electronics

MDPI

Review

An Overview of Technologies for Improving Storage Efficiency in Blockchain-Based IIoT Applications

Nana Kwadwo Akrasi-Mensah [1,*], Eric Tutu Tchao [1,*], Axel Sikora [2], Andrew Selasi Agbemenu [1], Henry Nunoo-Mensah [1], Abdul-Rahman Ahmed [3], Dominik Welte [2] and Eliel Keelson [1]

[1] Department of Computer Engineering, Kwame Nkrumah University of Science and Technology, PMB, Kumasi AK-448-3929, Ghana

[2] Institute of Reliable Embedded Systems and Communication Electronics (ivESK), Offenburg University of Applied Sciences, 77652 Offenburg, Germany

[3] Department of Telecommunications Engineering, Kwame Nkrumah University of Science and Technology, PMB, Kumasi AK-448-3929, Ghana

* Correspondence: nakrasi97@gmail.com (N.K.A.-M.); ettchao.coe@knust.edu.gh (E.T.T.)

Abstract: Since the inception of blockchain-based cryptocurrencies, researchers have been fascinated with the idea of integrating blockchain technology into other fields, such as health and manufacturing. Despite the benefits of blockchain, which include immutability, transparency, and traceability, certain issues that limit its integration with IIoT still linger. One of these prominent problems is the storage inefficiency of the blockchain. Due to the append-only nature of the blockchain, the growth of the blockchain ledger inevitably leads to high storage requirements for blockchain peers. This poses a challenge for its integration with the IIoT, where high volumes of data are generated at a relatively faster rate than in applications such as financial systems. Therefore, there is a need for blockchain architectures that deal effectively with the rapid growth of the blockchain ledger. This paper discusses the problem of storage inefficiency in existing blockchain systems, how this affects their scalability, and the challenges that this poses to their integration with IIoT. This paper explores existing solutions for improving the storage efficiency of blockchain–IIoT systems, classifying these proposed solutions according to their approaches and providing insight into their effectiveness through a detailed comparative analysis and examination of their long-term sustainability. Potential directions for future research on the enhancement of storage efficiency in blockchain–IIoT systems are also discussed.

Keywords: blockchain; IIoT; scalability; storage efficiency; storage optimization; compression; summarization; machine learning

check for updates

Citation: Akrasi-Mensah, N.K.; Tchao, E.T.; Sikora, A.; Agbemenu, A.S.; Nunoo-Mensah, H.; Ahmed, A.-R.; Welte, D.; Keelson, E. An Overview of Technologies for Improving Storage Efficiency in Blockchain-Based IIoT Applications. *Electronics* **2022**, *11*, 2513. https://doi.org/10.3390/electronics11162513

Academic Editor: Jemal Abawajy

Received: 24 May 2022
Accepted: 21 July 2022
Published: 11 August 2022

1. Introduction

Blockchain technology holds great promise for various applications, such as Internet of Things (IoT) and edge computing [1–4]. Since it was brought into the public mainstream through the inception of cryptocurrencies such as Bitcoin [5] and Ethereum [6], blockchain technology has become a major research direction [7]. Over the years, it has garnered many applications and proposed works in fields such as health [8,9], agriculture [10], and manufacturing [11], and it continues to inspire works.

IoT also has many applications in many areas, such as smart homes for consumers [12,13], agriculture [14–16], health [17], energy management [18,19], manufacturing [20,21], and supply chain management [22,23]. The adoption of IoT in the collection, monitoring, and analysis of data and the control of industrial operations is referred to as Industrial Internet of Things (IIoT) [24]. At the heart of IoT is the use of centralized storage systems, such as with cloud computing platforms, to aggregate and analyze large amounts of data to gain insight and make well-informed decisions. This can, however, lead to privacy and security concerns, especially with data sharing among many industrial participants, such as in supply chains.

On their own, IoT and blockchain have contributed to massive advancements in the fields to which they have been applied [25,26]. The benefits of the blockchain, which include enhanced security, transparency, and greater traceability [27], make it a promising technology for integration with IIoT, which has long had issues with security [28–32]. However, there are several issues that limit the integration of blockchain into IIoT systems [25,33]. One of these issues is the huge storage requirement of the blockchain. The immutable nature of the blockchain ledger means that it is append-only; information stored on the ledger cannot be deleted. Thus, the volume of data stored on the blockchain increases over time as new blocks are added.

