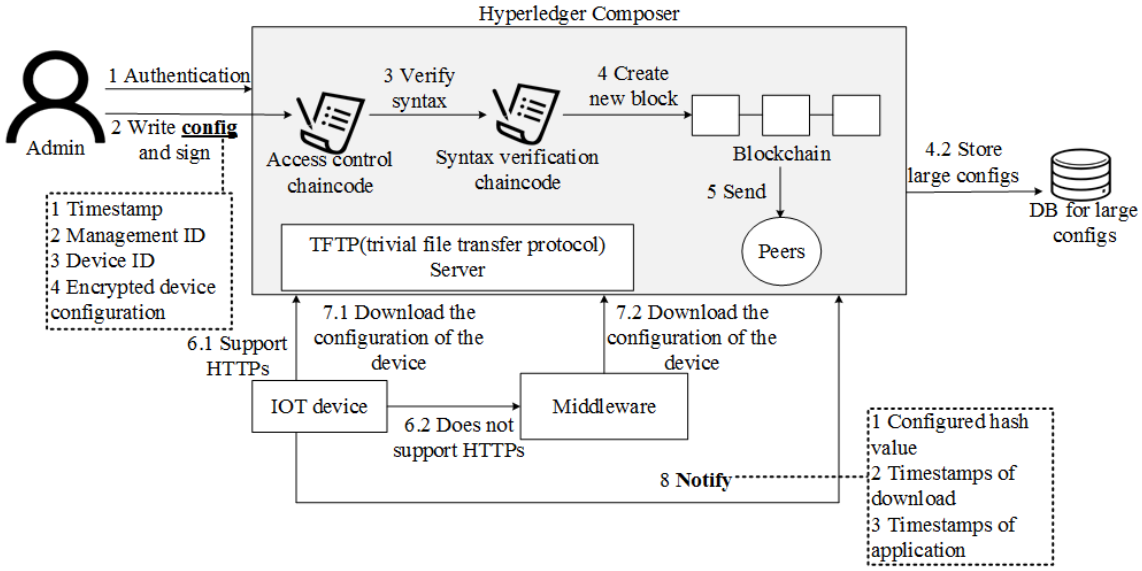

Figure 7. Reconstruction diagram of reference [28].

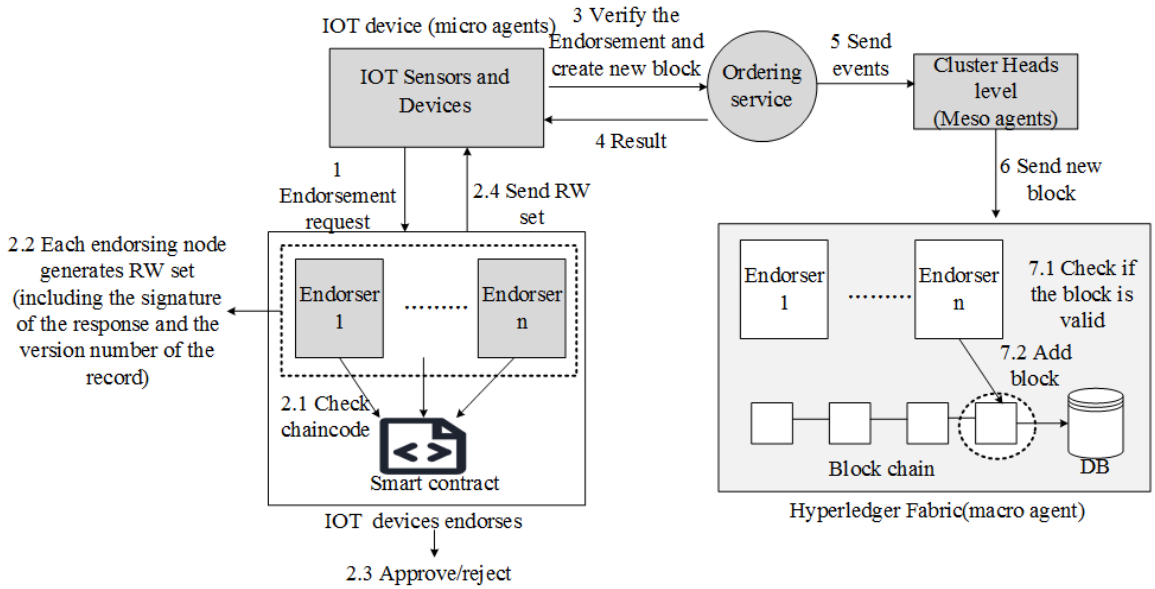

Figure 8. Reconstruction diagram of reference [29].

Several studies focused on security requirements such as authentication, authorization, and billing in IoT security. Hang [31] proposed an IoT communication platform based on Hyperledger (Fabric 1.2). In Figure 10, it is shown that the system uses smart contracts to achieve secure access between devices, and stores data in Hyperledger to improve the security of transactions. Due to its lightweight architecture, it provides feasibility for implementing large-scale IoT device communication. To improve the reliability of smart contracts, Liu [32] proposed a data access control system (Fabric 1.4.3). In Figure 11, multiple users are shown to jointly develop access control policies, and the system stores records and URLs for these data through Hyperledger. This scheme reduces the pressure on on-chain storage. To address the centralized root management in top-level domain authorization, Zhang [33] proposed a distributed root management scheme based on Hyperledger (Fabric 1.4). In Figure 12, it is shown that the transactions for a domain authorization are sent to multiple authorization nodes, and only the authorization nodes that respond within the time threshold are considered valid. The authorization messages are counted and processed automatically by a smart contract. To improve the efficiency of

authentication, Chi [34] proposed a data co-authentication scheme (Fabric). In Figure 13, it is shown that the user's identity information is split into labeled data and real data. The network is divided into multiple communities according to the K-medoids algorithm [35], and the similarity between the labeled data and the community data of the nodes is measured using the cosine similarity algorithm [36]. Users retrieve relevant information based on tags. The efficiency of identity related data retrieval and sharing is improved.

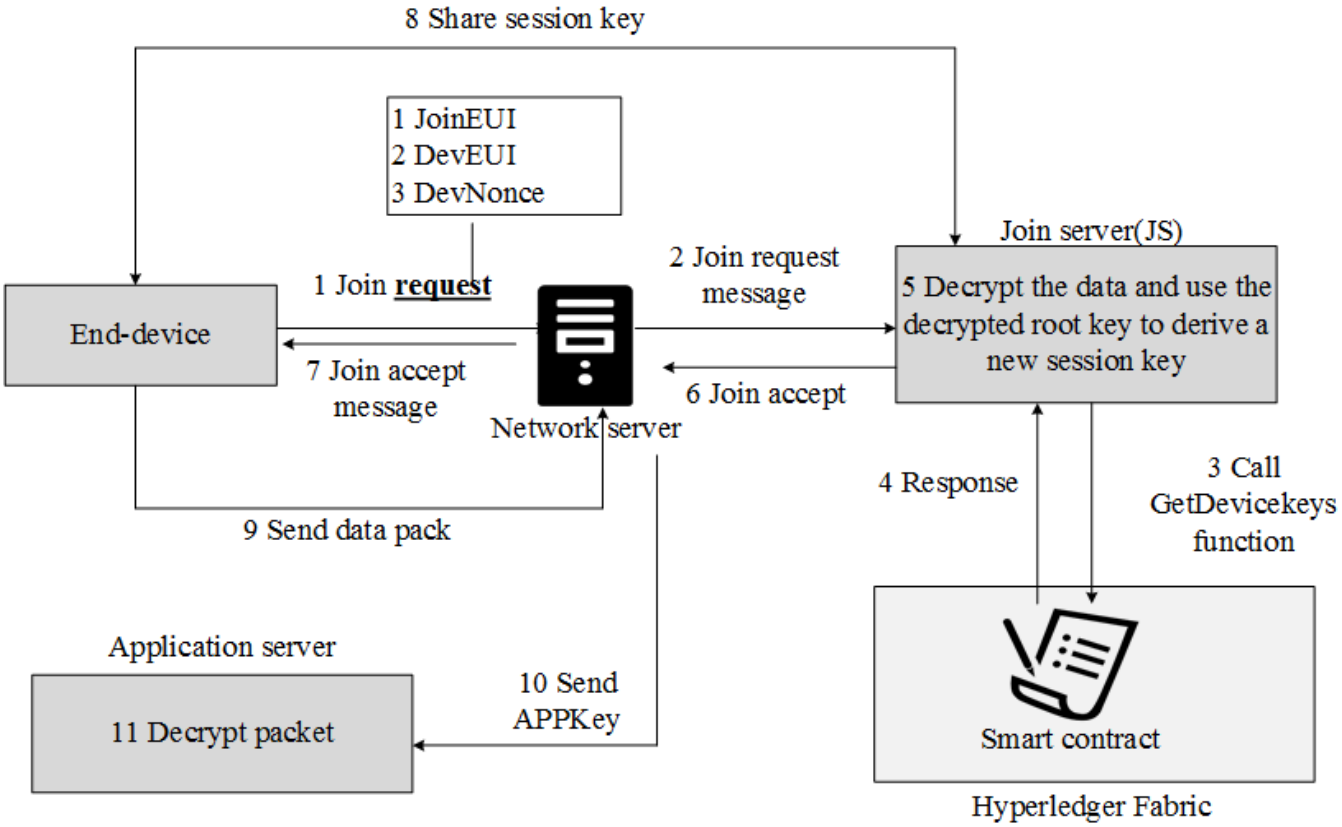

Figure 9. Reconstruction diagram of reference [30].

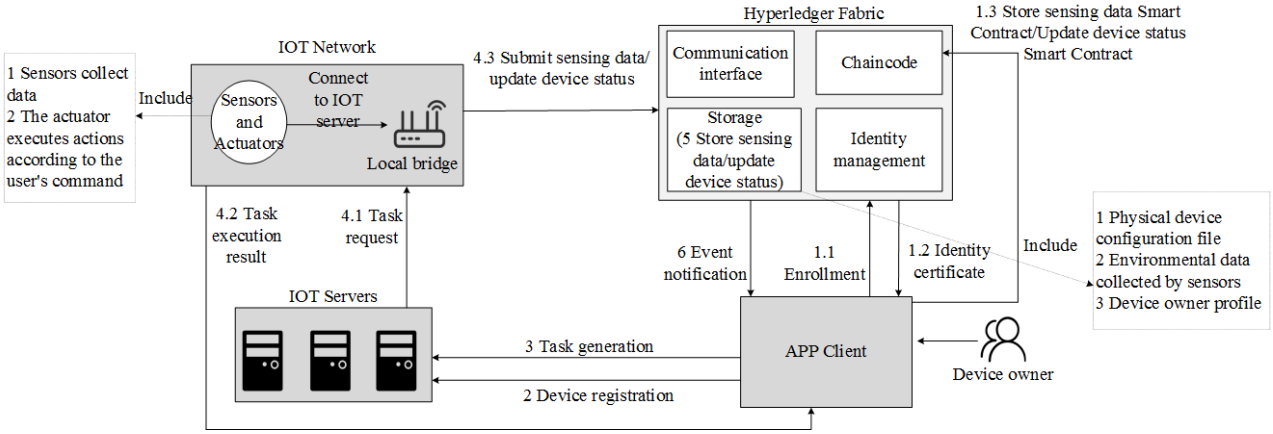

Figure 10. Reconstruction diagram of reference [31].

In Hyperledger, the centralized authorization and authentication of CAs may generate risks such as tampering and forgery. To solve the problem of centralized CA authorization, Siris [37] proposed two decentralized authorization strategies based on Hyperledger (Fabric). In Figure 14, it is shown that multiple organizations are authorized instead of unified authorization by CA nodes, and the authorized nodes for transactions at a certain

moment are filtered according to the corresponding time of the nodes. The authorization efficiency is improved while ensuring the security of distributed authorization, but the first strategy requires higher computational cost. To solve the problem of centralized CA authentication, Kakei [38] proposed a strategy for distributed CA authentication (Fabric). In Figure 15, it is shown that the CA nodes in Texas are divided into meta-CA and CA. The cross-authentication between meta-CA and CA determines whether this CA node is a trusted party, and this scheme improves the reliability of CA nodes to a certain extent.

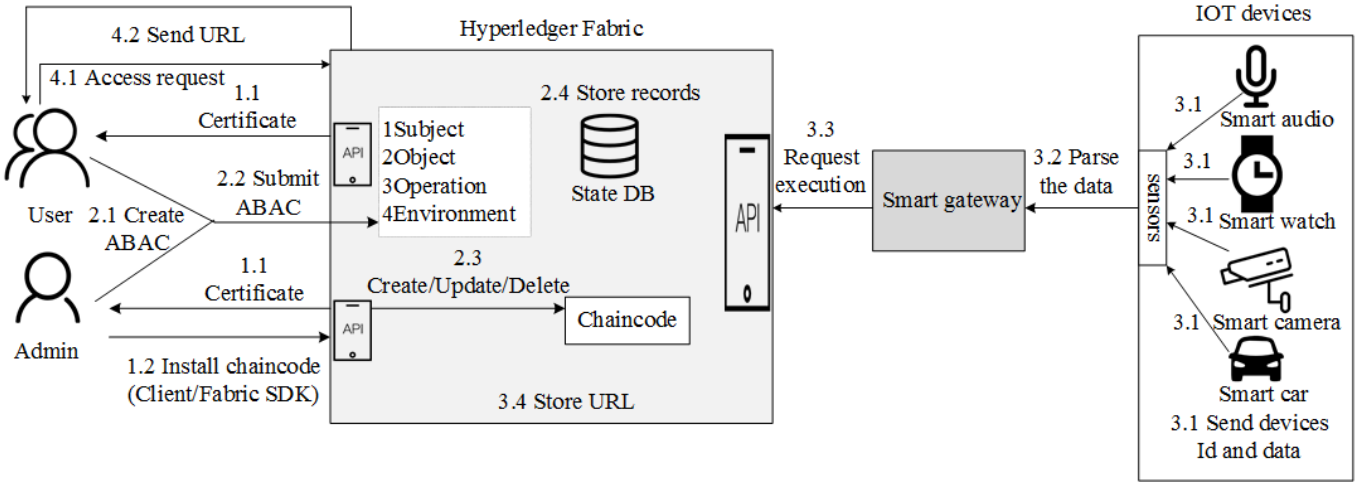

Figure 11. Reconstruction diagram of reference [32].

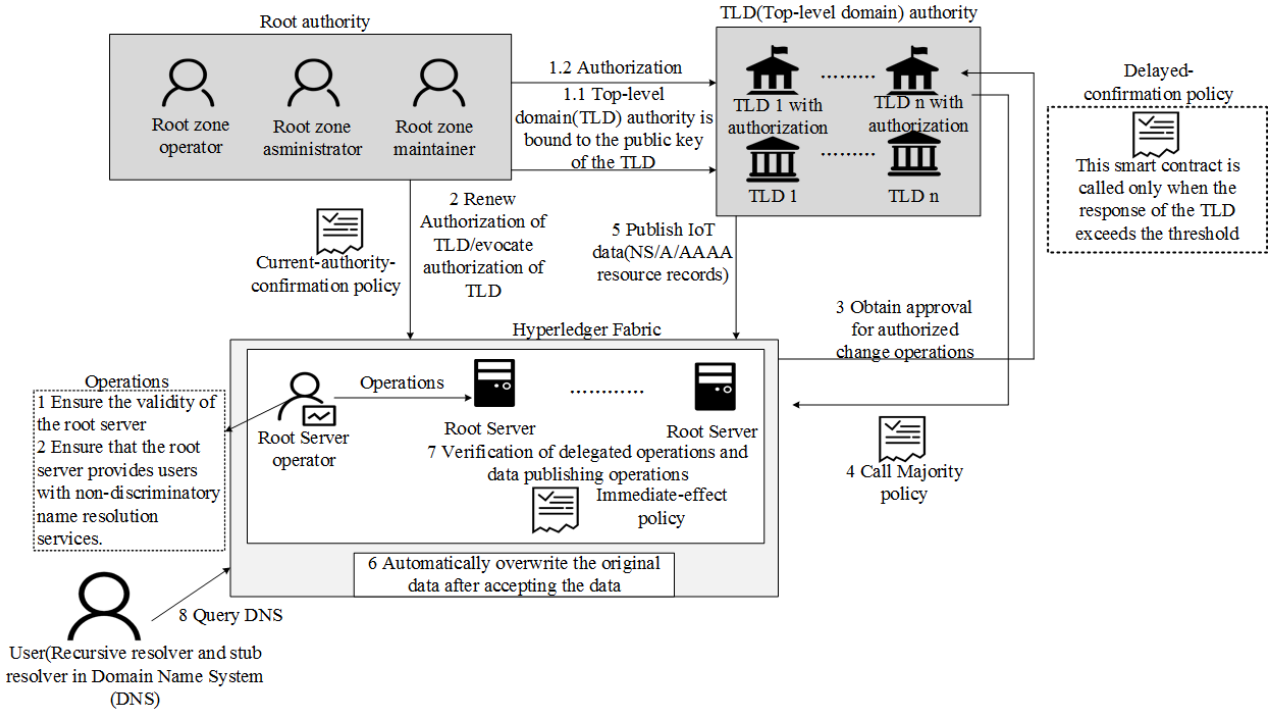

Figure 12. Reconstruction diagram of reference [33].

To provide a generalized Hyperledger-based authorization architecture, Pajooh [39] proposed a multilayer blockchain model (Fabric) based on a cellular system. In Figure 16, it is shown that the network is divided into three layers based on SI (swarm intelligence) and EC (evolutionary computation) algorithms. Multiple base stations are connected in Hyperledger to achieve distributed authorization and authentication of the IoT devices. The model reduces the network load, but does not actually build a testbed.

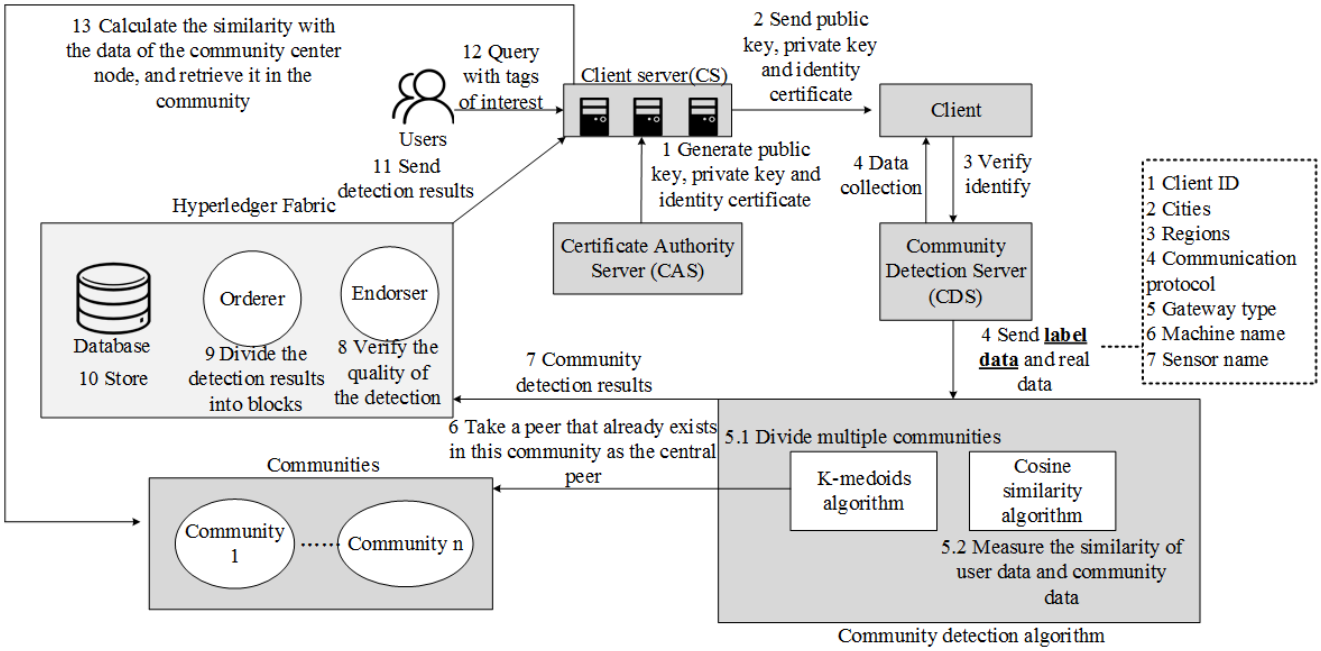

Figure 13. Reconstruction diagram of reference [34].

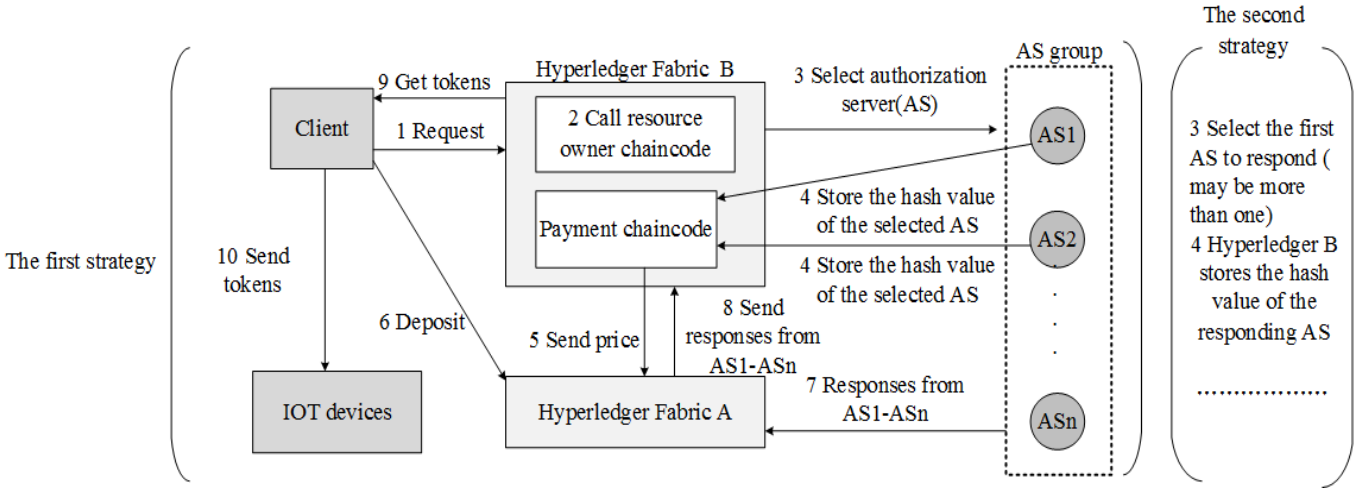

Figure 14. Reconstruction diagram of reference [37].