With the Hyperledger Fabric platform, which is a permissioned enterprise-grade blockchain platform, in medium- to high-level transaction environments, storage for the blockchain alone could be as high as 197 Tb or more per year, according to IBM [34]. IIoT systems generate data much in a greater volume than financial transaction data and at a faster rate [35,36]; thus, the storage volume required to store the ledger recording the data would also increase over time [34]. The huge storage requirement restricts the participation of resource-constrained devices, such as sensors and other low-power IoT devices [37]. The growing data volume also reduces the storage efficiency of the underlying database and causes delays in reading and writing to the database [38]. This affects the scalability and performance of the blockchain system [38,39]. Eventually, the decentralization of the ledger will also be affected, since fewer and fewer nodes will be able to join the network due to the high storage requirements.

Therefore, there is a need for blockchain architectures or storage models that alleviate the drawbacks of the blockchain's append-only approach. Various works have been proposed to improve the storage efficiency of blockchain systems. This paper attempts to explore the viability of these works in relation to IIoT by asking relevant questions about how these works approach the storage inefficiency problem, how effective they are at handling the storage of the ledger, and how they deal with the exponential growth of the ledger.

There has been a rapid accumulation of data in many areas, such as scientific research, manufacturing, and production in the era of big data; the data need to be made full use of [40]. This has contributed to the rapid progress and remarkable results of machine learning in regards to theory, methods, and applications. Although there are works that have proposed the use of machine learning in blockchain optimization, there is a distinct lack of focus on the optimization of the storage schemes used in these systems. This paper also explores the potential of machine learning for augmenting storage optimization schemes on the blockchain.

The main contributions of this survey include:

- Investigating the need for storage optimization schemes in blockchain–IIoT systems.
- Establishing a taxonomy for the various approaches to storage optimization in blockchain systems.
- Examining the contributions and shortcomings of various works proposed under each approach.
- Establishing the potential of machine learning to augment storage optimization in blockchain systems.
- Presenting open issues and directions for future work.

The rest of this paper is organized as follows: Section 2 presents the review methodology, Section 3 discusses the storage and scalability concerns of blockchain–IIoT integration, Section 4 presents a review of the approaches to ensuring storage efficiency in the blockchain, Section 5 discusses some works for blockchain optimization using machine learning techniques, Section 6 discusses the challenges faced in ensuring storage efficiency in blockchain–IIoT systems and explores the possibility of future works, and Section 7 concludes the paper.

2. Methodology

A number of proposed works for the storage optimization of blockchain-based IIoT applications and the performance optimization of blockchain systems using machine learning are reviewed in this paper.

2.1. Data Sources

This review used literature from four electronic databases, which were:

- Google Scholar;
- Elsevier Scopus;
- IEEE Xplore;
- Springer SpringerLink.

The search in all the databases returned a total of 113,888 results from January 2016 to May 2022. However, Google Scholar and SpringerLink returned many unrelated results, which led to only the first 100 results of each search on these databases being considered for the work. Thus, the total number of relevant results considered was 1747 results.

The keywords outlined in this paper were used to search through these databases. The search strings used included:

- "blockchain" AND "IIoT" AND "scalability"—33,885 results;
- "blockchain" AND "storage" AND "optimization"—6400 results;
- "blockchain" AND "storage" AND "compression"—6655 results;
- "blockchain" AND "summarization"—20,036 results;
- "blockchain" AND "optimization" AND "machine" AND "learning"—46,912 results.

The search was conducted from September 2020 to May 2022. The search focused on works concerned with optimizing storage for blockchain-based IIoT, as well as those for generic blockchain systems. The works covered in this review are not exhaustive, but provide a good overview of attempts to improve the storage efficiency of blockchain systems for IIoT applications. A number of works regarding the use of machine learning in blockchain systems are also explored to encourage blockchain storage optimization using machine learning techniques.

2.2. Selection Criteria for Reviewed Works

The selection of works for review in this paper was achieved with the aid of a temporal criterion to ensure that the state of the art in storage optimization for blockchain and machine learning optimization of the blockchain were examined. To further narrow down these works and improve the quality of the review, it was determined that works that showed relevance to the subject of the review by focusing on the storage concerns of the blockchain and the use of machine learning to improve blockchain performance would be considered. Figure 1 shows the selection process undertaken for this review, and Table 1 describes the criteria used to select the works for review.

Table 1. Inclusion criteria for selected works.

No.	Criterion
1	Publication date between 2018 and 2022
2	Emphasis on solutions for the storage and scalability of blockchain systems
3	Emphasis on blockchain performance optimization using machine learning techniques
4	Detailed descriptions of the proposed schemes and the underlying technologies

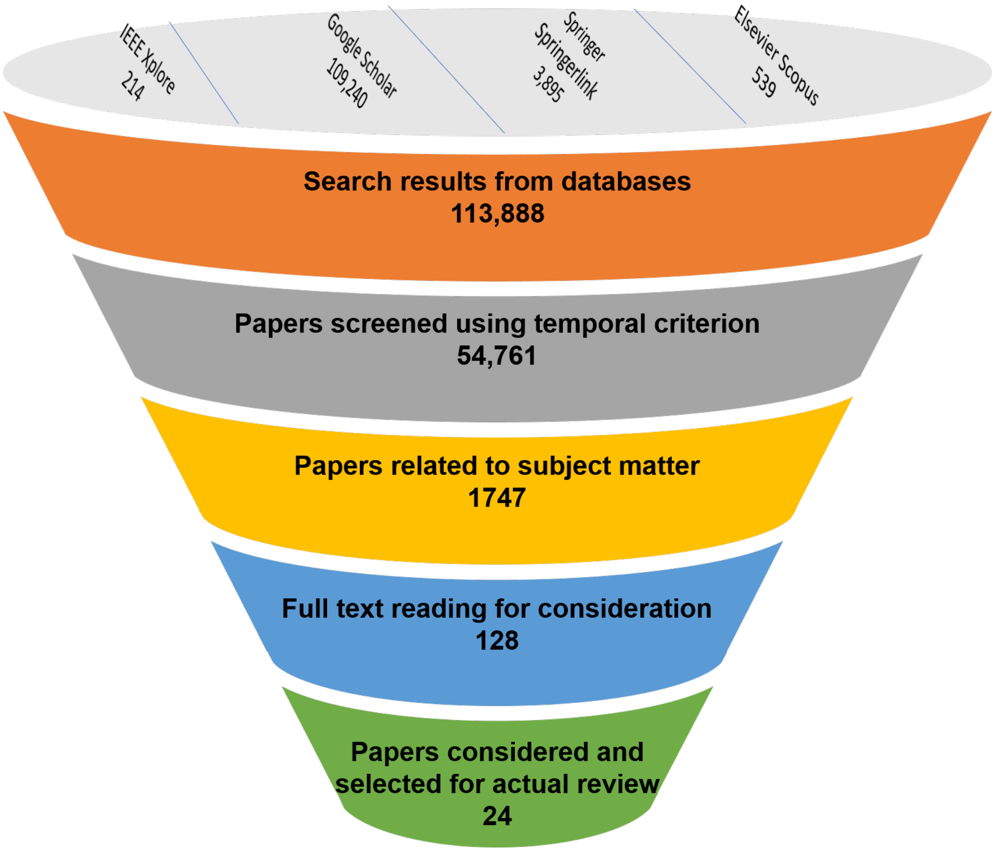
IEEE Xplore
214
Google Scholar
109,240
Springer
Springerlink
3,895
Elsevier Scopus
539
Search results from databases
113,888
Papers screened using temporal criterion
54,761
Papers related to subject matter
1747
Full text reading for consideration
128
Papers considered and
selected for actual review
24

Figure 1. Selection process.

The literature considered in the search results was selected based on a four-year temporal criterion, from 2018 to 2022. The studies considered for 2022 were those available at the time of conducting this research. The reason for choosing this criterion was to ensure that the most recent and relevant studies were provided to enable great future research in the field.

The selected works for this review were compiled after reviewing the complete text of those works that fell within the temporal range specified. Some of the literature appeared as duplicates in the different databases. Studies that were not closely related to the research topic were also excluded from the review. The final list of works that were explored in the review section of this paper were those that provided a clear and detailed description of their approach and the underlying technologies. The works considered provide novel solutions and architectures for the optimization of blockchain storage. Those works that did not have enough implementation or design details were excluded from the review. The works on the use of machine learning for optimization of blockchain performance that were considered were those that focused directly on improving blockchain parameters such as transaction throughput and transaction latency.

3. Storage and Scalability Concerns of Blockchain–IIoT Integration

The immutable nature of the blockchain and its reliance on consensus between participating nodes give rise to several issues around the storage of the blockchain ledger. The number of blocks that can be appended to the blockchain in a given period of time is limited due to the consensus mechanism and data broadcast between nodes [41]; thus, the throughput of transactions is much lower compared to more traditional database-based systems [42–44].

IIoT connects many devices, all of which generate data and require management, storage, and retrieval; the throughput of typical blockchain systems would be inadequate to deal with all of these connected devices. Full nodes on a blockchain network are required to store the entire blockchain ledger. Since the ledger is append-only, the capacity of these nodes to store the ledger will eventually be exceeded, and their storage capacity would have to be expanded to adapt [45–49].