An IoT system developed based on Hyperledger should address the security requirements for service availability, and encryption of data is one of the effective ways to ensure that data are not attacked. Zhou [40] proposed a fully homomorphic computing scheme (Fabric) for IoT data protection. In Figure 17, it is shown that, by encrypting the session message using a homomorphic encryption algorithm, the system verifies that the message did not change through multiple servers. It effectively protects the IoT data from attacks with good performance. Hou [41] proposed a scheme for edge computing to protect data. In Figure 18, it is shown that the messages of the devices are obtained through LoRa gateway and the uplink messages are stored in Hyperledger, which reduces the possibility of the messages being attacked.

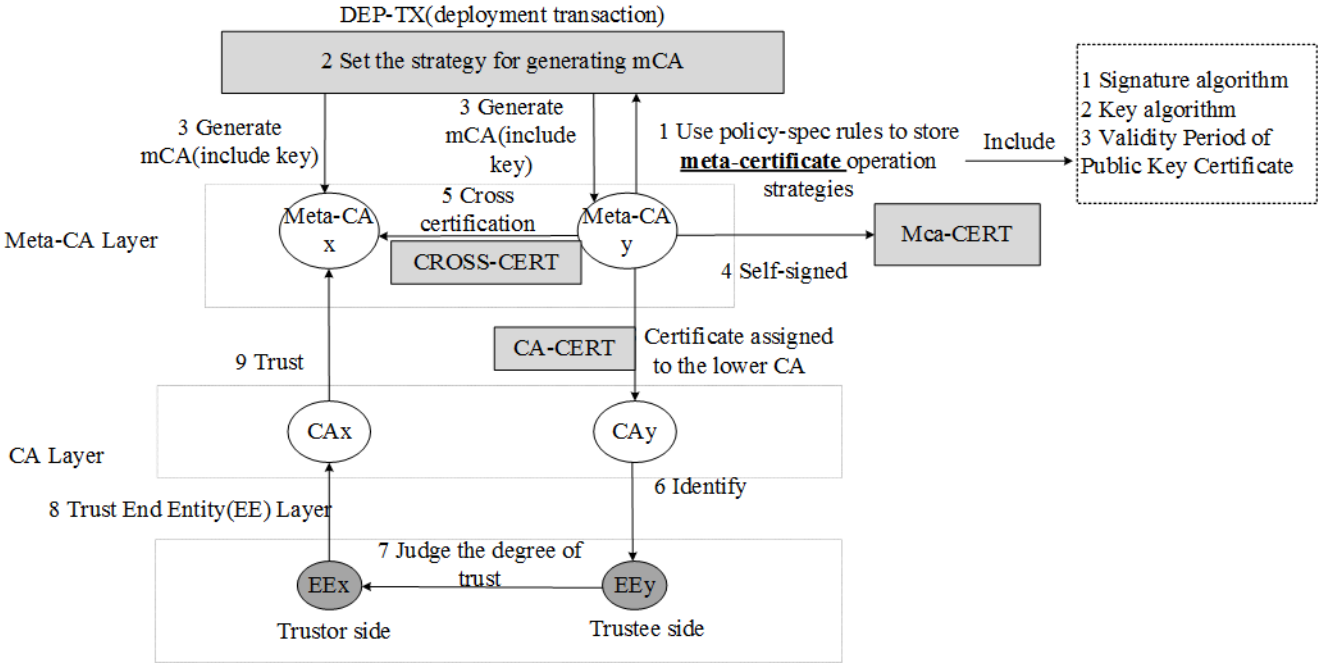

Figure 15. Reconstruction diagram of reference [38].

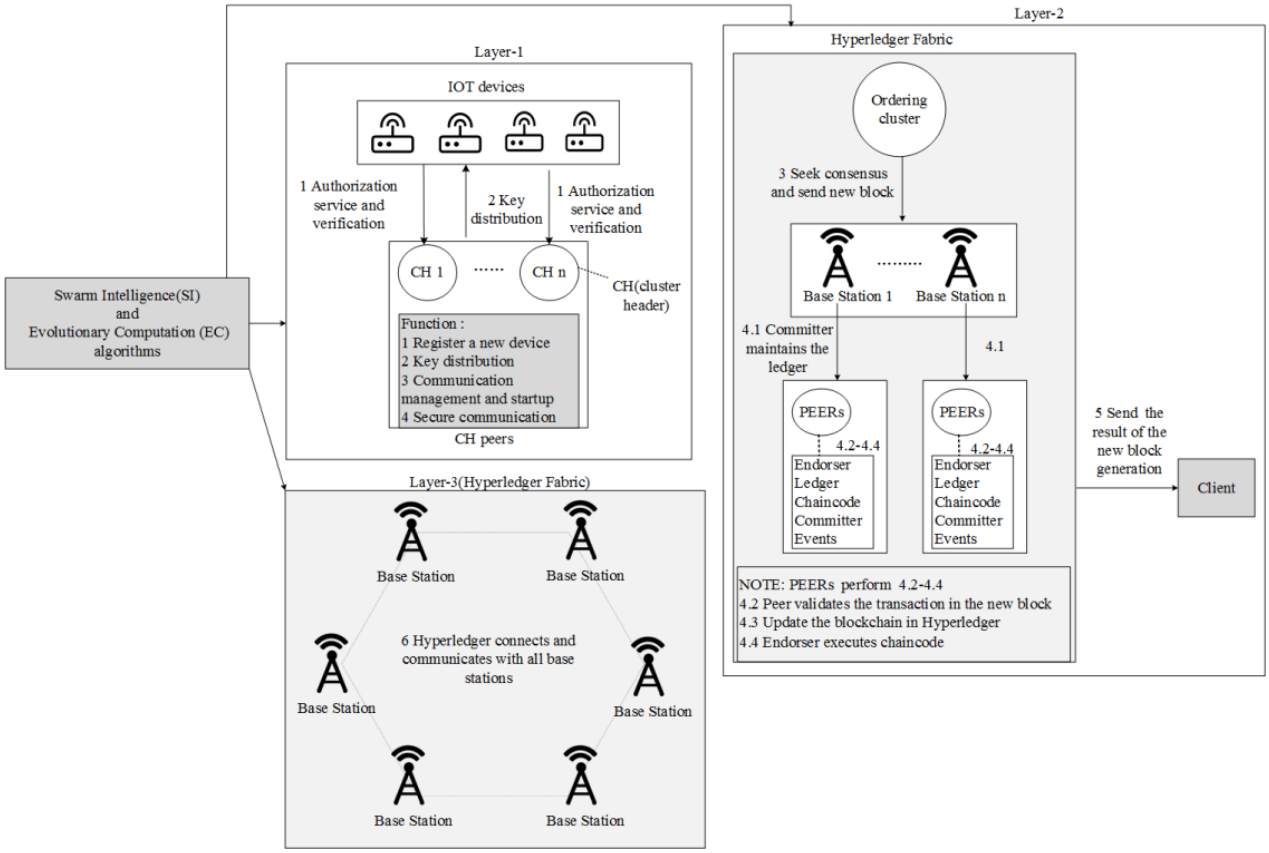

Figure 16. Reconstruction diagram of reference [39].

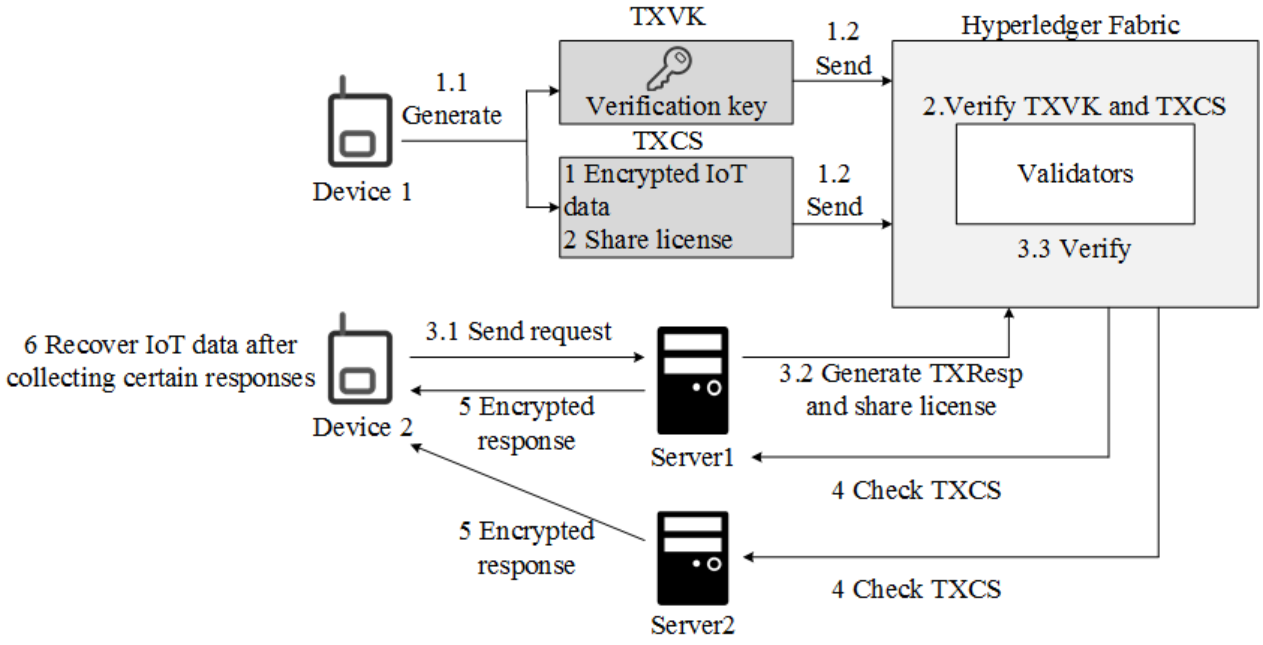

Figure 17. Reconstruction diagram of reference [40].

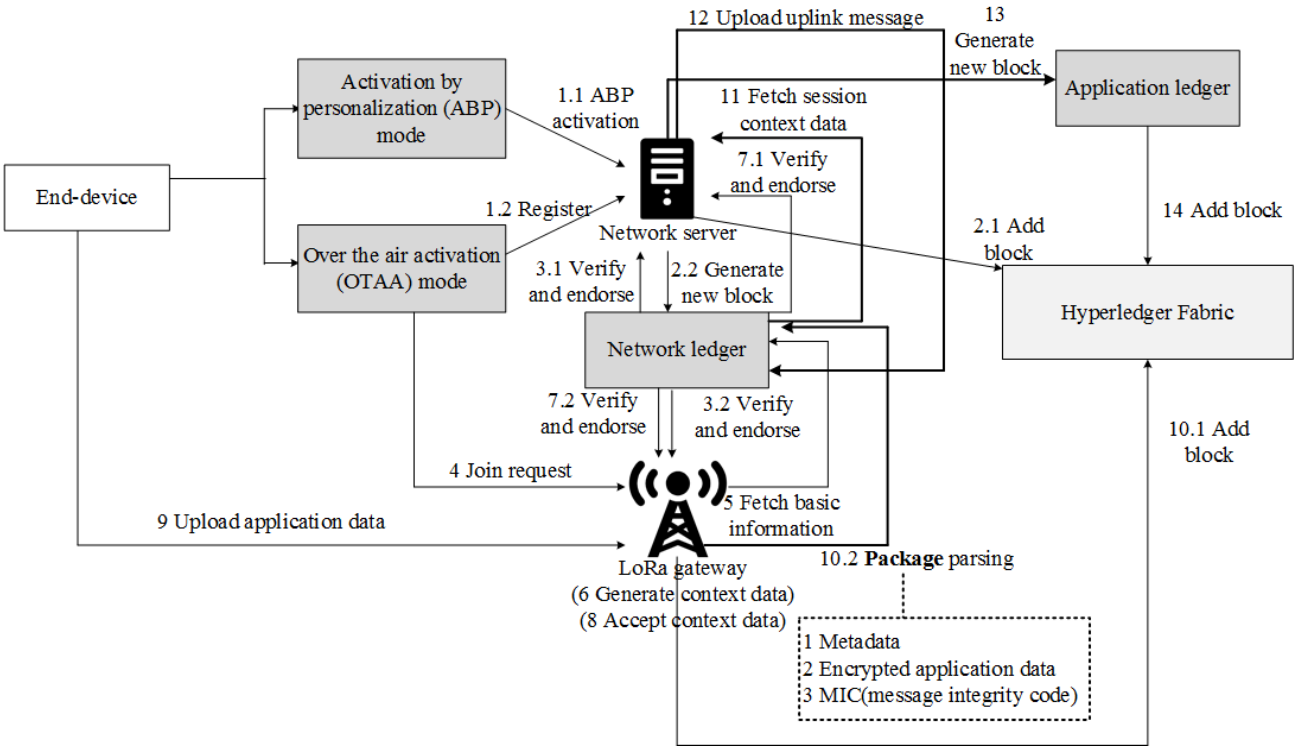

Figure 18. Reconstruction diagram of reference [41].

In Tables 1–3, this paper presents a comprehensive comparison of the above schemes. The table compares their differences in six aspects: year, consensus algorithm, incentive mechanism, application domain, issue addressed, and performance evaluation. In the performance evaluation metrics, this paper presents some of the main experimental results of the schemes.

Table 1. Differences between IoT security solutions (NS: mention but not specify, NM: not mentioned).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issue Addressed	Performance Evaluation
Wang et al.	2019	NM	×	IoT data	Integrity verification for large-scale IoT data	The number of fragments of data is 200,300,400 and 500, the time cost is about 20 s, 26.5 s, 42.5 s, and 59 s.
Yang et al.	2020	NM	×	JointCloud	Secure transactions	The transactions are 100, 1000, and 2000, the latency time is about 0.18 s, 2.01 s, and 5.23 s.
DIb et al.	2019	Solo	×	IOT data	Intelligent data exploitation	The transaction sending rate is 50,100,500, the average latency is 0.98 s, 3.14 s, and 5.25 s
Cao et al.	2020	Proof of concept (PoC)	×	Steel products	Quality traceability	System response time is less than 10 ms at 1000 users
Rodriguez et al.	2015	PBFT	×	IOT devices	Security	The average committed latency for the four networks configured by the system is 298.62 ms, 514.42 ms, 514.32 ms, and 730.11 ms
Kim et al.	2021	Proof of elapsed time	×	IOT networks	Communication efficiency	The minimum convergence time for the final solution obtained by the genetic algorithm is 2600 ms, the maximum convergence time is 4300 ms.
Helebrandt et al.	2019	NM	√	IOT networks	Security	The profiles is 2, 5, 10, the processing time is 4.2 ms, 407 ms, and 402 ms.

Table 2. Differences between IoT security schemes (continue Table 1).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Mbarek et al.	2019	NM	×	IOT devices	Device security	The number of nodes is 200, 600, 800, and 1200, the average latency is about 2.4 ms, 2.6 s, 3.5 ms, and 5 ms.
Ribeiro et al.	2020	NS: lightweight consensus algorithms	×	Low-power wide-area network (LPWAN)	Efficiency, security	The number of requests is 1000, 3000, 5000, and the average system latency is about 13 s, 35 s, and 59.24 s.
Hang et al.	2019	NS: byzantine fault tolerant (BFT)/crash fault tolerant (CFT)/PBFT	×	IOT platforms	Data accessing	The records is 500, 250, 5000 and 10,000, the average latency is 271 ms, 559 ms, m656 ms, and 752 ms.
Liu et al.	2017	PoW	×	IoT data	Access control	The requests are 50, 200, 500 and 1000, the average time is about 0.12 s, 0.075 s, 0.065 s, and 0.062 s.
Zhang et al.	2021	NM	×	Domain name system	Root zone management	The TPS of write data is 100,200,250, the average latency is about 100 ms, 1450 ms and 10,000 ms. Read data TPS is 100,200,250, the average latency is about 10 ms, 15 ms, and 30 ms.
Chi et al.	2020	NM	×	IIOT	Efficiency of data sharing	The query tag is 6, when the query request reaches 200, 300, and 400, the time cost of the system is 12 s, 17 s, and 24 s.

Table 3. Differences between IoT security schemes (continue Tables 1 and 2).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Siris et al.	2020	Proof-of-work (POW), Proof-of-authority (POA)	×	Constrained IOT devices	Decentralized authorization	The latency for the request event is approximately 4.2 s.
Kakei et al.	2010	NM	×	Public key infrastructure	Distributed CA	The average processing time for initialization, participation, deployment, and validation is 2215.6 ms, 2366.7 ms, 2463.3 ms, 2404.8 ms, 65.7 ms.
Pajooh et al.	2021	Proposed	×	5G-enabled IoT networks	Security	The transaction sending rate is 100,200,400, and 500, the average delay is about 0.5 s, 1.6 s, 5.7 s, and 9.8 s
Zhou et al.	2018	NM	×	IoT data	Data security	The number of sending transactions per second is 100, 200,300, the average latency is 2.4 s, 5.8 s, and 6.4 s.
Hou et al.	2020	PBFT	×	Long range (LoRa) system	Security, availability	The application package is 500, 1000, and 2000, the average processing time is about 12 s, 20 s, 25 s. The number of requests is 400, 1200, 2200, the system connection is about 3.5 s, 3.7 s, 6.3 s.

4.2. Smart Fisheries and Smart Agriculture

Advances in information technology contributed to the digital transformation of fisheries and agriculture. At the conceptual level, smart fisheries are similar to smart agriculture. Both offer, through the deep integration of big data, blockchain, artificial intelligence and other information technology, access to real-time data collection, quantitative decision making, intelligent control, accurate investment, yield prediction, and other personalized services [42]. Smart fisheries focus on water quality monitoring to achieve analysis and regulation of water quality in large areas. In contrast, the main need of smart agriculture is to make intelligent decisions through real-time monitoring, and analysis to improve productivity and resource efficiency [43]. Currently, Hyperledger is less used in smart fisheries and agriculture, and the problems solved are mainly focused on data tamper resistance and real-time data flow.

It is difficult to regulate the fishery accurately, and the data are not tampered, so Hang [44] proposed a smart fish farming platform based on Hyperledger (Fabric 1.4.3). In Figure 19, it is shown that the actual water level data are predicted by the water level sensor, and the error is eliminated by using the Kalman filter algorithm. The system calculates the actual required water level and duration for automatic regulation. This platform provides a safer development idea for smart fisheries, but lacks interaction with different fisheries.

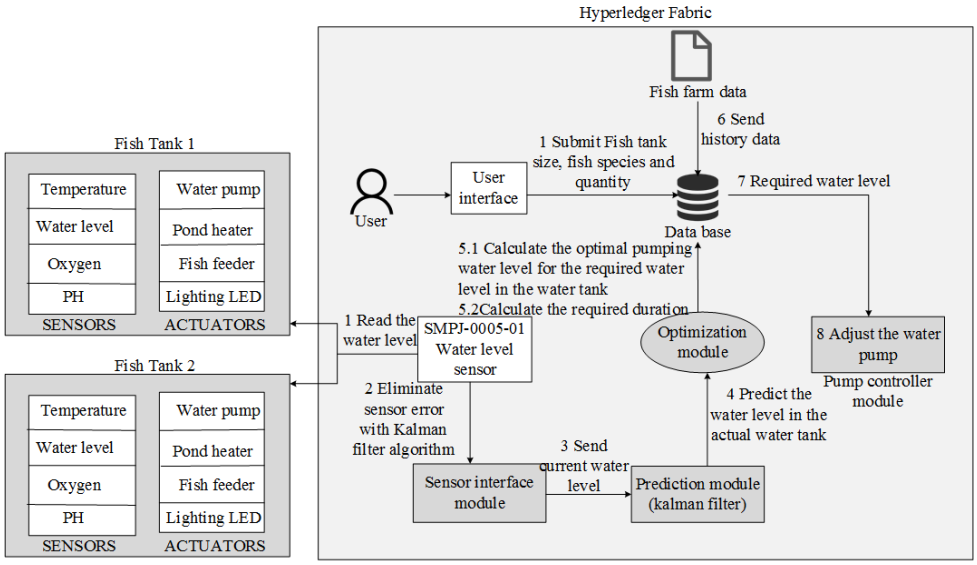

Figure 19. Reconstruction diagram of reference [44].

In a smart farm system, there are issues regarding the real-time monitoring of crops and reliability of product data. Lee [45] proposed a middleware for monitoring the food growth environment based on Hyperledger (Sawtooth). In Figure 20, it is shown that the crop data collected by the sensors are up-linked, and Hyperledger performs 10 cycles of authentication of the monitored data. The POET (proof of elapsed time) consensus is proven to have practical applicability with faster processing efficiency.

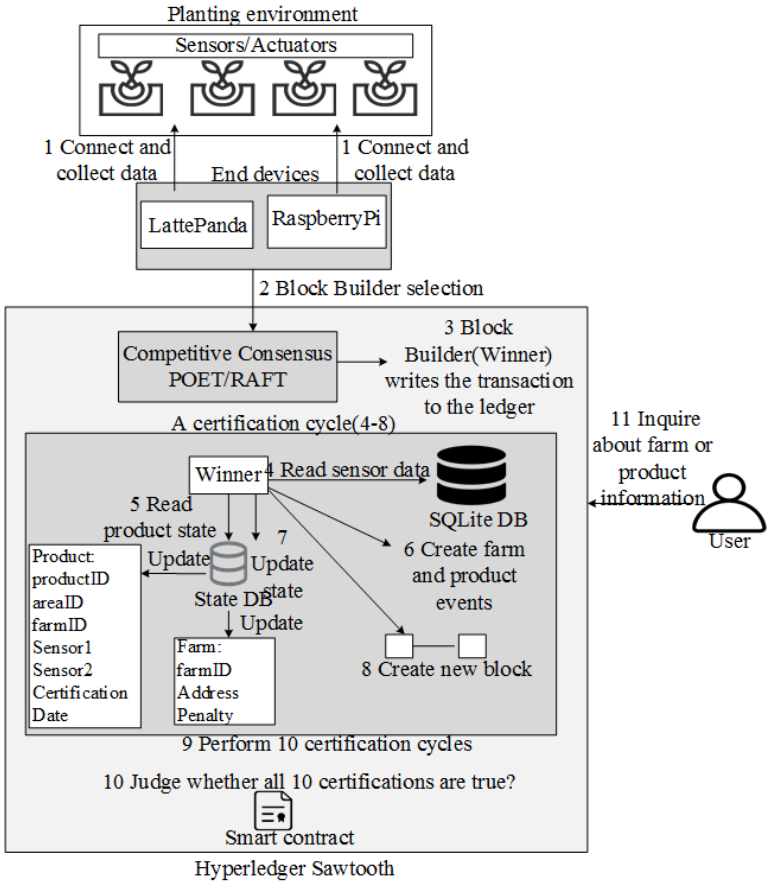

Figure 20. Reconstruction diagram of reference [45].