The growth of the blockchain ledger greatly affects the scalability of the blockchain system. The number of full nodes on the blockchain is also restricted due to the high storage requirements [50]. This increases centralization in the blockchain, which, in turn, affects the security of the system. These three blockchain characteristics—decentralization, scalability, and security—are considered crucial and are at the heart of the blockchain trilemma, a concept first described by Vitalik Buterin, the co-founder of Ethereum, as shown in Figure 2 [44].

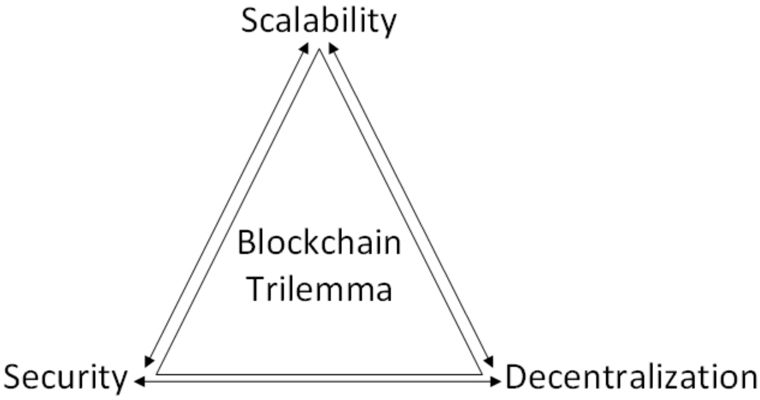

Figure 2. The blockchain trilemma.

The blockchain trilemma proposes that tradeoffs among the decentralization, scalability, and security of a blockchain system are inevitable [44,51]. The blockchain is, by nature, decentralized, and security is an essential property in its operation. However, this affects its scalability. A classic example is in the Bitcoin network, where reducing latency to improve transaction throughput may result in weakened security due to a higher probability of creating forks in the blockchain [44].

Some works, such as those of Xu et al. [52] and Nartey et al. [53], propose off-chain solutions for reducing the storage demand on peers. The cloud service, as the off-chain storage, improves the scalability of the blockchain system and allows resource-constrained devices to join the network as peers. However, the cloud service introduces centralization to the system and may compromise the security as well.

Many proposed works struggle to find a balance among the three essential characteristics in pursuing solutions to the storage and scalability problems of the blockchain. The integration of blockchain into IIoT can only be accelerated through the proposal of more sophisticated solutions for the storage inefficiency and scalability issues of the traditional blockchain system. Many works have been proposed in this regard, and they are explored in the next section.

4. Review of Approaches to Storage Efficiency in Blockchain–IIoT

The storage problem of the blockchain has been approached in different ways by works that propose solutions for mitigating it. These storage optimization schemes or storage models are usually motivated by specific use cases and may be designed for either permissionless or permissioned blockchains. While the same principles underlie both blockchain architectures, their designs differ in many ways. Some storage optimization schemes capitalize on certain aspects of these architectures to achieve storage efficiency. The requirements of the use case influence the blockchain architecture and, particularly in IIoT, permissioned blockchains are used, since industrial participants are known and access to data can be controlled. Some of the schemes discussed in this section can be implemented on either permissioned or permissionless blockchains. Schemes of this nature generally do not change the operation of the underlying blockchain and may involve processing of data before submission to the blockchain or changing the storage system of the peers.

This section explores the various works proposed to deal with the problem of storage overhead in blockchain systems. The taxonomy proposed in this review is based on the method by which storage optimization is achieved. As shown in Figure 3, these works can be categorized into:

- Compression-based schemes;
- Summarization-based schemes;
- Storage scheme optimization;
- Machine-learning-enabled schemes.

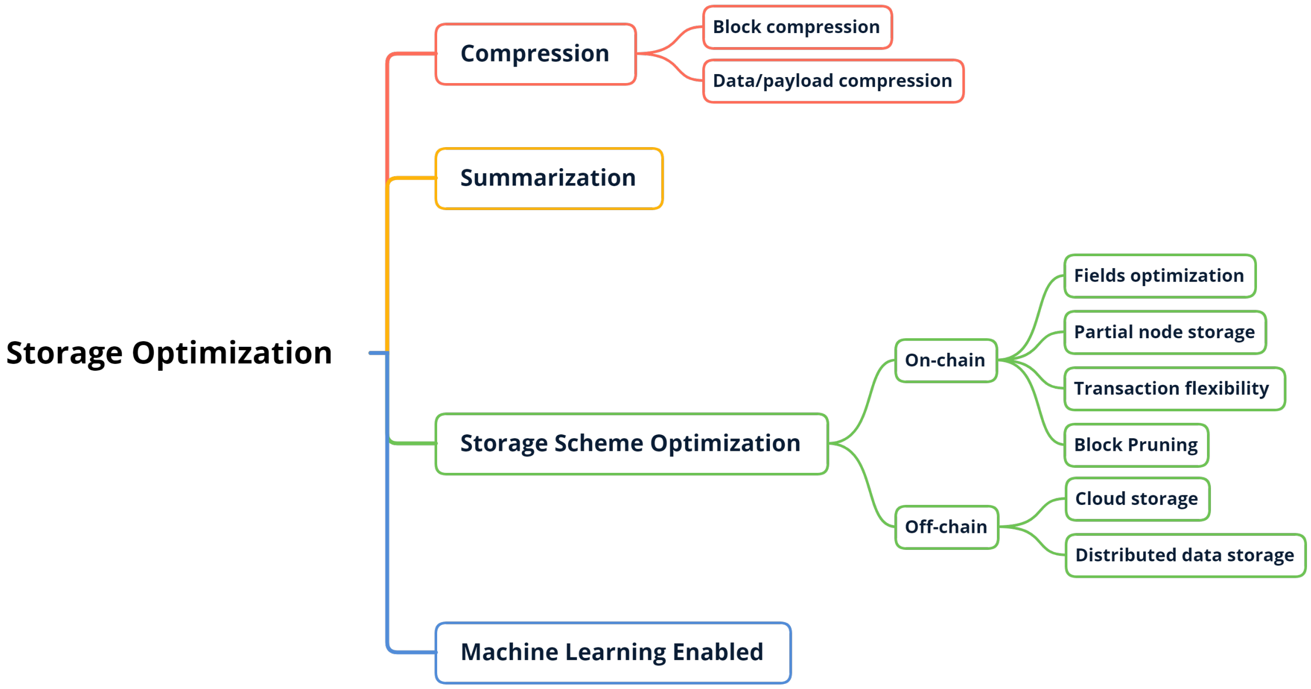

Figure 3. Taxonomy of storage optimization schemes for blockchain systems.

4.1. Compression-Based Schemes

Compression-based schemes utilize a compression algorithm to reduce the amounts of data that are submitted as transactions to the blockchain or to reduce the size of the blocks in the blockchain. They can be divided into block compression techniques and data compression techniques. Table 2 shows a comparison of these schemes.

Table 2. Comparison of compression-based schemes.

Proposed Work	Approach	Algorithm	Compression Ratio/ Storage Reduction	Limitations
Qi et al. [28]	Data Compression	Tree-based key-value compression	4–9×	May have a low compression ratio for large product record data
Kim et al. [54]	Block Compression	Block Merkle Tree	76.02% reduction	Sidechain requires synchronization between nodes

Table 2. *Cont.*

Proposed Work	Approach	Algorithm	Compression Ratio/ Storage Reduction	Limitations
Spataru et al. [55]	Block Compression	Huffman coding and LZW compression	48.5% reduction	Only suited for Ethereum and Ethereum-like blockchains, only focused on smart contract code size
Chen et al. [56]	Block Compression	Replacement of hash pointers with index pointers	12.71% reduction	Low storage overhead reduction, not suited for large-scale systems such as IIoT
Marsalek et al. [57]	Block Compression	Snapshot block	93% reduction	Accumulation of compression results over time, suitable for UTXO-based blockchains
Yu et al. [58]	Block Compression	Deflate algorithm	30.53%–42.16% of original block	Increased mining difficulty
Ding et al. [59]	Block Compression	Txilm Protocol	8	Increased latency

4.1.1. Block Compression

Block compression schemes aim at reducing the storage overhead of the blockchain by compressing the block after it is generated and committed to the blockchain. Kim et al. [54] proposed SELCOM, a selective compression scheme using a Block Merkle Tree, for lightweight nodes in blockchain systems. As shown in Figure 4, SELCOM allows nodes to maintain blocks selectively through a second chain called a checkpoint chain. It uses BMT to compress several blocks into a checkpoint. The compressed blocks can then be selectively removed or maintained depending on each node. Their results indicated an average storage reduction of 76.02%. The maintenance of a second chain introduces more complexity, as synchronization between peers for this chain is required. Unlike other works, the authors proposed an update mechanism to reduce the accumulation of compression results over time. While SELCOM can be used to verify numerous blocks with fewer compression results, the security of such an approach was not explored. Since IIoT systems have long been plagued with security concerns, the ability of lightweight nodes to selectively maintain blocks raises concerns, since it may be also be easier to have malicious nodes on the network. To improve the security of such sidechains, research should be undertaken to explore the use of further cryptographic proofs [60].