4.3. Smart Toys and IoT Games

While both smart toys and IoT games are intelligent entertainment services, they achieve different goals. The users of smart toys consist primarily of children, and are enabled by the integration of IT technologies to make phone calls, educate children, browse websites, as well as provide location tracking and other services. The types of smart toys available in the global market include additional mechanical toys, sound/image recognition toys, screenless toys, lifestyle toys, educational and construction games, as well as health tracking/wearable toys [46]. However, smart toys have the problem of the inability to exchange horizontal data. This is due to how difficult it is for heterogeneous APIs to accomplish data exchange between different systems [47], and results in a large amount of redundant data (data not needed by the user) that cannot be used effectively. IoT games break away from the traditional meaning of image and video-based games, which are games powered mainly by IoT technologies to interact with real objects in the physical world to obtain rewards. As a result, IoT games are oriented towards decentralized objects, mainly including location-based perception games. However, such games lack a robust technology to guarantee the authenticity of the tasks and the privacy of the users from being violated. Hyperledger provides an effective solution to the above problem.

In the data sharing of smart toys, horizontal data security exchange is difficult. Yang [47] proposed a toy data exchange model based on Hyperledger (Fabric 1.0). In Figure 21, it is shown that the toy data are desensitized and then the supplier generates a unique identifier for the toy, and Hyperledger checks and stores the toy data in Couch DB to ensure storage security.

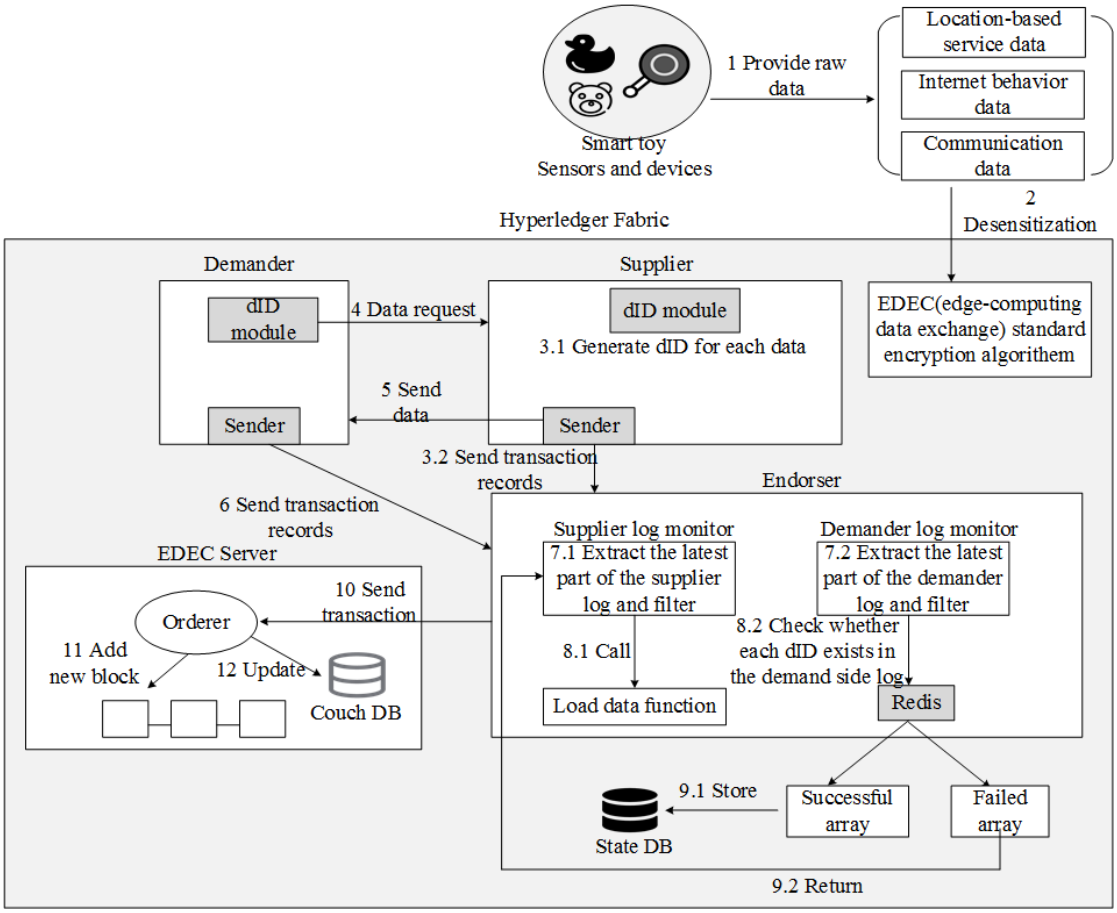

Figure 21. Reconstruction diagram of reference [47].

In the Hyperledger-based IoT gaming system, regarding real-time updates of game tasks, player privacy, and reliability of game task locations, Manzoor [48] proposed a

location-aware mobile hunting game (Fabric). In Figure 22, it is shown that the hunting task submitted by the player is validated by the smart contract, and only the reward information is posted without showing the hunting details. Players' rewards are secured through the wallet function that stores information about completed missions in Hyperledger. This enhances the transparency and security of rewards in location-based games, but the detection of IoT beacons is largely delayed and there is no guarantee that the location of the hunt is secure. Considering the situation that some players are unable to complete hunting tasks, Pittaras [49] developed a location-based mobile game for the interconnection of Ethereum and Hyperledger (Fabric 1.4). In Figure 23 (since the literature does not specify the design of Ethereum), only the design related to the Hyperledger is shown. Additionally shown in Figure 23, the system developed an advertising function and used chaincodes to count the number of times players watch the ads (advertisements) and automatically issues rewards.

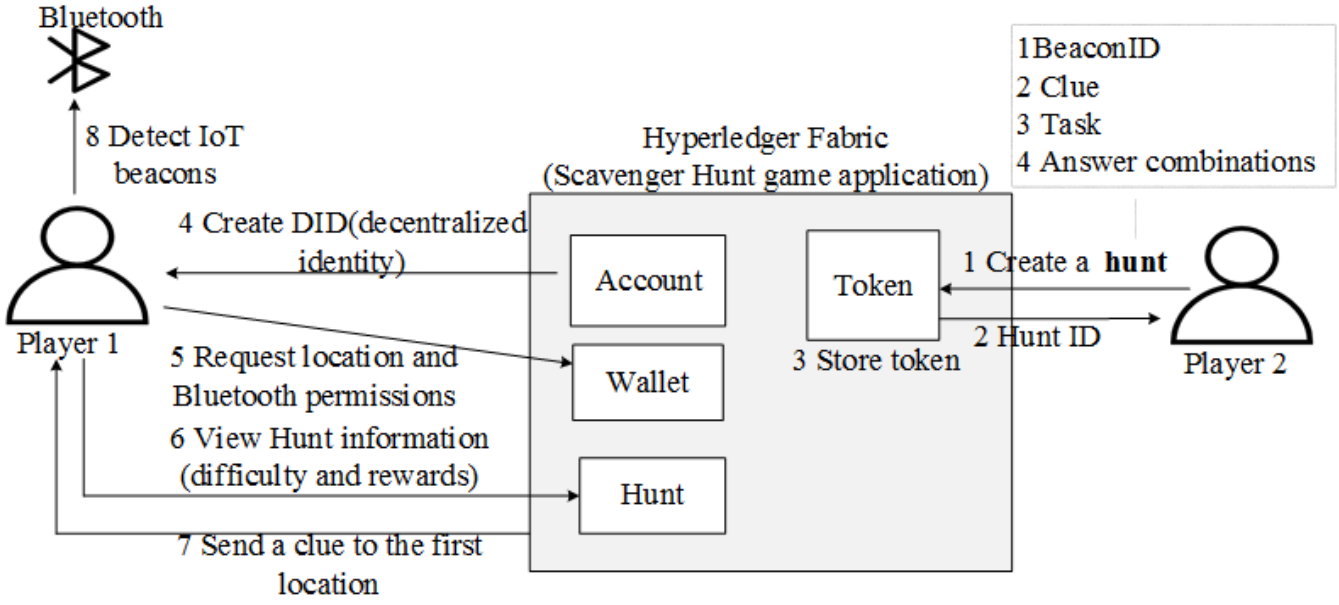

Figure 22. Reconstruction diagram of reference [48].

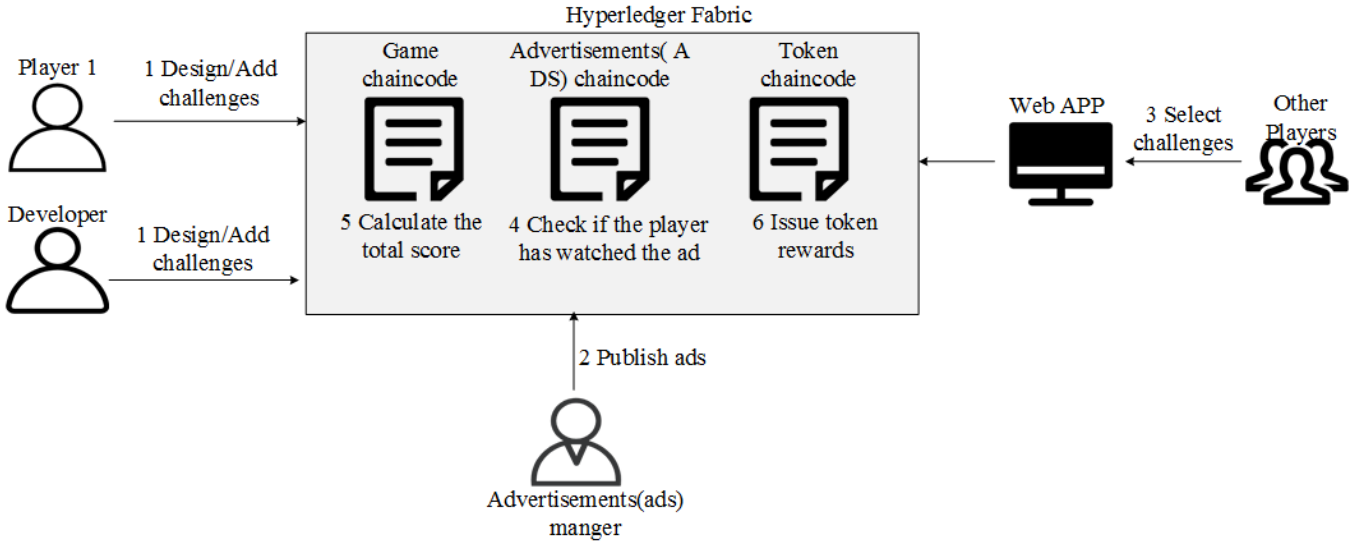

Figure 23. Reconstruction diagram of reference [49].

4.4. Smart Fitness

Smart fitness is one of the popular scenarios in the IoT. Smart fitness aims to acquire users' training data through sensors and combine them with artificial intelligence algorithms to provide services such intelligent training decisions, supervised diet, and predicted behavior. IoT-based smart fitness is divided into three categories: fitness trackers (including wearable and non-wearable sensors), exercise analytics, and fitness applications [50]. Currently, Hyperledger mainly addresses training models and decision making for smart fitness with secure, as well as enhanced accurate automation services.

In the Hyperledger-based fitness data system, to provide more secure and intelligent services, Jamil [51] proposed a fitness model based on Hyperledger (Fabric 1.2, Composer 1.13.0). In Figure 24, it is shown that an inference engine for fitness data is implemented using machine learning to provide reasonable fitness plans and diet plans. The inference knowledge threshold is compared and stored with the actual read data to update the inference information. The security of fitness data is improved.

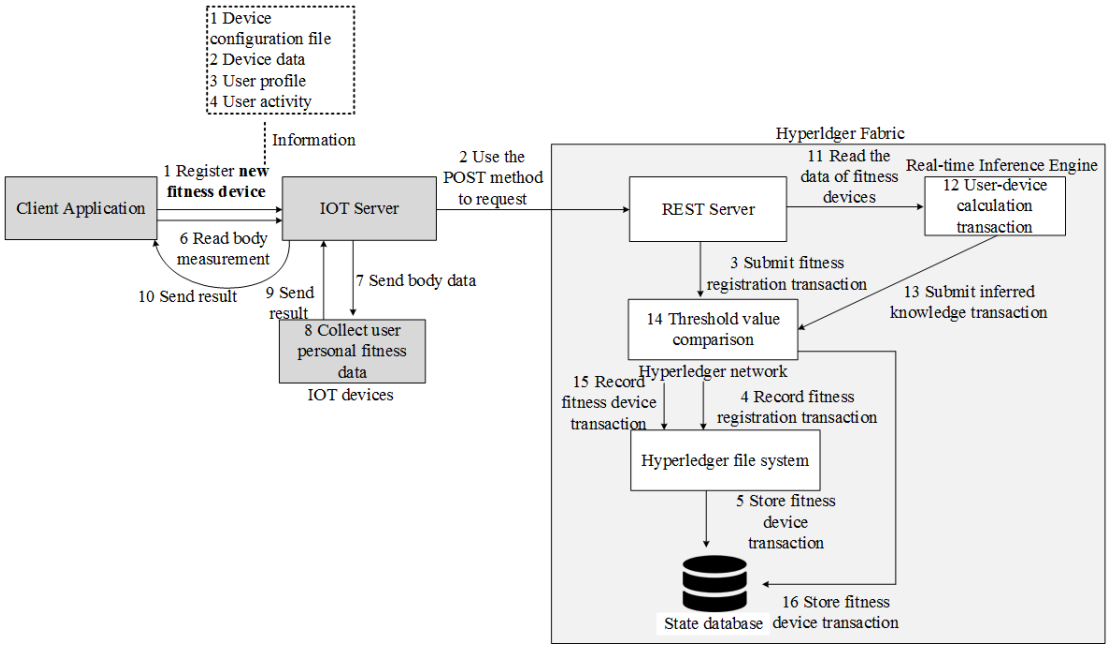

Figure 24. Reconstruction diagram of reference [51].

4.5. Smart Urban Monitoring

Urban monitoring is an important infrastructure in smart cities. IoT-based urban surveillance is the collection of information through cameras to obtain real-time status of geographic space, intelligent analysis, and detection of these data.

In urban monitoring systems, regarding the authenticity of monitoring information provided by users, Khan [52] proposed a monitoring information detection system based on Hyperledger. In Figure 25, it is shown that the importance of surveillance video/images is judged by the endorsement nodes of Hyperledger (Fabric 1.4), and the important information is detected and the frames are extracted for comparison with the original video using chaincode in priority. This system ensures the authenticity of the CCTV (closed-circuit television camera) data to some extent, but the detection mechanism is single.

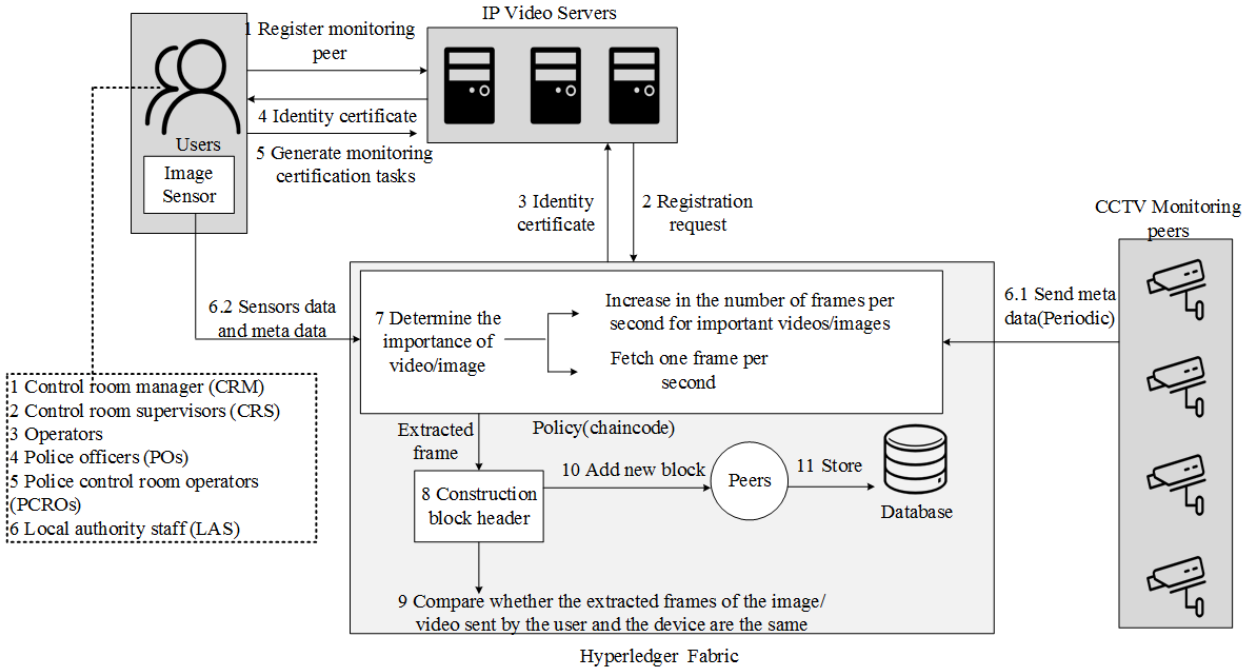

Figure 25. Reconstruction diagram of reference [52].

In Table 4, this paper presents a comprehensive comparison of the schemes in Sections 4.2–4.5. The table compares their differences in six aspects: year, consensus algorithm, incentive mechanism, application domain, issued addressed, and performance evaluation. In the performance evaluation metrics, this paper presents some of the main experimental results of the schemes.

Table 4. Differences between schemes of Sections 4.2–4.5 (NS: mention but not specify, NM: not mentioned).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Hang et al.	2020	NS:BFT/CFT/PBFT	×	Smart fish farming	Data quality	The average throughput of the system is 800 tps, 850 tps, and 875 s when the number of transactions reaches 200, 400, 500.
Lee et al.	2020	PoET, RAFT	×	Smart food farming	Monitoring, certification	-
Yang et al.	2018	NM	×	Smart toy	Security of data exchange	When the number of blocks is 50,100,416, the response time of the system is 130 s, 165 s, and 239 s.
Manzoor et al.	2020	NM	√	IOT mobile games	Incentives transparency	When the number of transactions is 10,000, the creation time is 250 tps, the query time is 500 tps, and the update time is 250 tps.
Pittaras et al.	2021	BFT	√	IOT mobile games	Cost and transaction delay	When the number of challenges is 60,100,140, the system response time is 0.3 s, 0.5 s, and 0.7 s.
Jamil et al.	2021	PBFT	×	Smart fitness	Intelligence, security	When the sending rate is 400,600,1000, the system throughput is 400 tps, 590 tps, and 950 tps.
Khan et al.	2020	NM	×	Smart city	Data security for CCTV (closed-circuit television camera) cameras	-

4.6. Smart Grid

A smart grid is an advanced digital bi-directional tidal power system that is self-healing, adaptive, resilient, and sustainable, with the ability to predict uncertainty [53]. Smart grids have high requirements for reliable, sustainable power supply [54], and secure two-way power transactions are an important factor in ensuring sustainable supply. Security includes reasonable privacy protection in addition to secure storage and traceability of transactions. Hyperledger-based research is focused on addressing power transactions, privacy protection, and energy consumption load.

In a smart grid system based on Hyperledger, a real-time scheduling strategy is an important part of power trading. Zhao [55] developed a micro grid market model based on Hyperledger (Fabric 1.1). In Figure 26, it is shown that multiple chaincodes are used for real-time dispatching of power resources, and transaction records are stored on Hyperledger. The transaction price and volume are determined according to the Bayesian Nash equilibrium theory of incomplete information static game, which effectively reduces the purchase cost of electricity users, but cannot guarantee the systematicity in handling a large number of transactions. Li [56] proposed a two-way electricity trading system based on Hyperledger (Fabric 1.4.0). In Figure 27, it is shown that a real-time scheduling policy is developed for EVs through an iterative two-tier optimization-based charging and discharging policy, and chaincodes are used for scheduling transactions and clearing. The structure of hierarchical power scheduling helps to improve the scalability of the system. Considering the stability of transactions, Li [57] proposed a power scheduling scheme (Fabric 1.4.0). In Figure 28, it is shown that the charging/discharging schedule for electric vehicles is developed based on an optimization model with an improved krill swarm algorithm, which minimizes the load variance of the grid and thus improves the security and stability of the electricity trading of electric vehicles. In power trading, a reasonable bidding strategy helps in power dispatching. Yu [58] proposed a power trading model based on Hyperledger (Fabric). In Figure 29, it is shown that the best bid strategy is provided to users by improving the Bayesian bidding algorithm, including the possible bid types, the best bid, and the probability distribution of the adversaries. A three-layer structure of user layer, agent layer, and Hyperledger layer is used to ensure that detailed transaction information is not accessed by agents and Hyperledger. To address the supply chain imbalance of users due to over scheduling, Lohachab [59] discussed a novel framework for electrical energy transactions (Fabric 1.4.0, Fabric 1.4.1). Instead of centralized microgrid scheduling of electricity, real-time scheduling of the dispatching of the Hyperledger is used. Reward algorithms and scheduling algorithms are designed to encourage users to sell excess electricity, maintain the demand balance of electricity, and guarantee the energy level of each user between minimum and maximum demand. This solution improves the utilization of electricity to a certain. extent.