Spataru et al. [55] proposed a blockchain architecture with adaptive smart contract compression to reduce the storage overhead of smart contracts. They used a hybrid compression algorithm that combines the Huffman coding and LZW compression schemes. They performed their testing using a custom blockchain called ABEY (Advanced Blockchain for Enhanced Yields) with an enhanced virtual machine similar to that of Ethereum's VM. Their results indicated a 48.5% reduction of smart contract code size on average. This approach is suited for the Ethereum blockchain. Its design is based on the Ethereum VM and is practically suited for the Ethereum network's large number of smart contracts or distributed applications. For large-scale IIoT systems, the proposed work may not cause a significant reduction in the storage overhead of the blockchain ledger, since it is only focused on the smart contract code size. This approach could be improved by extending the compression techniques proposed to not only bytecode, but also transaction data stored in other blocks. Further research could investigate the effect of compressing other blocks on the network as well.

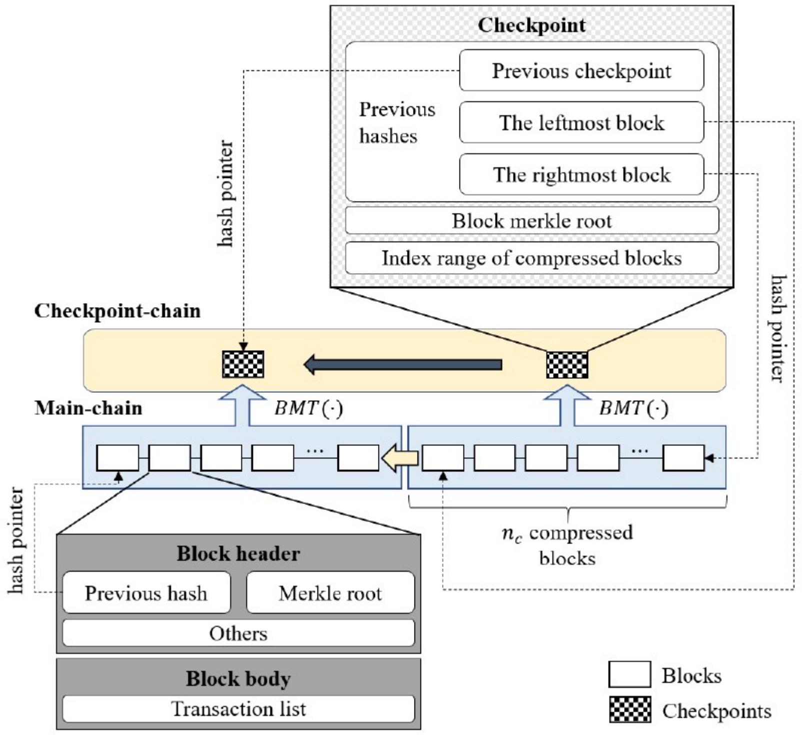

Figure 4. SELCOM [54].

Chen et al. [56] proposed a blockchain compression algorithm for blank node synchronization on the bitcoin blockchain. A new blank node that joins the bitcoin blockchain must download the full ledger from a full node. To reduce the bandwidth required to send these data, they proposed this compression algorithm. Their approach involves replacing hash pointers in transactions with index pointers. They deduced that hash pointers take up 15.07% of the space in blocks. Their results indicated a storage reduction by up to 12.71%. Since it was proposed based on the Bitcoin blockchain, it may be difficult to implement on other blockchain systems. The amount of storage space saved is also quite low compared to that in other works. Further improvement to this work could involve further reduction of redundancies in the transaction and block structure.

Marsalek et al. [57] proposed a compressible blockchain architecture for reducing the size of the blockchain. They proposed a snapshot-based approach where snapshot blocks containing the complete Unspent Transaction Output (UTXO) and all the block headers are created periodically. These blocks form a snapshot chain, which is a second chain that can be stored on lightweight or resource-constrained devices. They implemented their prototype using Java. Their results indicated that this approach could reduce the size of the blockchain by up to 93%. This approach is suited for UTXO-based blockchains, such as Bitcoin. It has a relatively high complexity due to the maintenance of a second chain, which requires synchronization of the nodes. It also does not account for the accumulation of snapshots over time. The introduction of update mechanisms to reduce the accumulation of snapshots over time would be a good improvement.

Ding et al. [59] proposed the Txilm protocol, which compresses the transactions in the blockchain using the short hash of their transaction ID (TXID). The proposed method reduces the size of the transaction from 32 bytes to 40 bits. The authors further optimized Txilm by sorting transactions based on their TXIDs, which results in the reduction of the transaction size to 32 bits and a compression ratio of 8. This approach, however, increases the probability of hash collisions. To prevent collision attacks, the authors introduced a salt while calculating the short hash, which makes it harder for malicious attackers to create malicious transactions.

Yu et al. [58] proposed a consensus algorithm called the Proof-of-Work-and-Block-Compression (POW-BC), which was meant to improve the Proof-of-Work consensus algorithm through transmission efficiency and the reduction of disk space occupied by blocks. This was done by compressing blocks and adjusting consensus parameters. The block compression is achieved using the deflate algorithm.

4.1.2. Data Compression

Some works have proposed the compression of product data before they are encapsulated in blockchain transactions. Qi et al. [28] proposed Cpds, a framework for efficient and private data sharing for product traceability using Industrial Internet of Things (IIoT) over the blockchain. As shown in Figure 5, the authors employ an off-chain procedure that compresses and encrypts product data before its eventual submission to the blockchain. Cpds uses a tree-based data compression mechanism that leverages the tree structure of traditional industrial systems for the amortization of data compression overhead. Participants along the path in an industrial process submit point transactions with the latest off-chain storage address of product data to the blockchain when they transfer product records to the next participant. Terminal participants compress the final product data and submit them to the blockchain as a data transaction. The authors implemented their prototype of Cpds using Java and Python, and they used Hyperledger Fabric as the blockchain. Their results showed that Cpds reduces storage overhead by 4–9 times compared to the baseline design and has between 4.8 and 20 times faster access time than that of the baseline design. Cpds is designed for permissioned blockchains for IIoT. It has a low impact on the blockchain's core operations, since it is only an overlay framework that sits atop the blockchain platform. In terms of complexity, this approach is still relatively high, since it involves building a unified data-sharing service encompassing compression and encryption techniques that handle product record transfer between industrial participants, compression of product data, data access control, and authentication. Further research could be undertaken to determine how well Cpds performs with large product data, since their tests were performed on small product data from 100 bytes to 10 Kb.