To ensure the stability of energy trading in different periods, Jamil [60] proposed a smart power trading platform by combining machine learning and Hyperledger (Fabric 1.2). In Figure 30, it is shown that customer information is collected based on the physical network, and machine learning is used to analyze data characteristics and predict short-term and long-term scheduling transactions. The network load is effectively ensured, but a single metric is predicted. To better alleviate network congestion during power system peaks, crowdsourcing the transaction is an effective solution. Sciume [61] proposed an energy consumption load response scheme (Fabric). In Figure 31, it is shown that the network load reduction transaction is crowdsourced to users by predicting the next day's network load through a data hub. The actual load capacity of each user involved in reducing the power network load is evaluated, based on the baseline, using a smart contract, and the corresponding reward task is assigned to effectively solve the network congestion caused by peak loads in the power system.

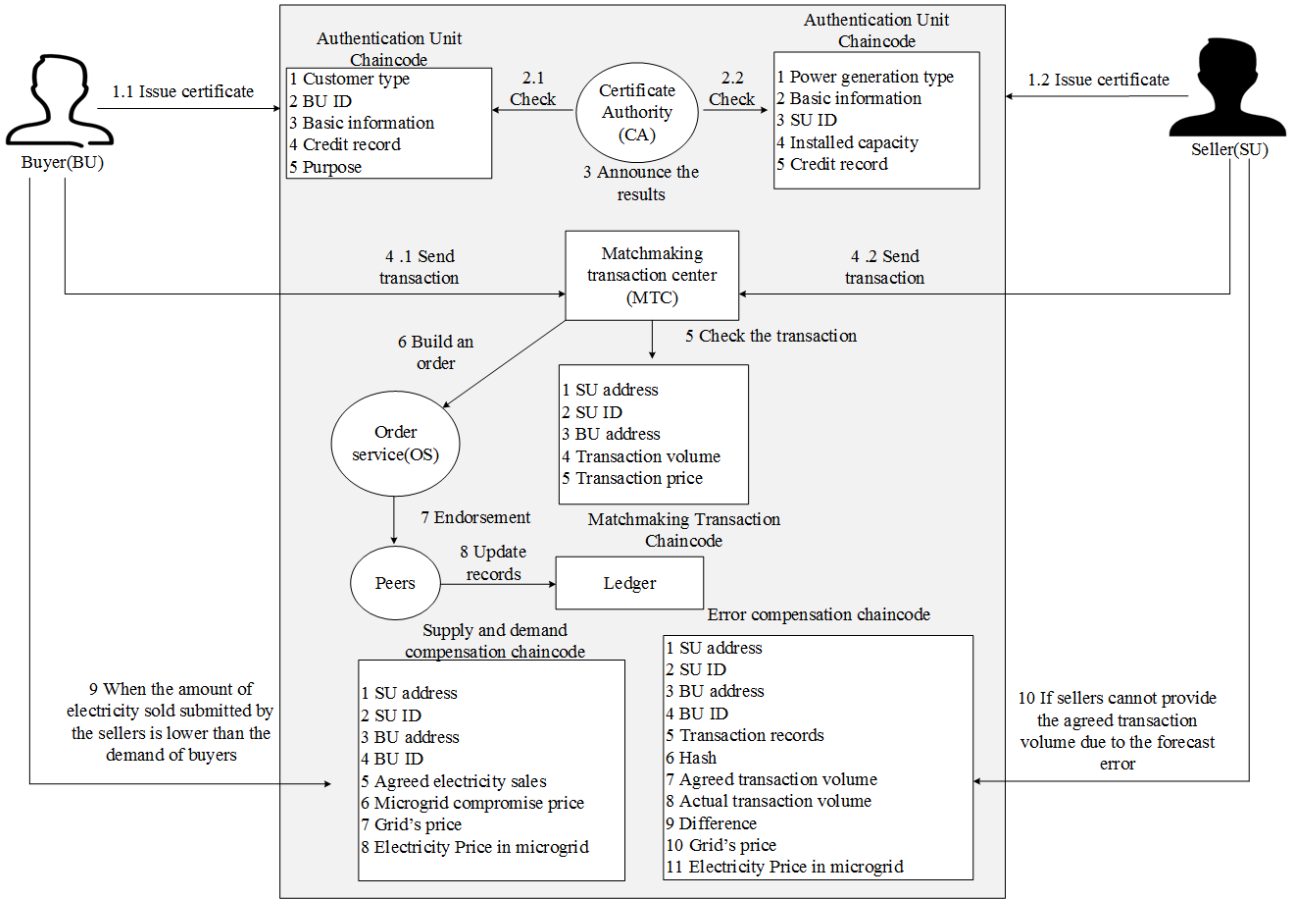

Figure 26. Reconstruction diagram of reference [55].

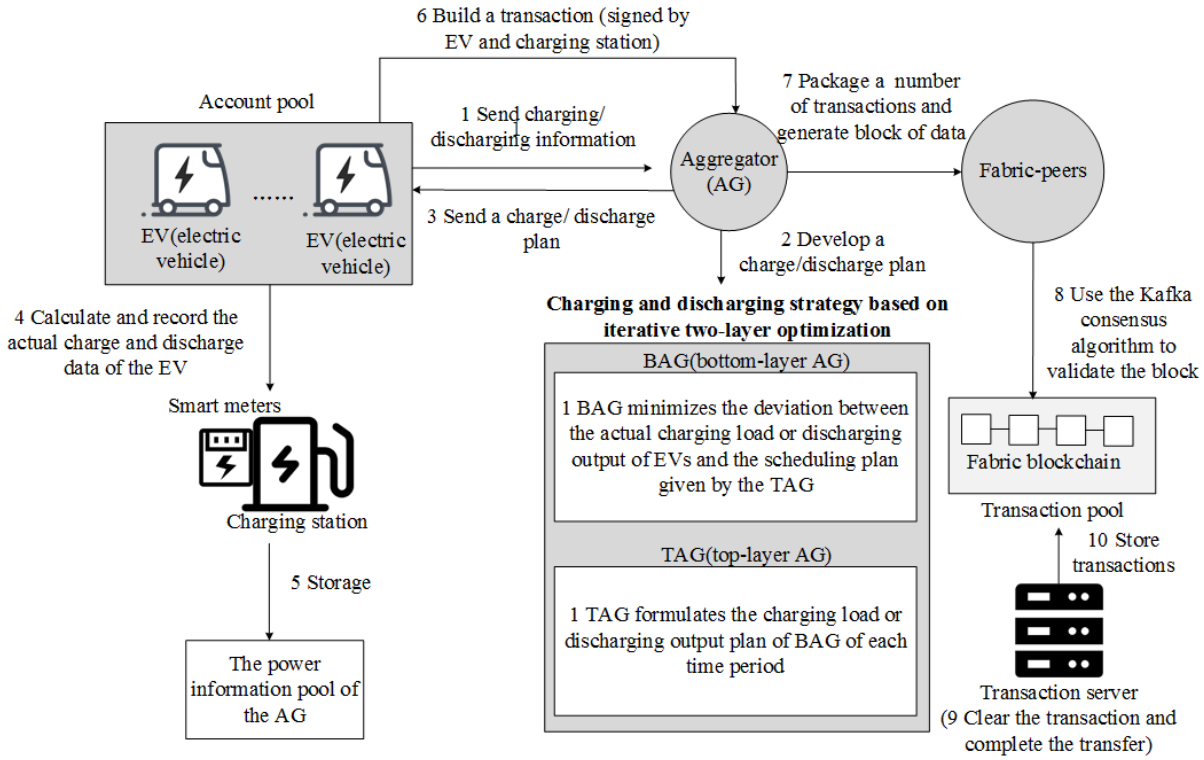

Figure 27. Reconstruction diagram of reference [56].

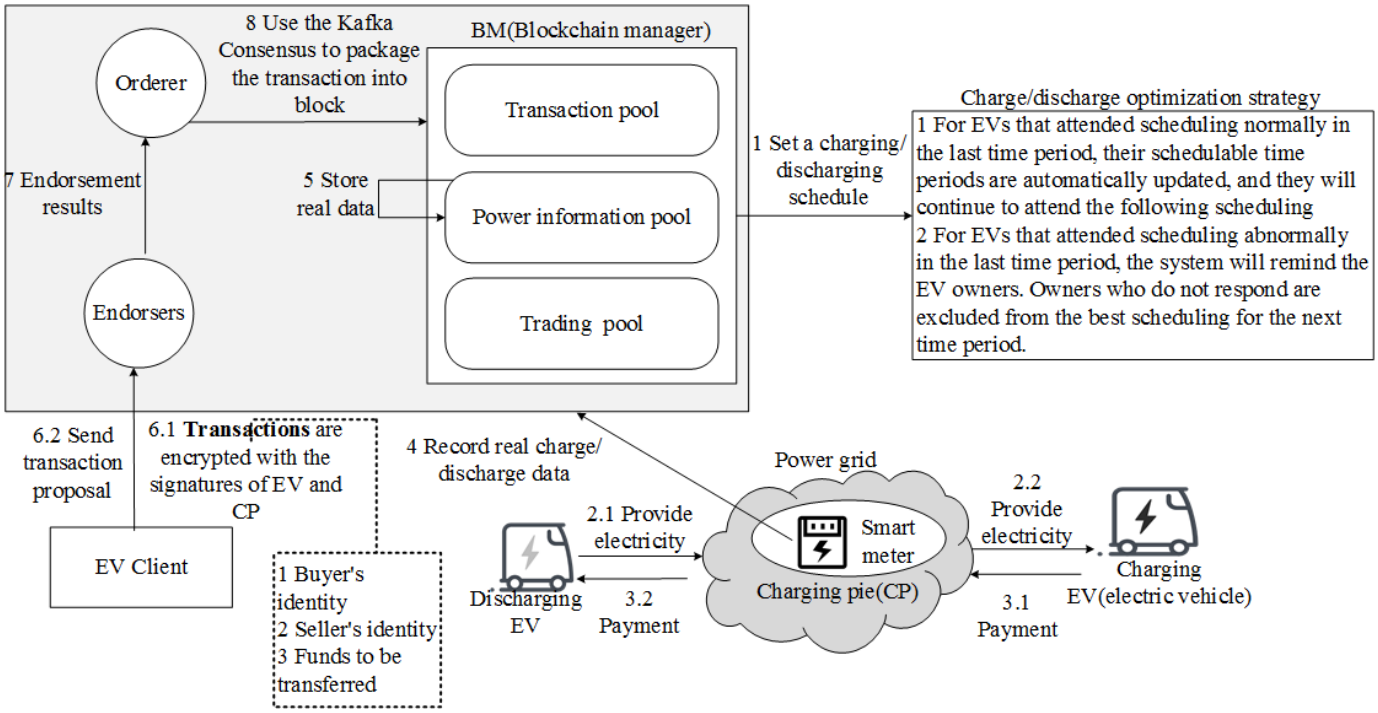

Figure 28. Reconstruction diagram of reference [57].

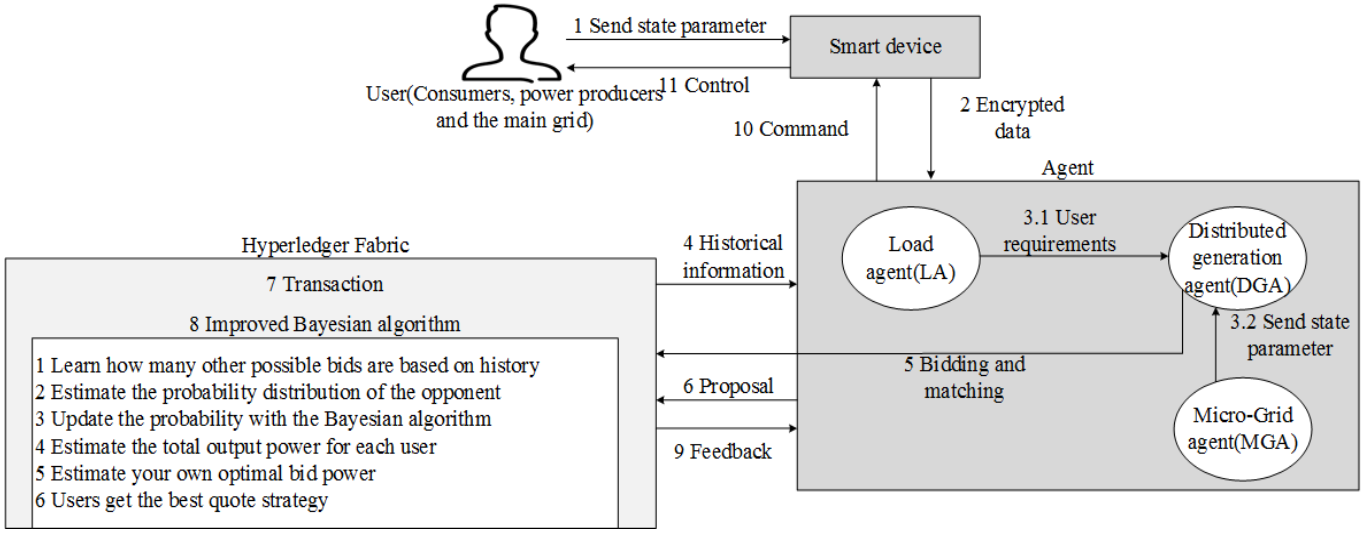

Figure 29. Reconstruction diagram of reference [58].

In smart power trading, regarding user privacy protection, Wang [62] proposed an electrical energy management system. In Figure 32, it is shown that an authentication method combining entity mapping protocol and zero-knowledge proof is used to separate user information and ensure the privacy of users.

In Table 5, this paper provides a comprehensive comparison of smart grid schemes. The table compares their differences in six aspects: year, consensus algorithm, incentive mechanism, application domain, issued addressed, and performance evaluation. In the performance evaluation metrics, this paper presents some of the main experimental results of the schemes.

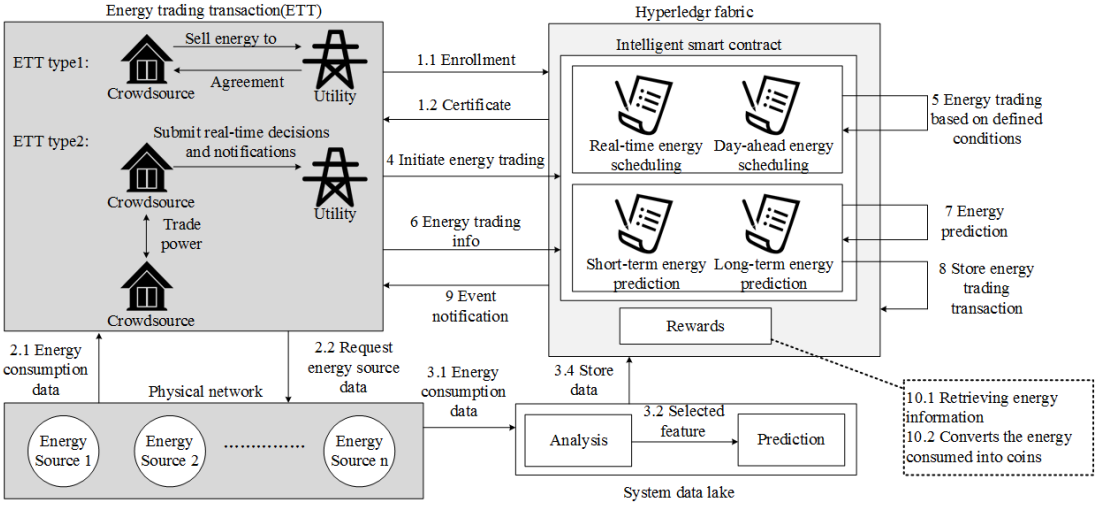

Figure 30. Reconstruction diagram of reference [60].

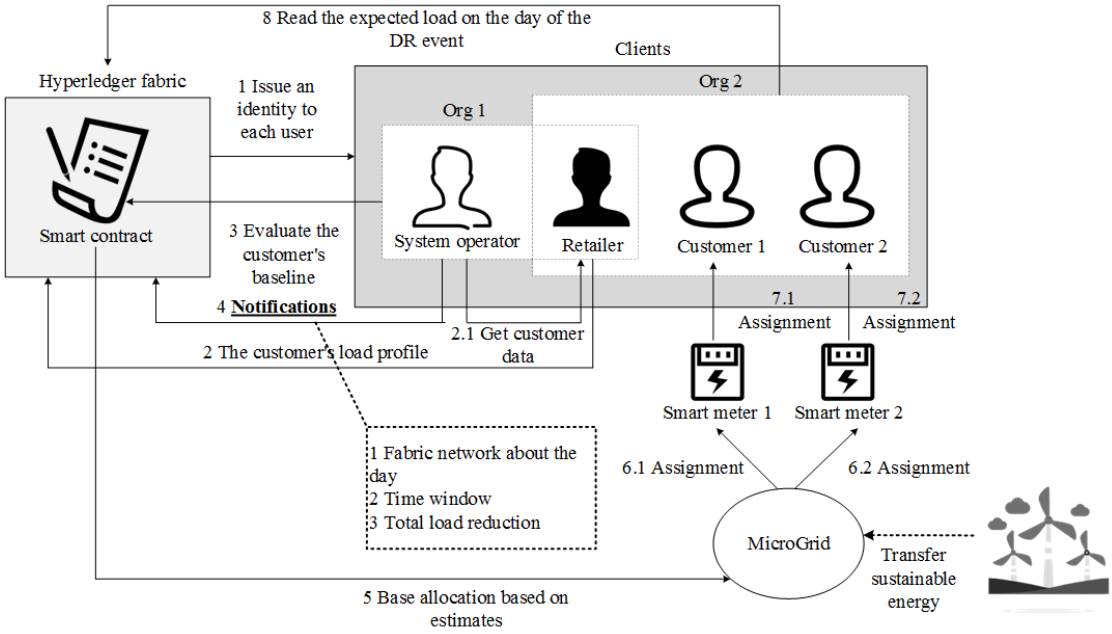

Figure 31. Reconstruction diagram of reference [61].

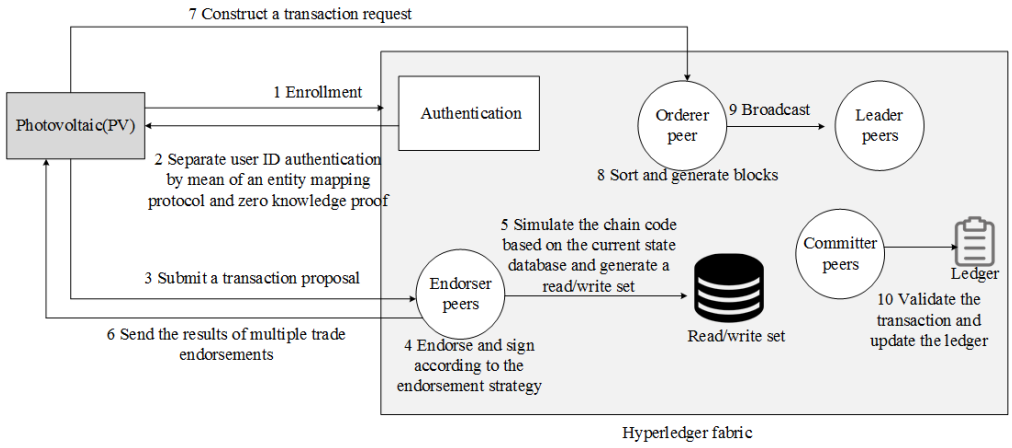

Figure 32. Reconstruction diagram of reference [62].

Table 5. Differences between schemes of smart grid (NS: mention but not specify, NM: not mentioned).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Zhao et al.	2019	NS: Kafka/SBFT(simple Byzantine fault tolerance)/Solo	×	Microgrid trading	Pricing strategy	When the transmission rate is 100,300,460, the average system delay is 3000 s, 5900 s, and 8000 s.
Li et al.	2020	Kafka	×	Power trading for charge and discharge	Transaction scheduling, mixed-integerprogramming (MIP)	When the number of concurrent transactions per second is 350, the system's minimum transaction confirmation time is 366 ms.
Li et al.	2021	Kafka	×	Power trading for charge and discharge	Security and privacy, MIP	With 1000 concurrent transactions per second, the system has a throughput of 500 tps or more, outperforming Ethereum and Bitcoin systems.
Yu et al.	2019	NM	×	Peer-to-peer (P2P) power trading	Competitive trading	When the power demand is 160 MW, the revenue is around USD 443.77.
Lohachab	2021	NS: Kafka/Raft/Solo	√	Power transactions supported by CPS (Cyber-Physical Systems)	Performance issues and optimized configuration	When the asset size is 1000 bytes, 8000, 32,000, the system query performance is around 570 tps, 400 tps, and 200 tps.
Jamil et al.	2021	PBFT	√	Trading for distributed energy resources	Predict short-term energy consumption	Maximum throughput of 43 tps, 68 tps, and 95 tps for user groups of 500, 1000, and 1500.
Sciume et al.	2020	Solo	√	Power consumption demand response	Network load	The average system downloading energy file is 10.2 s, the average query downloading files is 0.065 s, and the average time to get the reward is 2.3 s.
Wang	2021	NS: Kafka/SBFT/Solo	×	Microgrid energy management	Privacy protection of microgrid	The annual staff salary cost is approximately USD 15,000 and the staff cost savings from the energy management approach is USD 30,000/year.

4.7. Smart Transportation

Smart traffic is the development of big data-driven intelligent traffic management solutions that harness the potential for artificial intelligence to enable effective decision making [63]. Most of this decision making refers to the effective avoidance, mitigation of traffic congestion, and traffic accidents [64]. Making fast and accurate regulations in the face of highly mobile and dynamic traffic situations becomes an urgent challenge to be solved. The research of Hyperledger in the field of smart transportation covers several aspects, including automatic authentication, intersection regulation and monitoring, ETC (electronic toll collection), air–land integrated authentication, and connected vehicle data security.

In a Hyperledger-based vehicle system, regarding real-time authentication, Feng [65] proposed an automatic authentication vehicle information system based on Hyperledger (Fabric, Composer 0.20.7). In Figure 33, it is shown that the on-board unit is used as the unique identity of the vehicle, and the roadside unit and the on-board unit are used for real-time detection and automatic authentication by chaincode. Among them, the vehicle's identity is encrypted during the authentication process, which improves the privacy of authentication. To address cross-domain identity authentication, Li [66] proposed a vehicle location-aware system based on Hyperledger (Fabric 1.2, Ursa). In Figure 34, it is shown that the I-SIG system is used to obtain the data of the vehicle, providing the optimal signal scheme for the intersection. Encryption of the vehicle information was achieved using the ZKPR (zero knowledge range proofs protocol), and finally verified the legitimacy of the vehicle identity through the intelligent gateway. It has better advantages in terms of transaction latency, throughput, and success rate. Due to the limited monitoring range of roadside units, some schemes are dedicated to combining air resources for monitoring. Luo [67] proposed an air–land integrated vehicle cross-domain identity monitoring system (Indy). In Figure 35, it is shown that, using USRP (universal software radio peripheral) technology to provide the identity of the vehicle, the vehicle identity is authenticated by the UAV (unmanned aerial vehicle), and the legitimacy of the UAV's identity is ensured by using the cross-authentication of neighboring UAVs. The use of airborne nodes extends the monitoring range, but the latency of authentication is high.

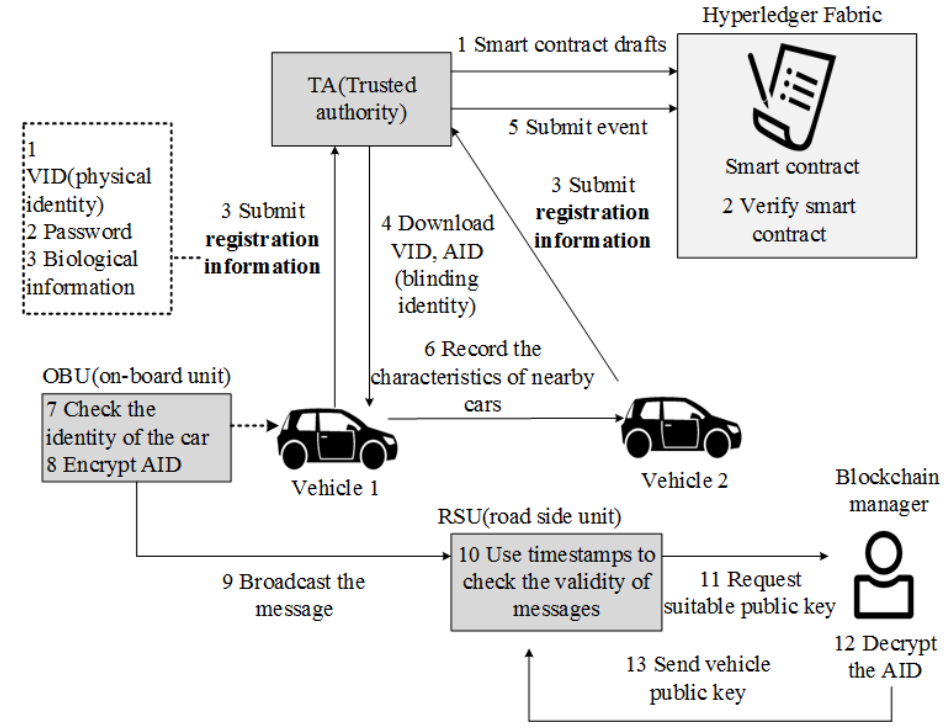

Figure 33. Reconstruction diagram of reference [65].

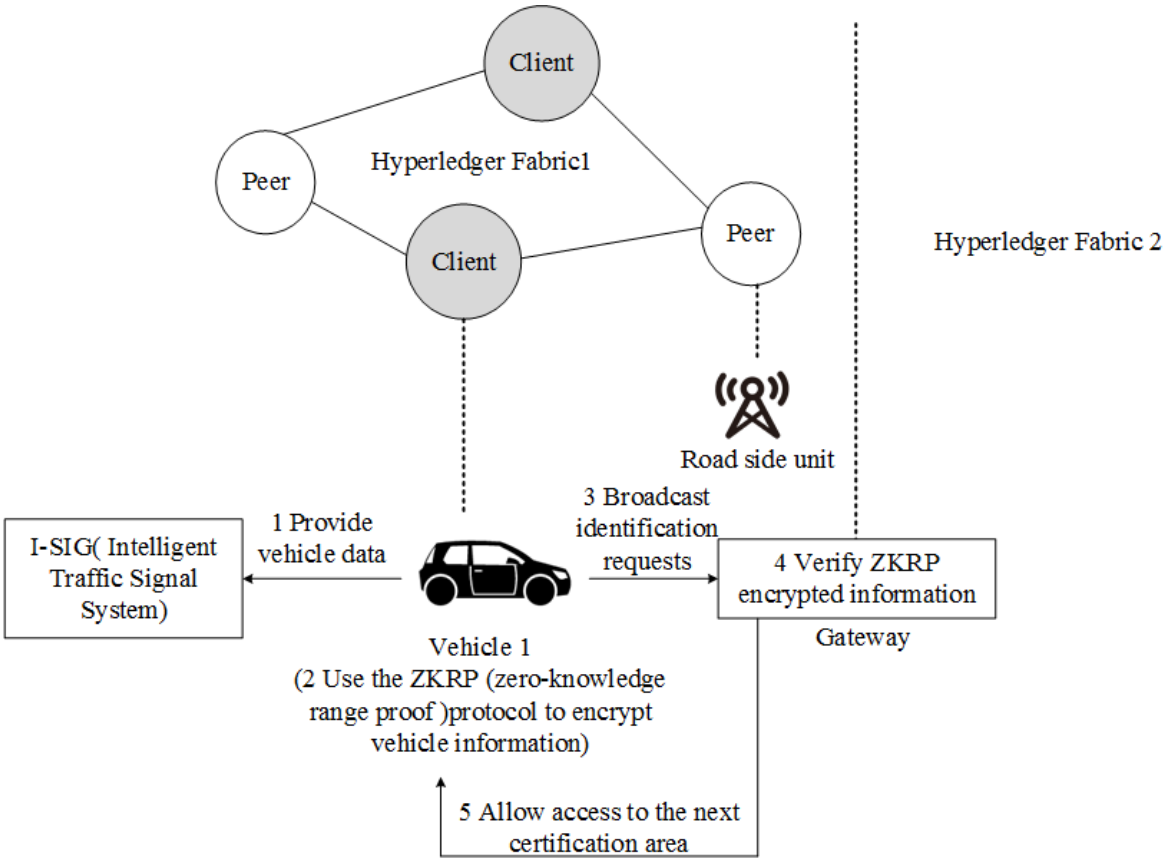

Figure 34. Reconstruction diagram of reference [66].

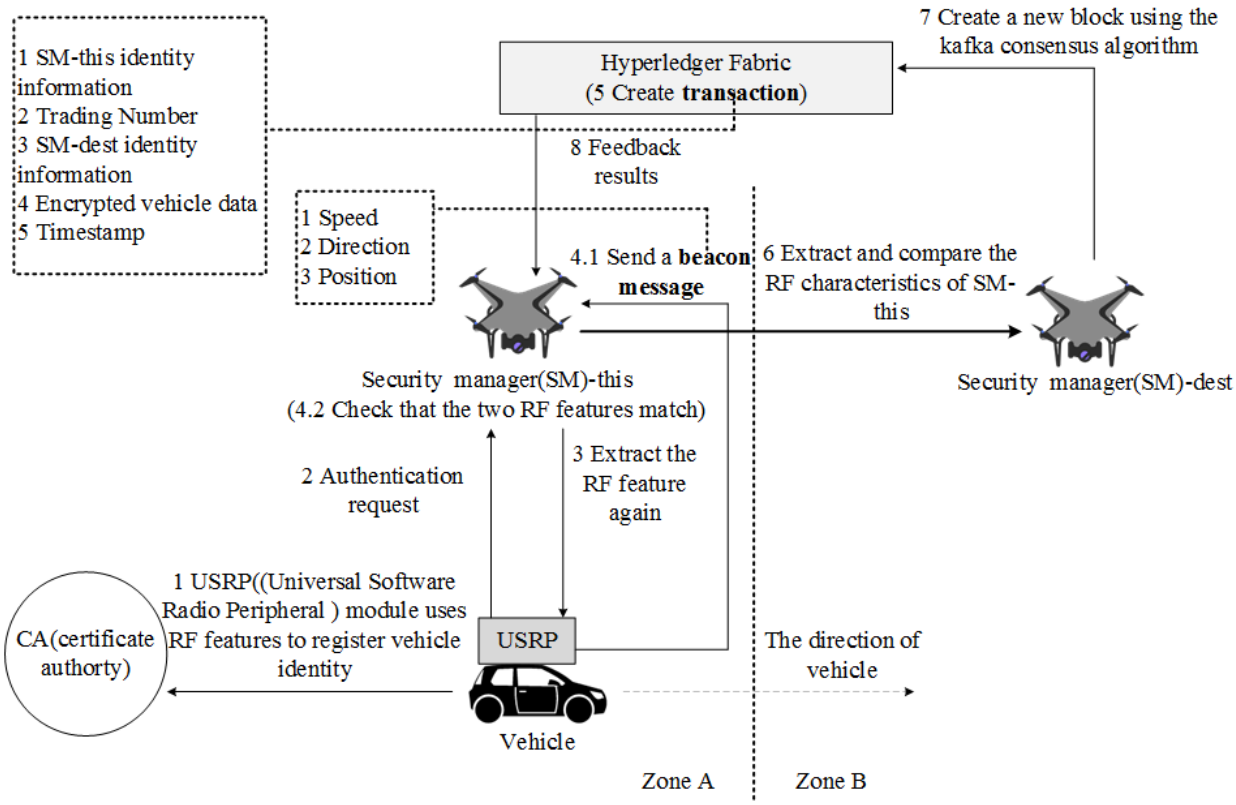

Figure 35. Reconstruction diagram of reference [67].

To address the traffic safety problem at intersections, Buzachis [68] proposed a system for monitoring vehicles at intersections based on Hyperledger (Fabric). In Figure 36, the trajectory of the vehicle is simulated in real time by chaincode and the endorsement node is responsible for detecting the simulation results. The usability of the system is demonstrated by testing self-driving vehicles through intersections at 1–2 intersections, but there is no design for multiple intersections. To address the problem of real-time assistance in case of vehicle hazards, Mbarek [69] proposed a multi-level endorsement vehicle communication system (Fabric). In Figure 37, it is shown that the BF-DF-AF-IF (belief function–desire function–analysis function–intention function) model is used to refine the vehicle's needs into specific repair action needs. Endorsement level mechanisms are designed (according to the score obtained by the exchange, chaincode automatically upgrades or downgrades the endorsement level), and each transaction is endorsed by a higher-level endorsement node to ensure the reliability of the transaction. An intelligent endorsement mechanism is realized to enhance the efficiency of endorsement, but the scoring mechanism is not complete. To address the authenticity of accident information in telematics, Xiao [70] proposed a telematics fake news detection model (Fabric). In Figure 38, a Bayesian algorithm is used to detect the probability of authenticity of telematics messages and stored in Hyperledger. Load balancing is achieved and its feasibility is demonstrated in terms of prior probability, transaction processing speed, and accuracy. To enable timely access to road conditions and avoid traffic accidents, Chen [71] proposed an edge server-based vehicle area information auction scheme. In Figure 39, an edge server is used to divide the area and issue a request task for information reporting in a certain area. The vehicles completing the task are identified by the road side unit (RSU) technology and the authenticity information is evaluated using the expectation maximization (EM) algorithm. Suitable for low-power devices, it ensures data quality and rewards.

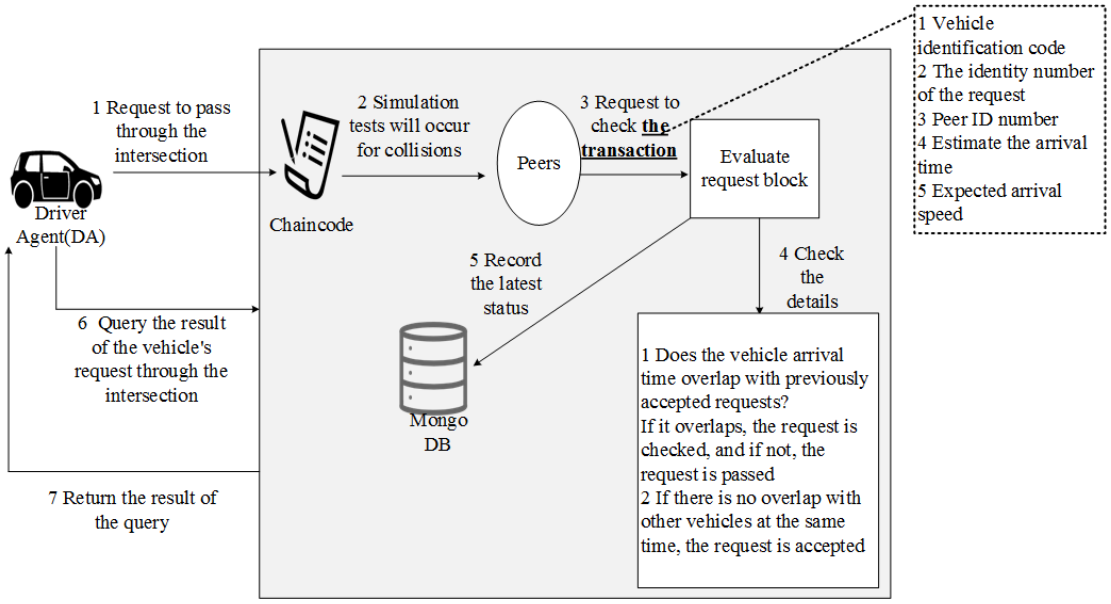

Figure 36. Reconstruction diagram of reference [68].

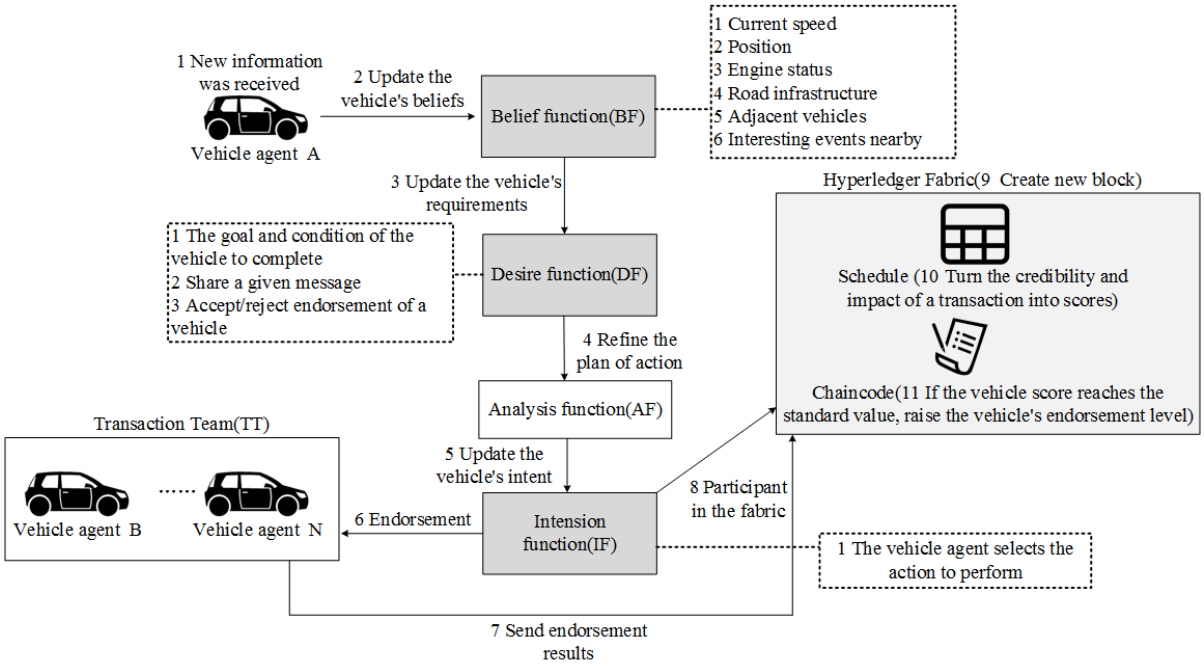

Figure 37. Reconstruction diagram of reference [69].

Several studies are devoted to the problem of secure transactions in smart transportation. Gao [72] proposed a V2G (vehicle-to-grid) payment model based on Hyperledger (Fabric 0.6). In Figure 40, it is shown that privacy is ensured by the ability of the payers to create multiple accounts in the same transaction. Chiu [73] proposed an ETC system based on Hyperledger (Fabric 2.2). In Figure 41, it is shown that the vehicle is cross-authenticated with the ETC gate, which detects the legitimacy of the vehicle's identity and stores the transaction records in Hyperledger. It has stability and high performance, but PBFT (practical Byzantine fault tolerance) consensus is not applicable to large networks. In the toll station system, to solve the problem of electronic identity, Viera [74] proposed a 5G-based C-V2X (vehicle-to-everything) road tolling system (Fabric). In Figure 42, it is shown that Indy's portable identity technology is used to send identity information through smart-

phones instead of RSU to obtain identity information. Toll requests are processed and transaction records are stored via cell phones. This proposal demonstrates for the first time the feasibility of combining 5G with Hyperledger in a V2X system. Lee [75] proposed a traffic system (Fabric) based on an auction mechanism and fog computing. In Figure 43, it is shown that fog computing is used to allocate public transportation resources, and an auction mechanism is designed to select the highest bidder for the connected vehicle user. A rational allocation of public transportation resources is achieved, but the winner is selected in a single way. In addition, the neighboring RSU nodes are secure by default, which reduces the credibility of the endorsement results.

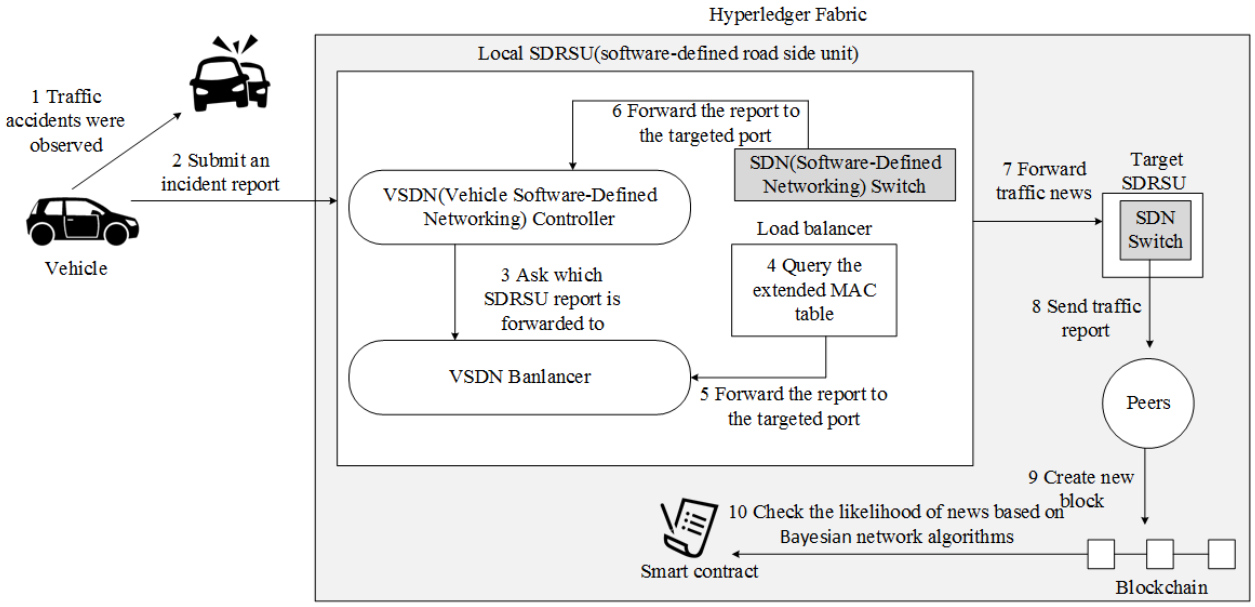

Figure 38. Reconstruction diagram of reference [70].

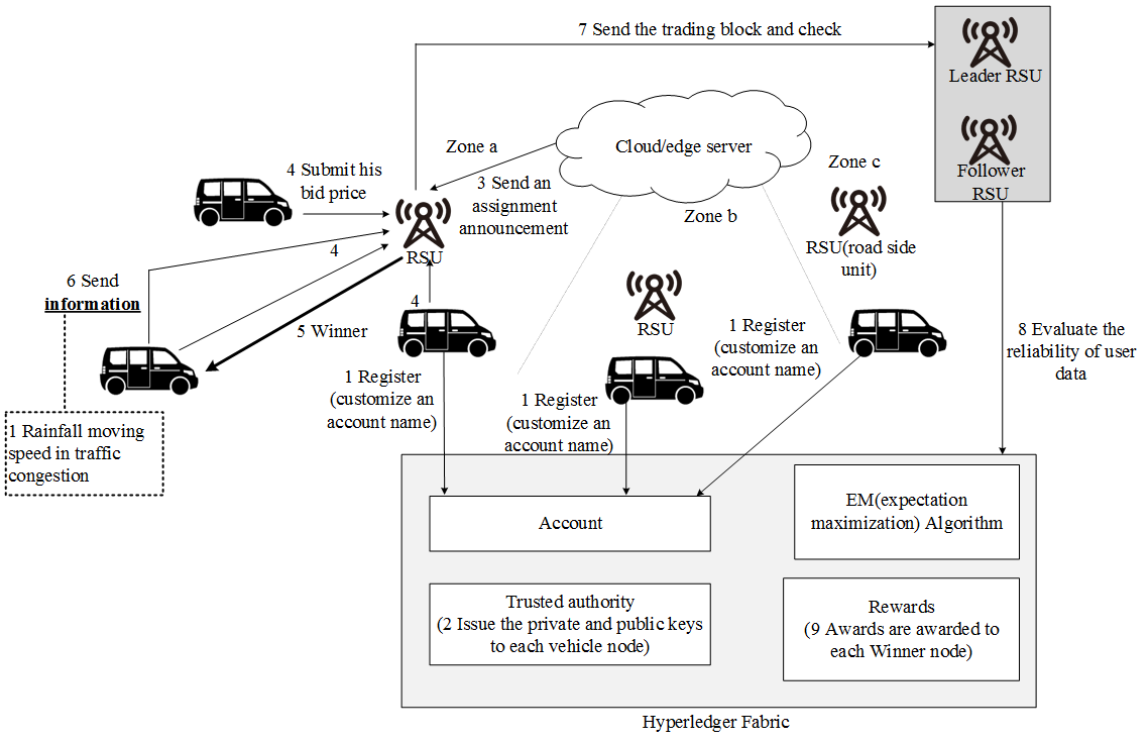

Figure 39. Reconstruction diagram of reference [71].

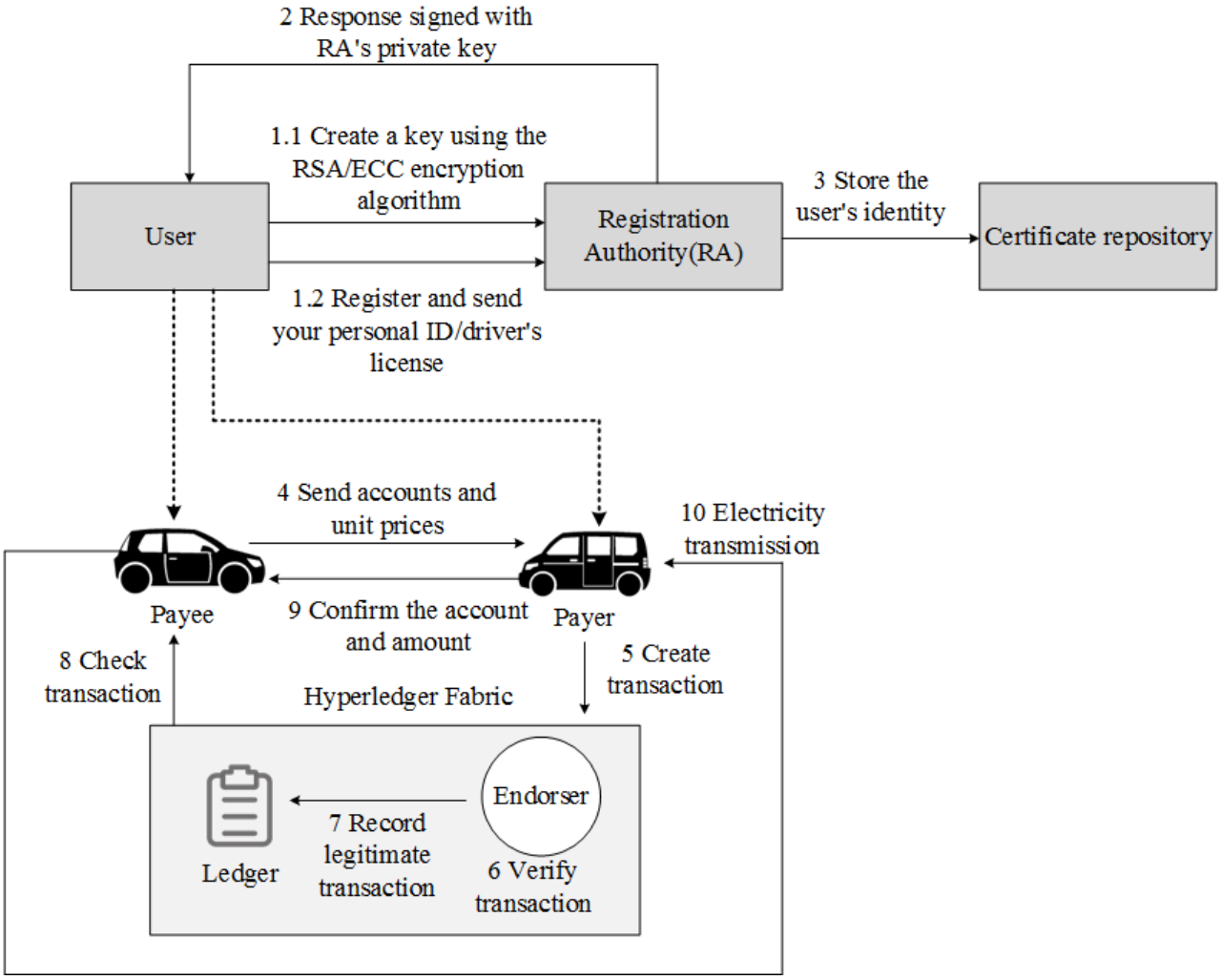

Figure 40. Reconstruction diagram of reference [72].

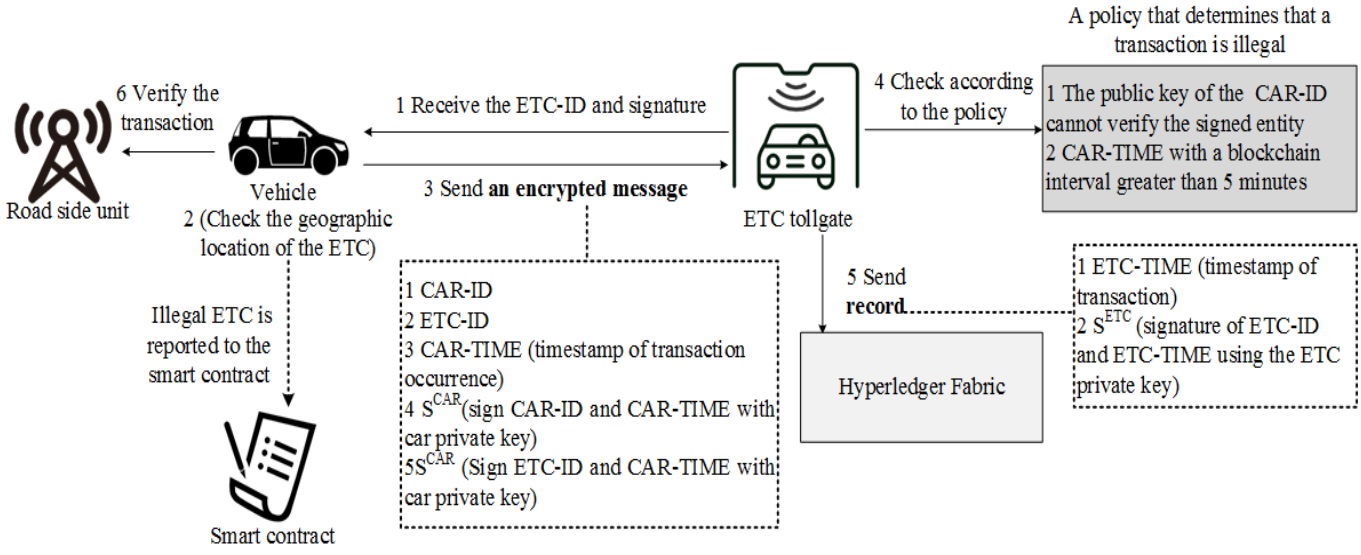

Figure 41. Reconstruction diagram of reference [73].

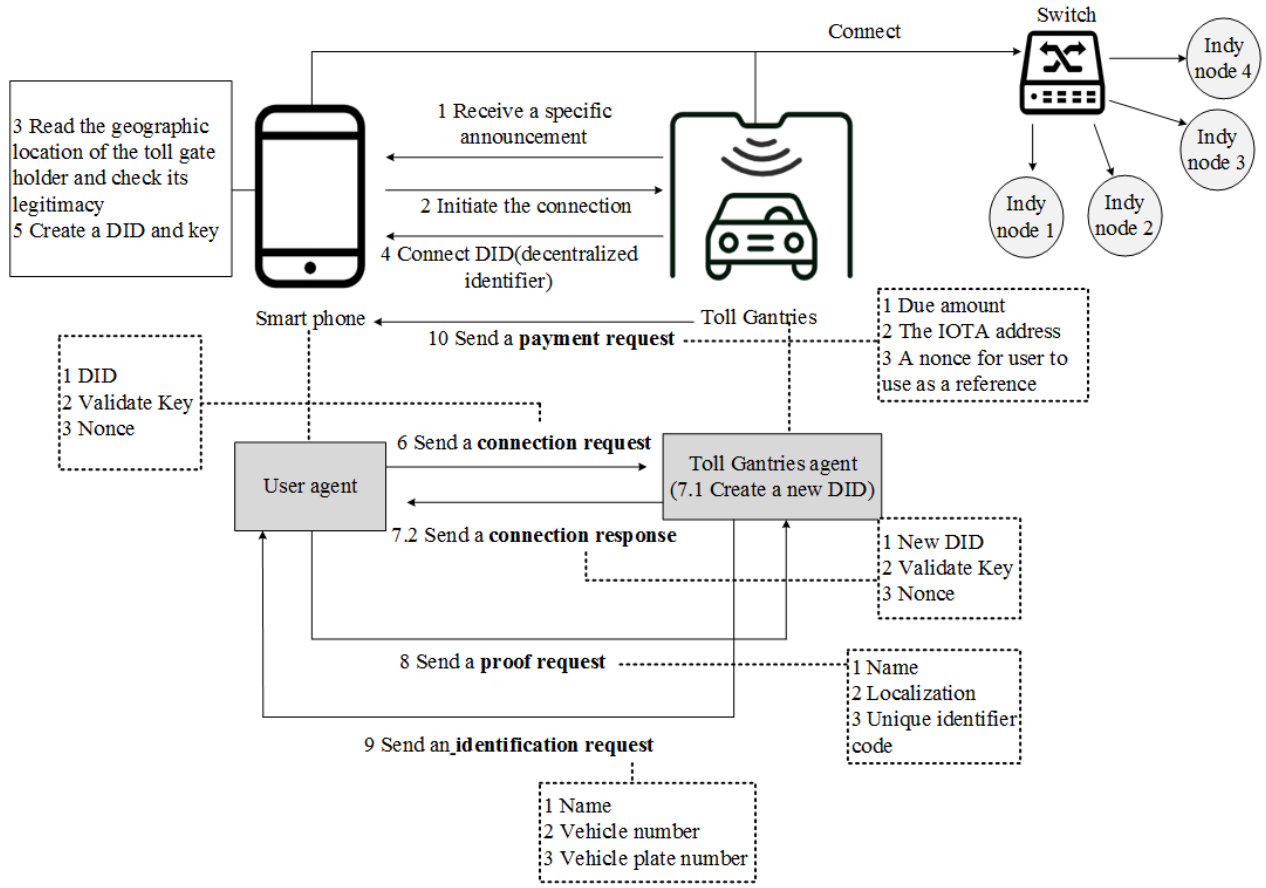

Figure 42. Reconstruction diagram of reference [74].

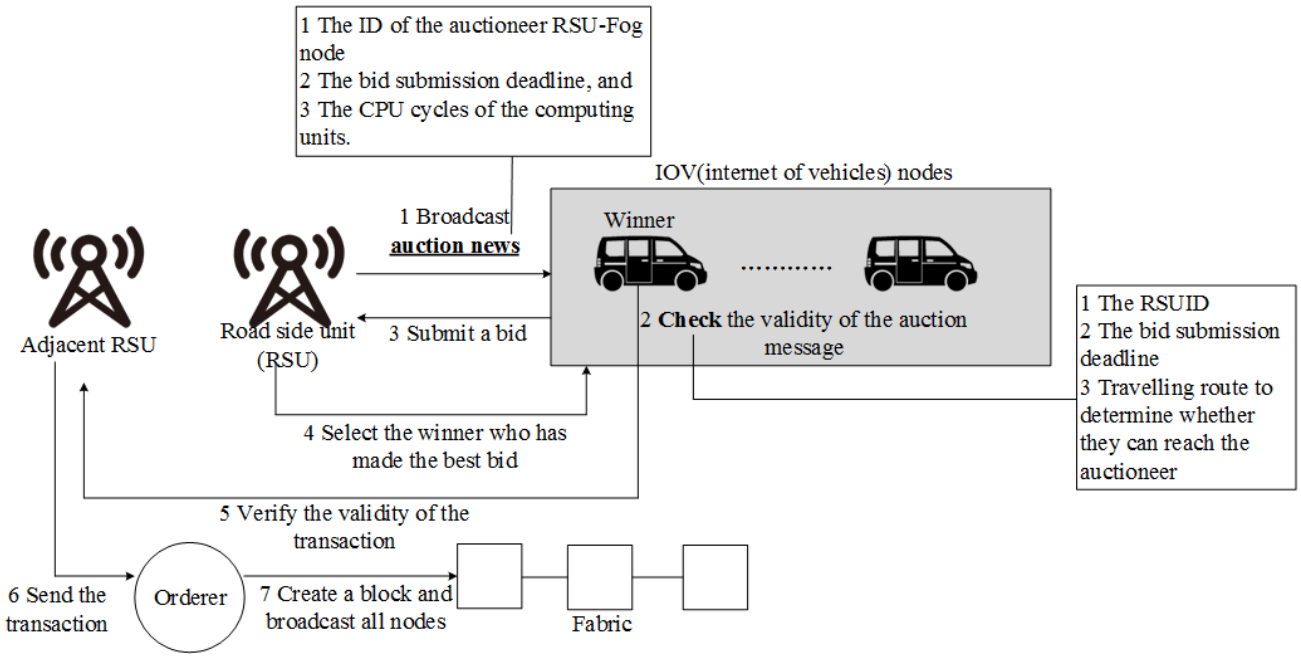

Figure 43. Reconstruction diagram of reference [75].

In Tables 6 and 7, this paper provides a comprehensive comparison of smart transportation schemes. The table compares their differences in six aspects: year, consensus algorithm, incentive mechanism, application domain, issue addressed, and performance evaluation.

In the performance evaluation metrics, this paper presents the main experimental results of the schemes.

Table 6. Differences between schemes of smart transportation (NS: mention but not specify, NM: not mentioned).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Feng et al.	2019	PBFT (Practical Byzantine Fault Tolerance)	×	Vehicular ad hoc networks (VANETs)	Automatic authentication	Point addition, scalar multiplication, multiplication, the average time of these three operations are 0.184 ms, 64.99 ms, and 0.179 ms.
Li et al.	2020	NM	×	Traffic management systems	Privacy protection	Proof generation time and validation time remain at 98 ms and 97 ms.
Luo et al.	2020	PBFT, Kafka	×	Space–air–ground integrated network (SAGIN)	Cross-regional certification	Maximum request delay for Kafka message packaging, maximum number of packaged messages, maximum block capacity, and maximum message size are 2 s, 100, 99 mb, and 512 kb.
Buzachis et al.	2020	NM	×	Autonomous Intersection Management (AIM)	Collisions of AVs(Autonomous Vehicles) and traffic congestion	Response latency is about 0.7 s, average latency of about 0.72 s.
Mbarek et al.	2020	Proposed	×	Vehicular Ad-hoc Network (V ANET)	Communication overheads	For 250 transactions, the total execution time is 14 s, average execution time for transactions is 0.056 s.

Table 7. Differences between schemes of smart transportation (continue Table 6).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Xiao et al.	2020	POA	×	Vehicular ad hoc networks (VANETs)	Information security detection	When the batch size reaches 1000, the trading time is 20.246 s.
Chen et al.	2019	PoS(Proof of stake)	√	Internet of vehicles (IoV)	Incentive mechanism, data quality	For data vendors, USD 0.03045839 is required for a data sharing round.
Gao et al.	2018	Byzantine	×	V2G (vehicle-to-grid) networks	Privacy preserving payment mechanism	The saturation value of the system throughput is 300 TPS.
Chiu et al.	2021	PBFT	×	Electronic Toll Collection (ETC) system	Data storage, trustworthiness and transparency	1M of records has a trading time of 6 s.
Vieira et al.	2020	Proposed	×	5G C-V2X (cellular Vehicle-to-everything) communication	Privacy protection	The average time taken by the system is 1090.3 s and the certificate verification is 465.9 ms.
lee et al.	2020	Proposed	√	Vehicle-to-everything (V2X) communications	Service stability, data integrity	100 nodes, 200 nodes, and 300 nodes out of the block time is about 7.5 s.

4.8. Smart Construction Project

In this study, smart construction projects refer to the high integration of construction projects and cutting-edge IT technologies to achieve real-time updates in building modeling, transaction security, reduced delivery costs, and effective collaboration [76,77]. Most of the current research focused on solving the multiparty information exchange in construction-type projects.

To address the problem of information exchange in construction projects, Suliyanti [78] proposed a system for multiple interested parties to exchange construction information

(Fabric, composter). In Figure 44, it is shown that the system developed a system for bidding construction projects and stores the whole cycle of construction completion in Hyperledger. A complete record and exchange of building information modeling (BIM) information is explored, but the solution is owner centric, resulting in an owner's choice without an appropriate regulatory approach. To address the issue of financial allocation in construction projects, Elghaish [79] proposed a building information modeling system (Fabric). In Figure 45, it is shown that the system uses smart contracts to check the financial allocations of the construction team, and allocates the appropriate finances to each participant based on the net amount of total profit, cost savings and reimbursed costs. The scheme demonstrates the feasibility of applying Hyperledger to integrated project delivery (IPD) systems. To address the issue of the privacy of different construction project ledgers, Yang [80] discussed a multi-channel design scheme (Fabric). In Figure 46, it is shown that smart contracts enable communication between architects, suppliers, engineers, clients, building surveyors, and urban planners, as well as store information from each of them in different channels. This scheme identified advantages unique to Hyperledger-based construction project systems in terms of scalability, traceability, and auditability features, as well as challenges in terms of transaction processing efficiency, business changes, identity, cost, and security of smart contracts. To address the problem of incomplete information for construction projects, Sheng [81] proposed a construction project information management system based on Hyperledger (Fabric 1.4). In Figure 47, it is shown that the system checks the authenticity of construction information through the endorser of Hyperledger, and uses orderer to sort transactions, and queries the complete construction project information through the web. To a certain extent, it solves the problem of incompleteness and difficult traceability of construction project information.

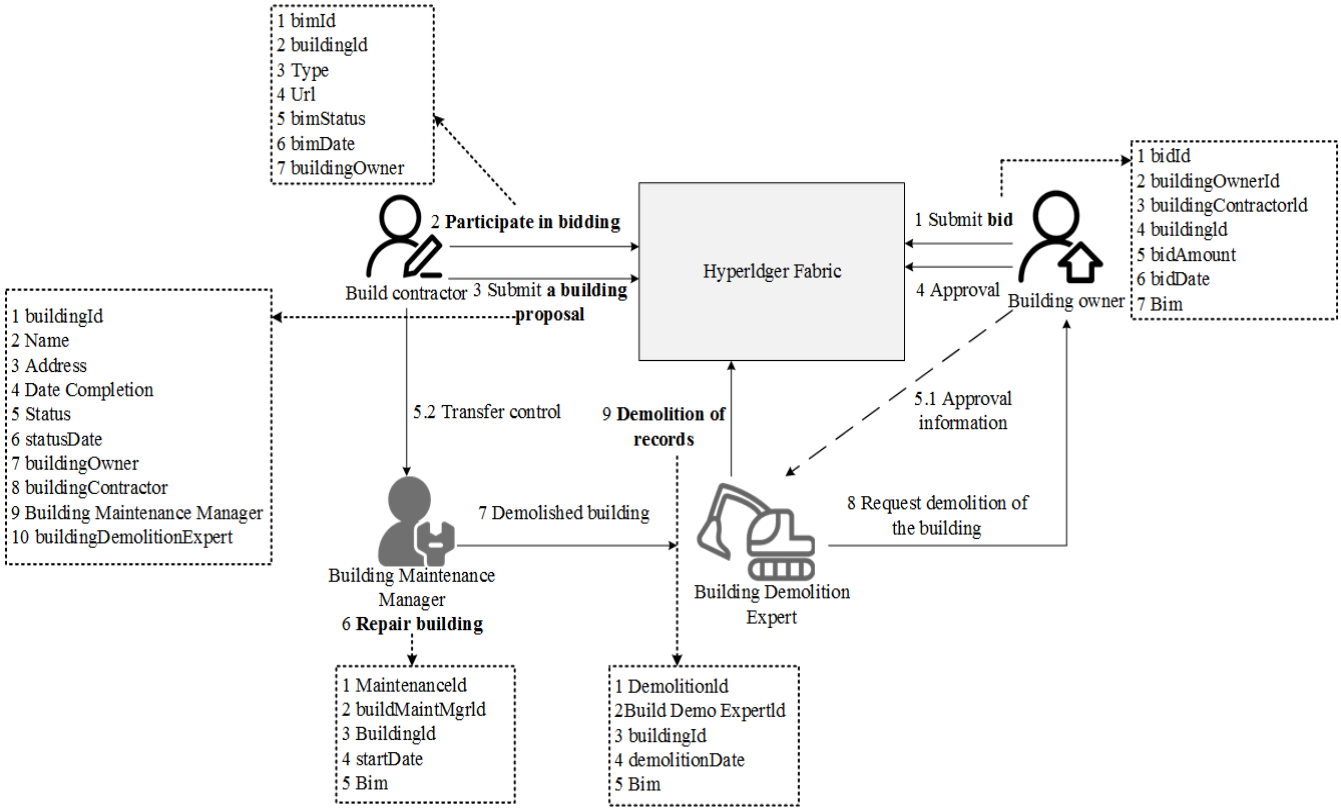

Figure 44. Reconstruction diagram of reference [78].

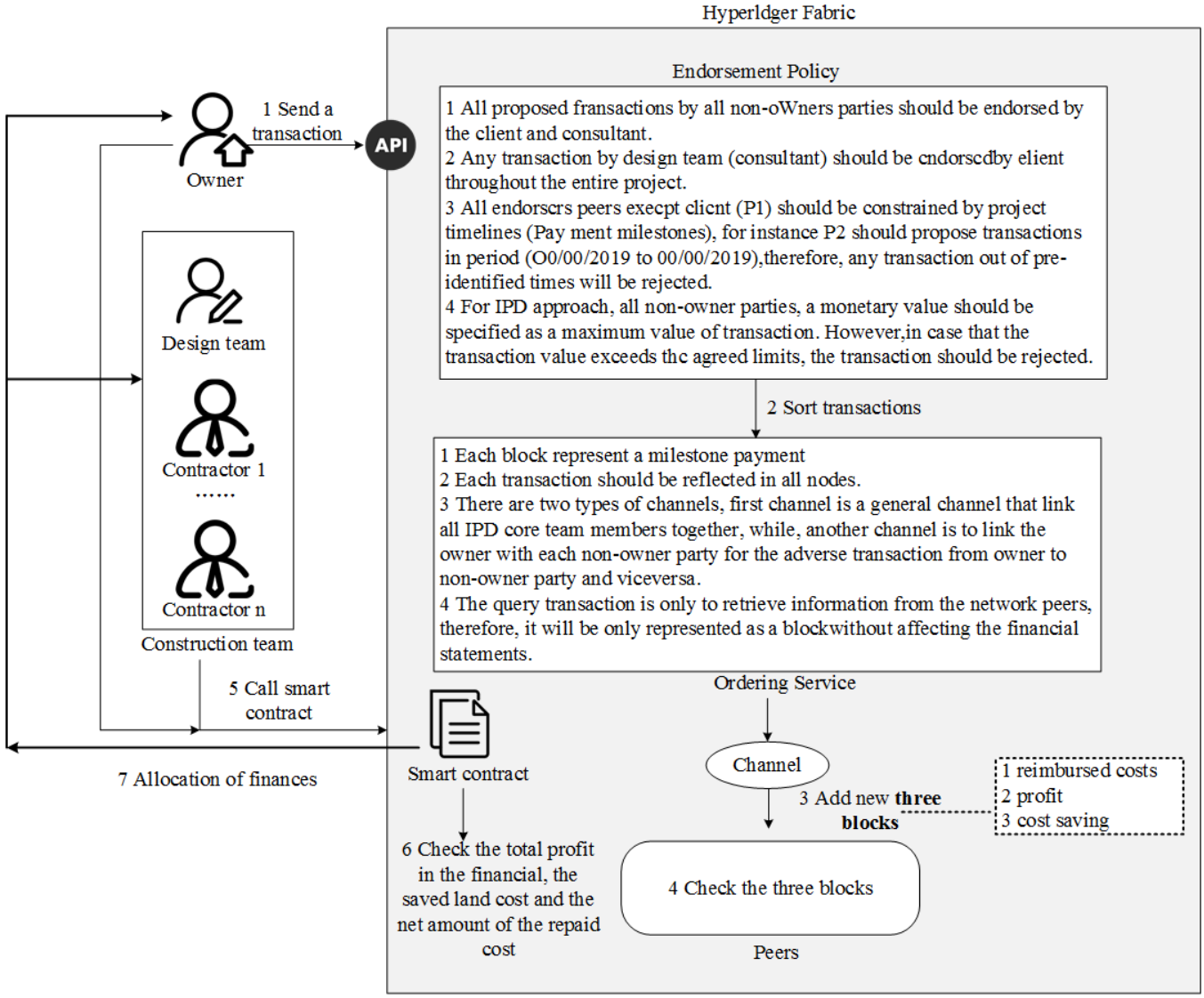

Figure 45. Reconstruction diagram of reference [79].

In Table 8, this paper provides a comprehensive comparison of the smart construction project' schemes. The table compares their differences in six aspects: year, consensus algorithm, incentive mechanism, application domain, issue addressed, and performance evaluation. In the performance evaluation metrics, this paper presents the main experimental results of the schemes.

Table 8. Differences between schemes of the smart construction project (NS: mention but not specify, NM: not mentioned).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Widya Nita Suliyanti et al.	2020	NM	×	Building information modeling (BIM)	Information interaction, monitoring status	The average response has a table value of about 8000 ms.
Faris Elghaish et al.	2020	NM	×	Integrated project delivery (IPD)	Automated financial	-
Rebecca Yang et al.	2020	NM	×	Construction projects	Efficiency, synergy	Transaction time is between 30 s and 60 s.
Da Sheng et al.	2020	Kafka	×	Construction projects	Management of quality information	The upload average latencies are 53.3 ms and the query average latencies are 57.8 ms.

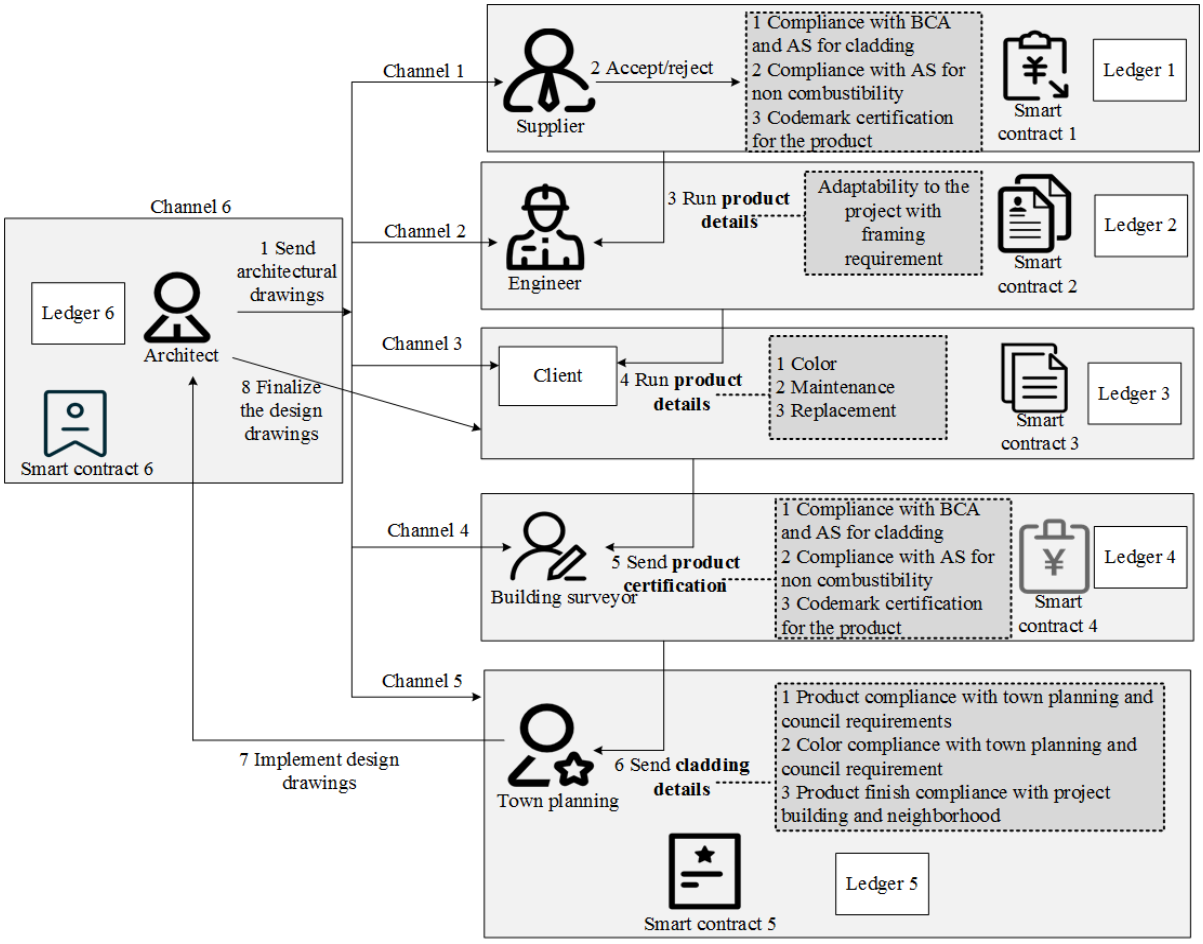

Figure 46. Reconstruction diagram of reference [80].

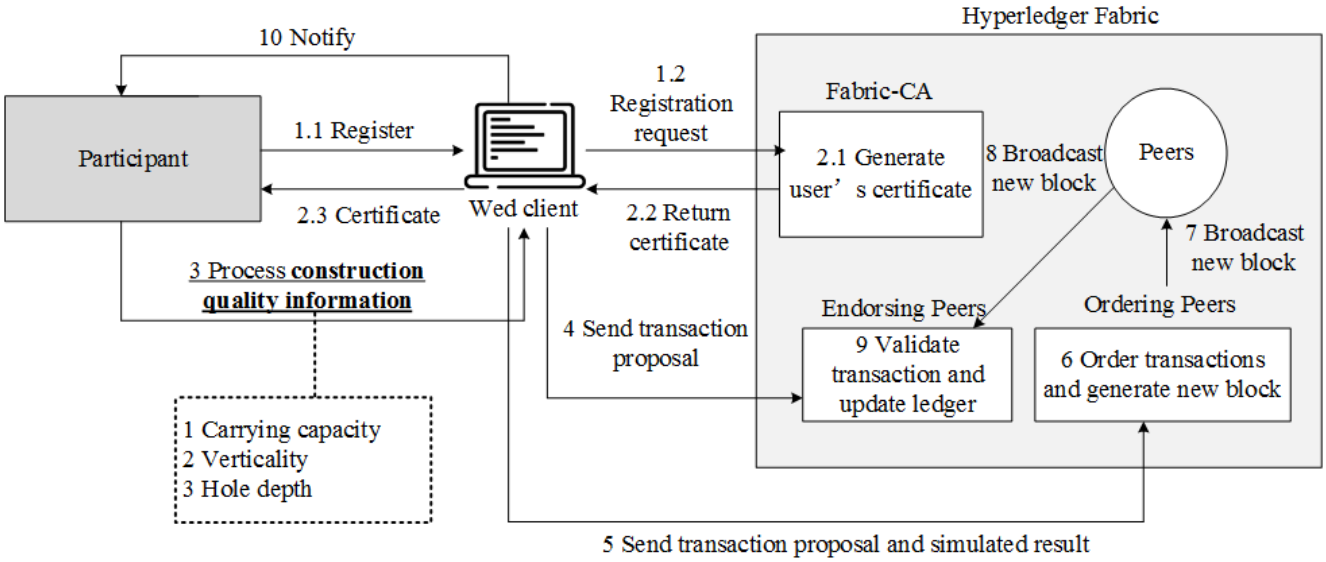

Figure 47. Reconstruction diagram of reference [81].

4.9. Smart Energy (Energy Saving Direction)

Smart energy trading aims to achieve autonomous energy regulation and efficient energy use by selling their surplus energy or buying the energy they need between consumers and businesses. The concept of energy in this section may be electrical energy or

carbon emissions. Smart energy trading improves the utilization of energy and reduces manual errors and management costs. Hyperledger mainly solves the problems of secure transactions and transaction integrity verification.

To address the secure scheduling of energy emissions, Yuan [82] proposed an energy emissions trading system based on Hyperledger (Fabric 1.1). In Figure 48, it is shown that nodes allocate emissions through specific channels and store and review energy emission transactions using smart contracts. To ensure the legitimacy of the energy emission trading identities, Hu [83] proposed a model of distributed energy trading (Fabric). In Figure 49, it is shown that the identity of the company and the requested emissions are verified by the endorser, and the transaction information is stored on the chain. To improve the validation efficiency of energy emission transactions, Che [84] proposed a scheme to jointly validate energy transactions (Fabric 1.1). In Figure 50, it is shown that a certain number of transactions are packaged and verified by the matching unit, and then re-verified and stored by the peer point on the chain. To improve the efficiency of energy dispatching, Silva [85] proposed an electric vehicle energy bidding system. In Figure 51, it is shown that the bidding of electric energy is designed using Hyperledger (Fabric, composter) and connected to the controller of the local parking lot for electric energy scheduling. The system uses chaincode to complete electricity transactions, which has better advantages in terms of the integrity and transparency of transactions, but the buyer is close to centralized in the transactions, and there are obvious shortcomings in regulating the buyer.

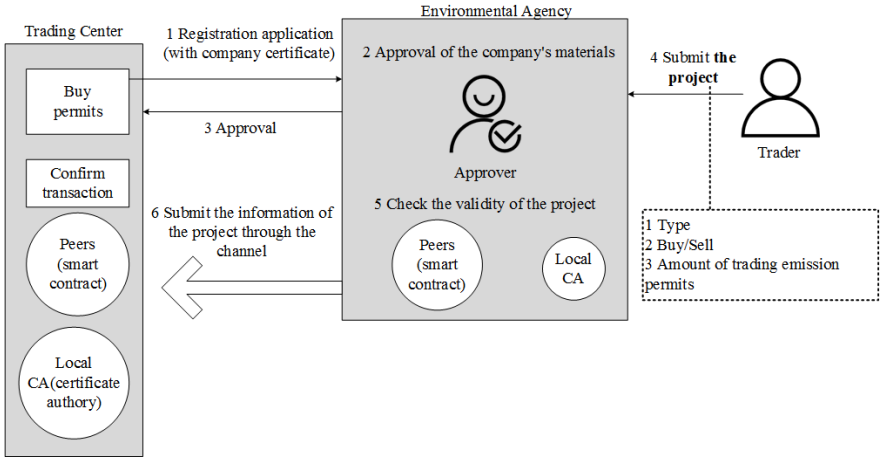

Figure 48. Reconstruction diagram of reference [82].

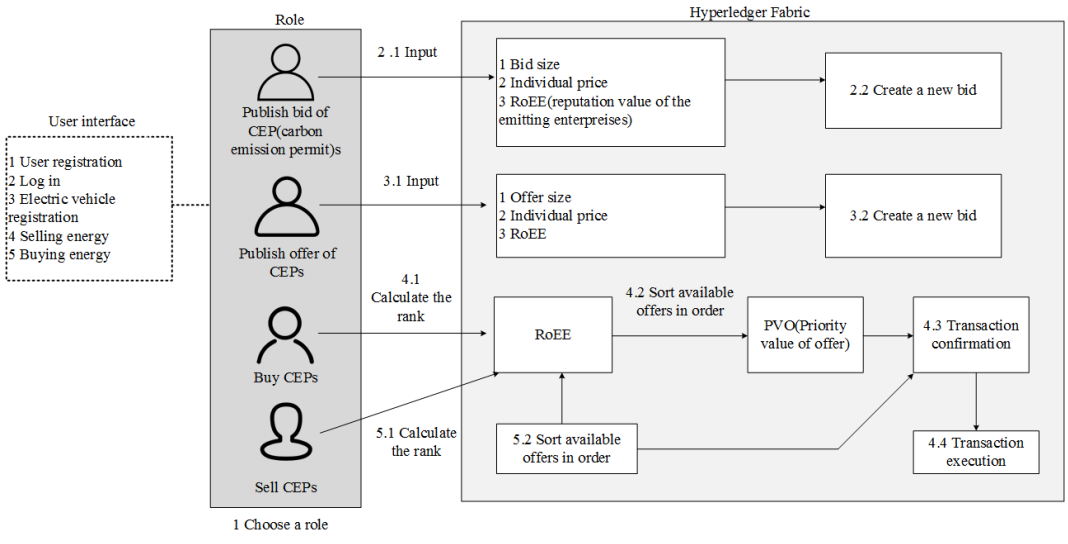

Figure 49. Reconstruction diagram of reference [83].

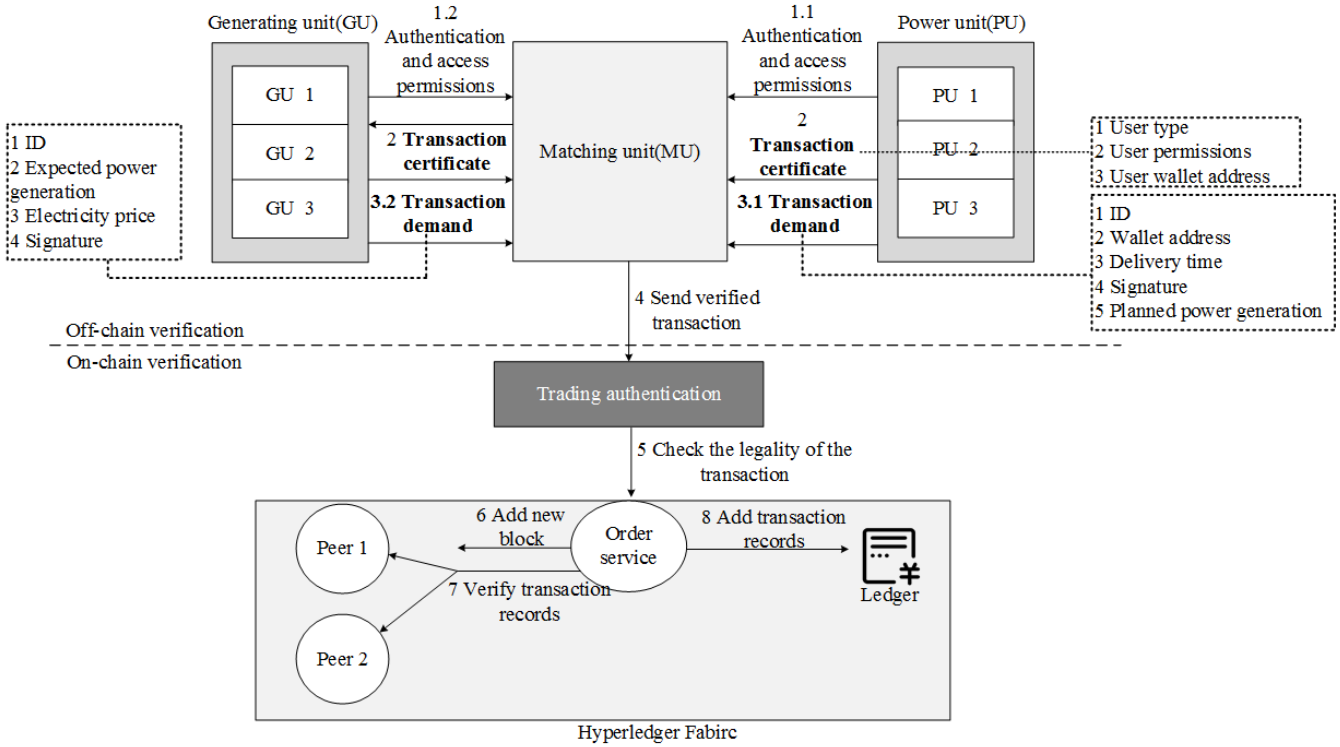

Figure 50. Reconstruction diagram of reference [84].

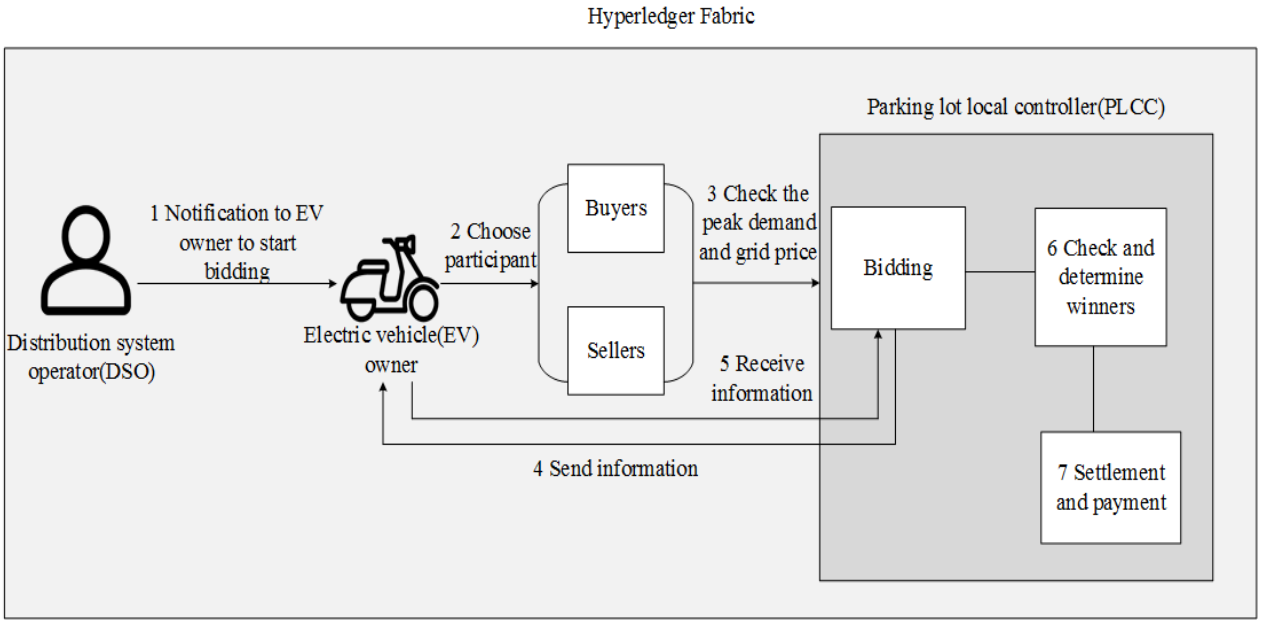

Figure 51. Reconstruction diagram of reference [85].

In Table 9, this paper presents a comprehensive comparison of smart energy schemes. The table compares their differences in six aspects: year, consensus algorithm, incentive mechanism, application domain, issue addressed, and performance evaluation. In the performance evaluation metrics, this paper presents the main experimental results of the schemes.

Table 9. Differences between schemes of smart energy (NS: mention but not specify, NM: not mentioned).

Authors	Year	Consensus Algorithm	Incentive Mechanism	Application Domain	Issued Addressed	Performance Evaluation
Pu Yuan et al.	2018	NM	×	Emission trading	Security	The saturation point for throughput is approximately 340 RPS (requests per second).
Zhou Hu et al.	2020	Delegated Proof of Reputation	√	Carbon emission trading	Trading scheme and consensus mechanism	-
Zheng Che et al.	2019	Solo	×	Distributed renewable energy transaction	Transaction authentication	250 TPS (transactions per second)
Felipe Condon Silva et al.	2019	NM	×	Energy trading for electric vehicles in smart campus	Intelligence, security	-

5. Future Trends and Prospects

With the adoption and research of Hyperledger, the convergence of the IoT and Hyperledger attracted a lot of attention. Hyperledger faces many landing obstacles in the practical deployment of the IoT systems page, such as data inflation, performance bottlenecks, data maintenance, and migration difficulties. In order to promote the application of landing and popularization, Hyperledger still needs to solve these problems. In addition, there is a lack of effective legal regulation. As an emerging technology, Hyperledger will bring resistance to landing if there is no clear legal red line. In order to better plan the landing scenarios, there is still a need for unified international norms and industry standards. Among the existing research advances, Hyperledger addresses the shortcomings of other blockchain technologies in terms of flexibility, robustness, and privacy, but there are still some issues with Hyperledger-based IoT systems that are not fully researched and addressed, and these limitations are mainly focused on performance and incentive mechanisms. To this end, this study proposes four future directions.

1. Low-power consensus mechanism

The consensus algorithm is a key factor in determining the performance of a Hyperledger-based IoT system. Most of the IoT devices cannot fully satisfy the computation and high energy required to handle large amounts of consensus. Therefore, low-energy consensus algorithms for most IoT devices are an important issue that needs to be addressed urgently.

2. Intelligent transaction validation

The "Endorse + Kafka + Commit" model in Hyperledger does not completely solve the performance problem of transaction validation, and the existing transaction validation performance is still unable to meet the needs of handling the information exchange among a large number of IoT devices. The long response time of some nodes involved in verification will affect the efficiency of transaction verification. Therefore, using some intelligent clustering algorithms to filter the nodes with high current activity to assume the verification role may solve this problem [27].

3. Mixed on-chain and off-chain storage

The blocks of Hyperledger are stored in nodes, however, these nodes are usually IoT devices. Some devices have very low storage capacity and cannot store multiple blockchains. Using some distributed databases (e.g., IPFS, etc.) to fuse with Hyperledger for on-chain and off-chain storage may ease the storage pressure on the devices.

4. Customized incentives

Hyperledger does not promote any cryptocurrency as a reward, but the distributed task undertaking still needs an incentive mechanism as the main driver. This study argues

that such incentives can be customizable, and developers can focus on consumer interest areas. For example, if a consumer is attracted to a certain game or a membership service of a website, then that consumer's task reward can be self-selected among these options. Such incentives would increase the efficiency of collaboration and motivate users to better engage in it.

6. Conclusions

IoT systems based on blockchain have significant shortcomings in terms of scalability, flexibility, robustness, and privacy. To address these issues, Hyperledger is considered as an ideal technology and attracted a lot of attention. This study summarized and concluded the research on Hyperledger in the IoT, and demonstrated the feasibility and effectiveness of Hyperledger application in the IoT. The aim was to show more intuitive differences and design ideas with a reconstruction diagram perspective, and to provide researchers with a quick guide to technology integration. Hyperledger is able to satisfy a variety of business scenarios, but the exploration in the IoT is still in the preliminary stage. In addition, the overview of the reconstruction diagrams approach initially shows unique advantages in visualizing business logic, technology convergence, and is easy to read and understand.

Author Contributions: Writing original draft preparation, Z.L.; writing—review and editing, K.W., Y.Z., X.Y. and T.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of Jilin Province under Grant No. 202101176JC from Prof. Yuefeng Zheng.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, M.A.; Salah, K. IoT security: Review, blockchain solutions, and open challenges. *Future Gener. Comput. Syst.* **2018**, *82*, 395–411. [CrossRef]
2. Cachin, C. Architecture of the hyperledger blockchain fabric. In Proceedings of the Workshop on Distributed Cryptocurrencies and Consensus Ledgers, Chicago, IL, USA, 25 July 2016; Volume 310, pp. 1–4.
3. Elrom, E. Hyperledger. In *The Blockchain Developer*; Apress: Berkeley, CA, USA, 2019; pp. 299–348.
4. Aggarwal, S.; Kumar, N. Hyperledger. In *Advances in Computers*; Elsevier: Amsterdam, The Netherlands, 2021; Volume 121, pp. 323–343.
5. Blummer, T.; Sean, M.; Cachin, C. *An Introduction to Hyperledger*; Hyperledger Organization: San Francisco, CA, USA, 2018. Available online: https://www.hyperledger.org/wp-content/uploads/2018/08/HL_Whitepaper_IntroductiontoHyperledger.pdf (accessed on 18 December 2021).
6. Leng, Z.; Tan, Z.; Wang, K. Application of Hyperledger in the Hospital Information Systems: A Survey. *IEEE Access* **2021**, *9*, 128965–128987. [CrossRef]
7. Hyperledger, S. Introduction. Available online: https://sawtooth.hyperledger.org/docs/core/releases/latest/introduction.html (accessed on 17 March 2019).
8. Ampel, B.; Patton, M.; Chen, H. Performance modeling of hyperledger sawtooth blockchain. In Proceedings of the 2019 IEEE International Conference on Intelligence and Security Informatics (ISI), Shenzhen, China, 1–3 July 2019; pp. 59–61.
9. Moriggl, P.; Asprion, P.M.; Schneider, B. Blockchain technologies towards data privacy—hyperledger sawtooth as unit of analysis. In *New Trends in Business Information Systems and Technology*; Springer: Cham, Switzerland, 2021; pp. 299–313.
10. Vlachou, V.; Kontzinos, C.; Markaki, O.; Kokkinakos, P.; Karakolis, V.; Psarras, J. Leveraging Hyperledger Iroha for the Issuance and Verification of Higher-Education Certificates. *Int. J. Educ. Pedagog. Sci.* **2020**, *14*, 755–763.
11. Iushkevich, N.; Lebedev, A.; Šketa, R.; Takemiya, M. D3ledger: The decentralized digital depository platform for asset management based on hyperledger iroha. In Proceedings of the OTS 2019 Advanced Information Technology and Services, Maribor, Slovenia, 18–19 June 2019; pp. 29–36. [CrossRef]
12. Available online: https://github.com/hyperledger/iroha/blob/main/README.md (accessed on 5 January 2022).
13. Dunphy, P. A Note on the Blockchain Trilemma for Decentralized Identity: Learning from Experiments with Hyperledger Indy. *arXiv* **2022**, arXiv:2204.05784.
14. Bhattacharya, M.P.; Zavarsky, P.; Butakov, S. Enhancing the security and privacy of self-sovereign identities on hyperledger indy blockchain. In Proceedings of the 2020 International Symposium on Networks, Computers and Communications (ISNCC), Montreal, QC, Canada, 16–18 June 2020; pp. 1–7.
15. Available online: https://www.edx.org/course/identity-in-hyperledger-aries-indy-and-ursa (accessed on 5 January 2022).

16. Abramson, W.; Hall, A.J.; Papadopoulos, P.; Pitropakis, N.; Buchanan, W.J. A distributed trust framework for privacy-preserving machine learning. In Proceedings of the International Conference on Trust and Privacy in Digital Business, Bratislava, Slovakia, 3 June 2020; pp. 205–220.
17. Dalla Palma, S.; Pareschi, R.; Zappone, F. What is your distributed (hyper) ledger? In Proceedings of the 2021 IEEE/ACM 4th International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), Madrid, Spain, 31 May 2021; pp. 27–33.
18. Available online: https://besu.hyperledger.org/en/stable/ (accessed on 6 January 2022).
19. Keramidas, G.; Voros, N.; Hübner, M. *Components and Services for IoT Platforms*; Springer International Publishing: Cham, Switzerland, 2016.
20. Olson, K.; Bowman, M.; Mitchell, J.; Amundson, S.; Middleton, D.; Montgomery, C. Sawtooth: An introduction. The Linux Foundation, Jan, 2018. Available online: https://www.hyperledger.org/wp-content/uploads/2018/01/Hyperledger_Sawtooth_WhitePaper.pdf (accessed on 6 January 2022).
21. Lee, I.; Lee, K. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Bus. Horizons.* **2015**, *58*, 431–440. [CrossRef]
22. Wang, H.; Zhang, J. Blockchain based data integrity verification for large-scale IoT data. *IEEE Access* **2019**, *7*, 164996–165006. [CrossRef]
23. Yang, H.; Yuan, J.; Yao, H.; Yao, Q.; Yu, A.; Zhang, J. Blockchain-based hierarchical trust networking for JointCloud. *IEEE Internet Things J.* **2019**, *7*, 1667–1677. [CrossRef]
24. Dib, O.; Huyart, C.; Toumi, K. A novel data exploitation framework based on blockchain. *Pervasive Mob. Comput.* **2020**, *61*, 101104. [CrossRef]
25. Cao, Y.; Jia, F.; Manogaran, G. Efficient traceability systems of steel products using blockchain-based industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2019**, *16*, 6004–6012. [CrossRef]
26. Seshadri, S.S.; Rodriguez, D.; Subedi, M.; Choo, K.-K.R.; Ahmed, S.; Chen, Q.; Lee, J. Iotcop: A blockchain-based monitoring framework for detection and isolation of malicious devices in internet-of-things systems. *IEEE Internet Things J.* **2020**, *8*, 3346–3359. [CrossRef]
27. Kim, J.H.; Lee, S.; Hong, S. Autonomous Operation Control of IoT Blockchain Networks. *Electronics* **2021**, *10*, 204. [CrossRef]
28. Košťál, K.; Helebrandt, P.; Belluš, M.; Ries, M.; Kotuliak, I. Management and monitoring of IoT devices using blockchain. *Sensors* **2019**, *19*, 856. [CrossRef]
29. Mbarek, B.; Jabeur, N.; Pitner, T.; Yasar, A.-U.-H. Mbs: Multilevel blockchain system for IoT. *Pers. Ubiquit. Comput.* **2021**, *25*, 247–254. [CrossRef]
30. Ribeiro, V.; Holanda, R.; Ramos, A.; Rodrigues, J.J.P.C. Enhancing key management in LoRaWAN with permissioned blockchain. *Sensors* **2020**, *20*, 3068. [CrossRef]
31. Hang, L.; Kim, D.H. Design and implementation of an integrated iot blockchain platform for sensing data integrity. *Sensors* **2019**, *19*, 2228. [CrossRef]
32. Liu, H.; Han, D.; Li, D. Fabric-IoT: A blockchain-based access control system in IoT. *IEEE Access* **2020**, *8*, 18207–18218. [CrossRef]
33. Zhang, Y.; Liu, W.; Xia, Z.; Wang, Z.; Liu, L.; Zhang, W.; Zhang, H.; Fang, B. Blockchain-Based DNS Root Zone Management Decentralization for Internet of Things. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 6620236. [CrossRef]
34. Chi, J.; Li, Y.; Huang, J.; Liu, J.; Jin, Y.; Chen, C.; Qiu, T. A secure and efficient data sharing scheme based on blockchain in industrial Internet of Things. *J. Netw. Comput. Appl.* **2020**, *167*, 102710. [CrossRef]
35. Yu, D.; Liu, G.; Guo, M.; Liu, X. An improved K-medoids algorithm based on step increasing and optimizing medoids. *Expert Syst. Appl.* **2018**, *92*, 464–473. [CrossRef]
36. Fauzi, M.A.; Utomo, D.C.; Setiawan, B.D.; Pramukantoro, E.S. Automatic essay scoring system using n-gram and cosine similarity for gamification based e-learning. In Proceedings of the International Conference on Advances in Image Processing, Bangkok, Thailand, 25–27 August 2017; pp. 151–155.
37. Siris, V.A.; Dimopoulos, D.; Fotiou, N.; Voulgaris, S.; Polyzos, G.C. Decentralized authorization in constrained IoT environments exploiting interledger mechanisms. *Comput. Commun.* **2020**, *152*, 243–251. [CrossRef]
38. Kakei, S.; Shiraishi, Y.; Mohri, M.; Nakamura, T.; Hashimoto, M.; Saito, S. Cross-certification towards distributed authentication infrastructure: A case of hyperledger fabric. *IEEE Access* **2020**, *8*, 135742–135757. [CrossRef]
39. Pajooh, H.; Rashid, M.; Alam, F.; Demidenko, S. Multi-layer blockchain-based security architecture for internet of things. *Sensors* **2021**, *21*, 772. [CrossRef]
40. Zhou, L.; Wang, L.; Ai, T.; Sun, Y. BeeKeeper 2.0: Confidential blockchain-enabled IoT system with fully homomorphic computation. *Sensors* **2018**, *18*, 3785. [CrossRef]
41. Hou, L.; Zheng, K.; Liu, Z.; Xu, X.; Wu, T. Design and prototype implementation of a blockchain-enabled LoRa system with edge computing. *IEEE Internet Things J.* **2020**, *8*, 2419–2430. [CrossRef]
42. Yang, X.; Zhang, S.; Liu, J.; Gao, Q.; Dong, S.; Zhou, C. Deep learning for smart fish farming: Applications, opportunities and challenges. *Rev. Aquac.* **2021**, *13*, 66–90. [CrossRef]
43. Feng, Y.; Niu, H.; Wang, F.; Ivey, S.J.; Wu, J.J.; Qi, H.; Almeida, R.A.; Eda, S.; Cao, Q. SocialCattle: IoT-based Mastitis Detection and Control through Social Cattle Behavior Sensing in Smart Farms. *IEEE Internet Things J.* **2021**, *9*, 10130–10138. [CrossRef]

44. Hang, L.; Ullah, I.; Kim, D.H. A secure fish farm platform based on blockchain for agriculture data integrity. *Comput. Electron. Agric.* **2020**, *170*, 105251. [CrossRef]
45. Lee, S.; Lee, J.; Hong, S.; Kim, J.-H. Lightweight end-to-end blockchain for IoT applications. *KSII Trans. Internet Inf. Syst.* **2020**, *14*, 3224–3242.
46. Kara, N.; Cagiltay, K. Smart toys for preschool children: A design and development research. *Electron. Commer. Res. Appl.* **2020**, *39*, 100909. [CrossRef]
47. Yang, J.; Lu, Z.; Wu, J. Smart-toy-edge-computing-oriented data exchange based on blockchain. *J. Syst. Archit.* **2018**, *87*, 36–48. [CrossRef]
48. Manzoor, A.; Samarin, M.; Mason, D.; Ylianttila, M. Scavenger Hunt: Utilization of Blockchain and IoT for a location-based Game. *IEEE Access* **2020**, *8*, 204863–204879. [CrossRef]
49. Pittaras, I.; Fotiou, N.; Siris, V.; Polyzos, G. Beacons and blockchains in the mobile gaming ecosystem: A feasibility analysis. *Sensors* **2021**, *21*, 862. [CrossRef]
50. Farrokhi, A.; Farahbakhsh, R.; Rezazadeh, J.; Minerva, R. Application of Internet of Things and artificial intelligence for smart fitness: A survey. *Comput. Netw.* **2021**, *189*, 107859. [CrossRef]
51. Jamil, F.; Iqbal, N.; Imran; Ahmad, S.; Kim, D. Peer-to-peer energy trading mechanism based on blockchain and machine learning for sustainable electrical power supply in smart grid. *IEEE Access* **2021**, *9*, 39193–39217. [CrossRef]
52. Khan, P.W.; Byun, Y.C.; Park, N. A data verification system for CCTV surveillance cameras using blockchain technology in smart cities. *Electronics* **2020**, *9*, 484. [CrossRef]
53. Dileep, G. A survey on smart grid technologies and applications. *Renew. Energy* **2020**, *146*, 2589–2625. [CrossRef]
54. Li, F.; Qiao, W.; Sun, H.; Wan, H.; Wang, J.; Xia, Y.; Xu, Z.; Zhang, P. Smart transmission grid: Vision and framework. *IEEE Trans. Smart Grid.* **2010**, *1*, 168–177. [CrossRef]
55. Zhao, W.; Lv, J.; Yao, X.; Zhao, J.; Jin, Z.; Qiang, Y.; Che, Z.; Wei, C. Consortium Blockchain-Based microgrid market transaction research. *Energies* **2019**, *12*, 3812. [CrossRef]
56. Li, Y.; Hu, B. An iterative two-layer optimization charging and discharging trading scheme for electric vehicle using consortium blockchain. *IEEE Trans. Smart Grid* **2019**, *11*, 2627–2637. [CrossRef]
57. Li, Y.; Hu, B. A consortium blockchain-enabled secure and privacy-preserving optimized charging and discharging trading scheme for electric vehicles. *IEEE Trans. Ind. Inform.* **2020**, *17*, 1968–1977. [CrossRef]
58. Yu, Y.; Guo, Y.; Min, W.; Zeng, F. Trusted transactions in micro-grid based on blockchain. *Energies* **2019**, *12*, 1952. [CrossRef]
59. Lohachab, A.; Garg, S.; Kang, B.H.; Amin, M.B. Performance evaluation of Hyperledger Fabric-enabled framework for pervasive peer-to-peer energy trading in smart Cyber–Physical Systems. *Future Gener. Comput. Syst.* **2021**, *118*, 392–416. [CrossRef]
60. Jamil, F.; Kahng, H.K.; Kim, S.; Kim, D.-H. Towards Secure Fitness Framework Based on IoT-Enabled Blockchain Network Integrated with Machine Learning Algorithms. *Sensors* **2021**, *21*, 1640. [CrossRef]
61. Sciume, G.; Palacios-Garcia, E.J.; Gallo, P.; Sanseverino, E.R.; Vasquez, J.C.; Guerrero, J.M. Demand response service certification and customer baseline evaluation using blockchain technology. *IEEE Access* **2020**, *8*, 139313–139331. [CrossRef]
62. Wang, L.; Jiao, S.; Xie, Y.; Mubaarak, S.; Zhang, D.; Liu, J.; Jiang, S.; Zhang, Y.; Li, M. A Permissioned Blockchain-Based Energy Management System for Renewable Energy Microgrids. *Sustainability* **2021**, *13*, 1317. [CrossRef]
63. Nallaperuma, D.; Nawaratne, R.; Bandaragoda, T.; Adikari, A.; Nguyen, S.; Kempitiya, T.; De Silva, D.; Alahakoon, D.; Pothuhera, D. Online incremental machine learning platform for big data-driven smart traffic management. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 4679–4690. [CrossRef]
64. Djahel, S.; Doolan, R.; Muntean, G.-M.; Murphy, J. A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches. *IEEE Commun. Surv. Tutor.* **2014**, *17*, 125–151. [CrossRef]
65. Feng, Q.; He, D.; Zeadally, S.; Liang, K. BPAS: Blockchain-Assisted Privacy-Preserving Authentication System for Vehicular Ad Hoc Networks. *IEEE Trans. Ind. Inform.* **2020**, *16*, 4146–4155. [CrossRef]
66. Li, W.; Guo, H.; Nejad, M.; Shen, C.-C. Privacy-preserving traffic management: A blockchain and zero-knowledge proof inspired approach. *IEEE Access* **2020**, *8*, 181733–181743. [CrossRef]
67. Luo, G.; Shi, M.; Zhao, C.; Shi, Z. Hash-Chain-Based Cross-Regional Safety Authentication for Space-Air-Ground Integrated VANETs. *Appl. Sci.* **2020**, *10*, 4206. [CrossRef]
68. Buzachis, A.; Celesti, A.; Galletta, A.; Fazio, M.; Fortino, G.; Villari, M. A multi-agent autonomous intersection management (MA-AIM) system for smart cities leveraging edge-of-things and Blockchain. *Inf. Sci.* **2020**, *522*, 148–163. [CrossRef]
69. Mbarek, B.; Jabeur, N.; Pitner, T.; Yasar, A.-U.-H. Empowering communications in vehicular networks with an intelligent blockchain-based solution. *Sustainability* **2020**, *12*, 7917. [CrossRef]
70. Xiao, Y.; Liu, Y.; Li, T. Edge computing and blockchain for quick fake news detection in IoV. *Sensors* **2020**, *20*, 4360. [CrossRef]
71. Chen, W.; Chen, Y.; Chen, X.; Zheng, Z. Toward secure data sharing for the IoV: A quality-driven incentive mechanism with on-chain and off-chain guarantees. *IEEE Internet Things J.* **2019**, *7*, 1625–1640. [CrossRef]
72. Gao, F.; Zhu, L.; Shen, M.; Sharif, K.; Wan, Z.; Ren, K. A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks. *IEEE Network* **2018**, *32*, 184–192. [CrossRef]
73. Chiu, W.Y.; Meng, W. EdgeTC-a PBFT blockchain-based ETC scheme for smart cities. *Peer Peer Netw. Appl.* **2021**, *14*, 2874–2886. [CrossRef]

74. Bartolomeu, P.C.; Vieira, E.; Ferreira, J. Pay as You Go: A Generic Crypto Tolling Architecture. *IEEE Access* **2020**, *8*, 196212–196222. [CrossRef]
75. Lee, Y.; Jeong, S.; Masood, A.; Park, L.; Dao, N.-N.; Cho, S. Trustful Resource Management for Service Allocation in Fog-Enabled Intelligent Transportation Systems. *IEEE Access* **2020**, *8*, 147313–147322. [CrossRef]
76. Lee, D.; Lee, S.H.; Masoud, N.; Krishnan, M.S.; Li, V.C. Integrated digital twin and blockchain framework to support accountable information sharing in construction projects. *Autom. Constr.* **2021**, *127*, 103688. [CrossRef]
77. Haaskjold, H.; Andersen, B.; Langlo, J.A. Dissecting the project anatomy: Understanding the cost of managing construction projects. *Prod. Plan. Control.* **2021**, 1–22. [CrossRef]
78. Suliyanti, W.N.; Sari, R.F. Blockchain-Based Implementation of Building Information Modeling Information Using Hyperledger Composer. *Sustainability* **2021**, *13*, 321. [CrossRef]
79. Elghaish, F.; Abrishami, S.; Hosseini, M.R. Integrated project delivery with blockchain: An automated financial system. *Autom. Constr.* **2020**, *114*, 103182. [CrossRef]
80. Yang, R.; Wakefield, R.; Lyu, S.; Jayasuriya, S.; Han, Y.; Yi, X.; Yang, X.; Amarasinghe, G.; Chen, S. Public and private blockchain in construction business process and information integration. *Autom. Constr.* **2020**, *118*, 103276. [CrossRef]
81. Sheng, D.; Ding, L.; Zhong, B.; Love, P.E.; Luo, H.; Chen, J. Construction quality information management with blockchains. *Autom. Constr.* **2020**, *120*, 103373. [CrossRef]
82. Yuan, P.; Xiong, X.; Lei, L.; Zheng, K. Design and implementation on hyperledger-based emission trading system. *IEEE Access* **2018**, *7*, 6109–6116. [CrossRef]
83. Hu, Z.; Du, Y.; Rao, C.; Goh, M. Delegated Proof of Reputation Consensus Mechanism for Blockchain-Enabled Distributed Carbon Emission Trading System. *IEEE Access* **2020**, *8*, 214932–214944. [CrossRef]
84. Che, Z.; Wang, Y.; Zhao, J.; Qiang, Y.; Ma, Y.; Liu, J. A distributed energy trading authentication mechanism based on a consortium blockchain. *Energies* **2019**, *12*, 2878. [CrossRef]
85. Silva, F.C.; Ahmed, M.A.; Martínez, J.M.; Kim, Y.-C. Design and implementation of a blockchain-based energy trading platform for electric vehicles in smart campus parking lots. *Energies* **2019**, *12*, 4814. [CrossRef]