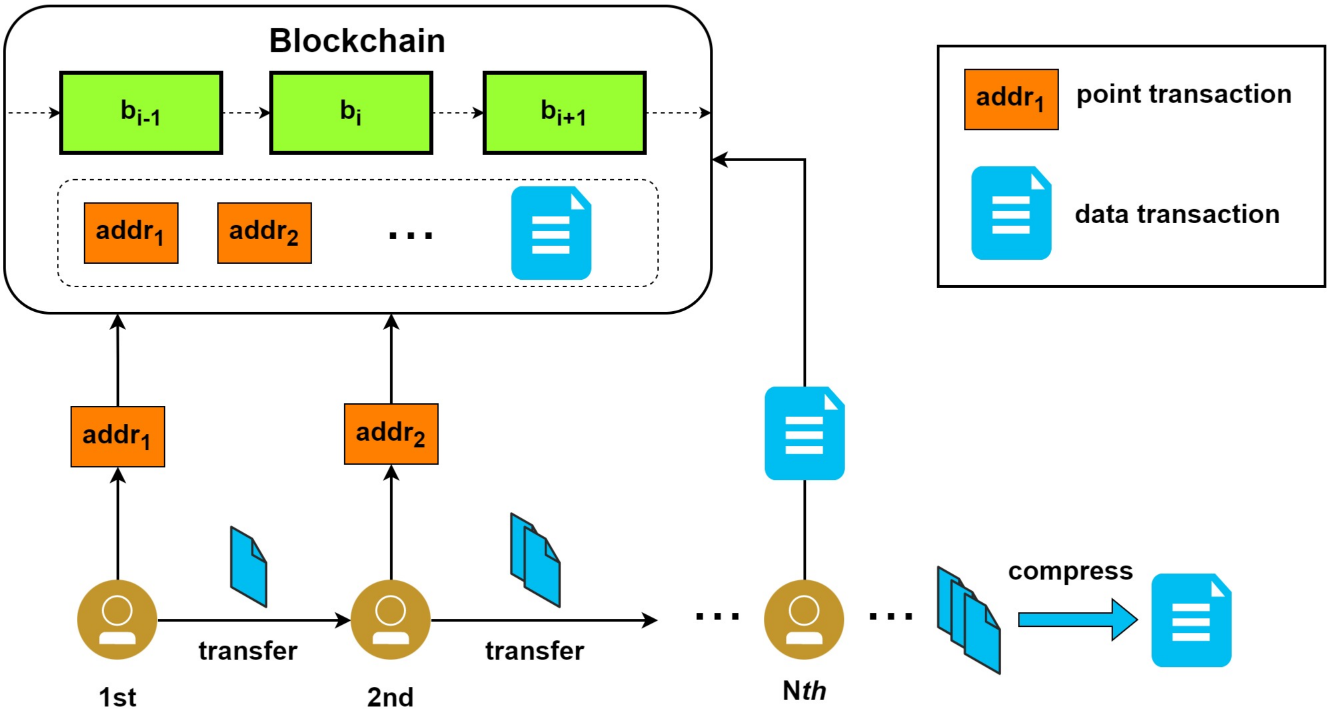

Figure 5. Compressed and private data sharing (Cpds) [28].

4.2. Summarization-Based Schemes

Works based on summarization propose the use of summary blocks to reduce storage overhead. These summary blocks contain details from original blocks that can then be replaced by the summary blocks. A comparison of these works can be found in Table 3.

Table 3. Comparison of summarization-based schemes.

Proposed Work	Approach	Algorithm	Storage Reduction	Limitations
Palai et al. [61]	Summarization	Recursive summarization tree	54%	Huge block summary size
Nadiya et al. [62]	Summarization	Recursive summarization tree and deflate compression algorithm	78.1%	Designed for bitcoin blockchain, lack of standard summary block for other blockchains

Palai et al. [61] proposed a block summarization method for reducing blockchain storage overhead on systems with transferable transactions. This method allows resource-constrained nodes to store the blockchain in a form that allows them to validate transactions independently. This approach targets transactions that involve the partial or full transfer of entities with a calculable net change for a series of transactions. The proposed method replaces actual blocks with corresponding summary blocks with sufficient details from the original blocks. This is achieved through a recursive summarization tree. The results of their experiments showed a compression ratio of 54%. Their work failed to deal with the problem of huge summary block sizes, which was addressed by the authors in [62].

Nadiya et al. [62] proposed a method for saving space on the blockchain that combines a block summarization algorithm and the deflate compression algorithm as an improvement of the work proposed in [61]. In this method, summary blocks, after they are formed, are compressed with the deflate compression algorithm. Their results showed space savings of 22.318% for summary blocks and 78.104% for compressed summary blocks. Whilst an improvement on the storage savings of the recursive summarization strategy was achieved, the target system was the Bitcoin blockchain, and thus, the lack of a standard summary block limits its application to other blockchain systems. The proposal of a standard summary block for different blockchain systems could be of interest for further research.

4.3. Storage Scheme Optimization

Another approach to improving the storage efficiency of blockchain systems is to improve or change the storage schemes of these systems. Generally, there are two ways in which blockchain data are stored; these are on-chain, where all blockchain data are either fully or partially stored by the blockchain peers, and off-chain, which introduces technologies such as cloud computing and secure distributed file storage to alleviate the storage burden on the blockchain peers.

4.3.1. Off-Chain Storage

An intuitive approach to reducing the storage burden on blockchain peers is to leverage the storage capabilities of other systems outside the blockchain network. There are two main ways in which this can be achieved: cloud storage and distributed file storage. Table 4 shows a comparison of these works.

Table 4. Comparison of off-chain storage scheme optimization works.

Proposed Work	Approach	Algorithm	Storage Reduction	Runtime	Limitations
Xu et al. [52]	Cloud storage	NSGA-C	30%	872.4 s	Long runtime
Nartey et al. [53]	Cloud storage	AT-MOPSO	-	384.2 s	Relatively poor solution for local space occupancy compared to NSGA-C
Zheng et al. [63]	Distributed data storage	IPFS-based storage	91.83%	-	Increased latency due to queries to IPFS network

Xu et al. [52] proposed the selection and storage of old blocks that are less likely to be queried in the cloud to expand blockchain capacity for each peer. In their architecture, shown in Figure 6, peers on the network are connected to cloud servers, to which selected blocks are offloaded. They proposed the block selection problem as a multi-objective optimization problem using objective functions of query probability, storage cost, and local space occupancy. To solve the problem, the authors proposed a nondominated sorting genetic algorithm with clustering (NSGA-C). Their results indicated an average storage overhead reduction of 30%. However, the results showed a relatively longer runtime than the benchmark algorithms.

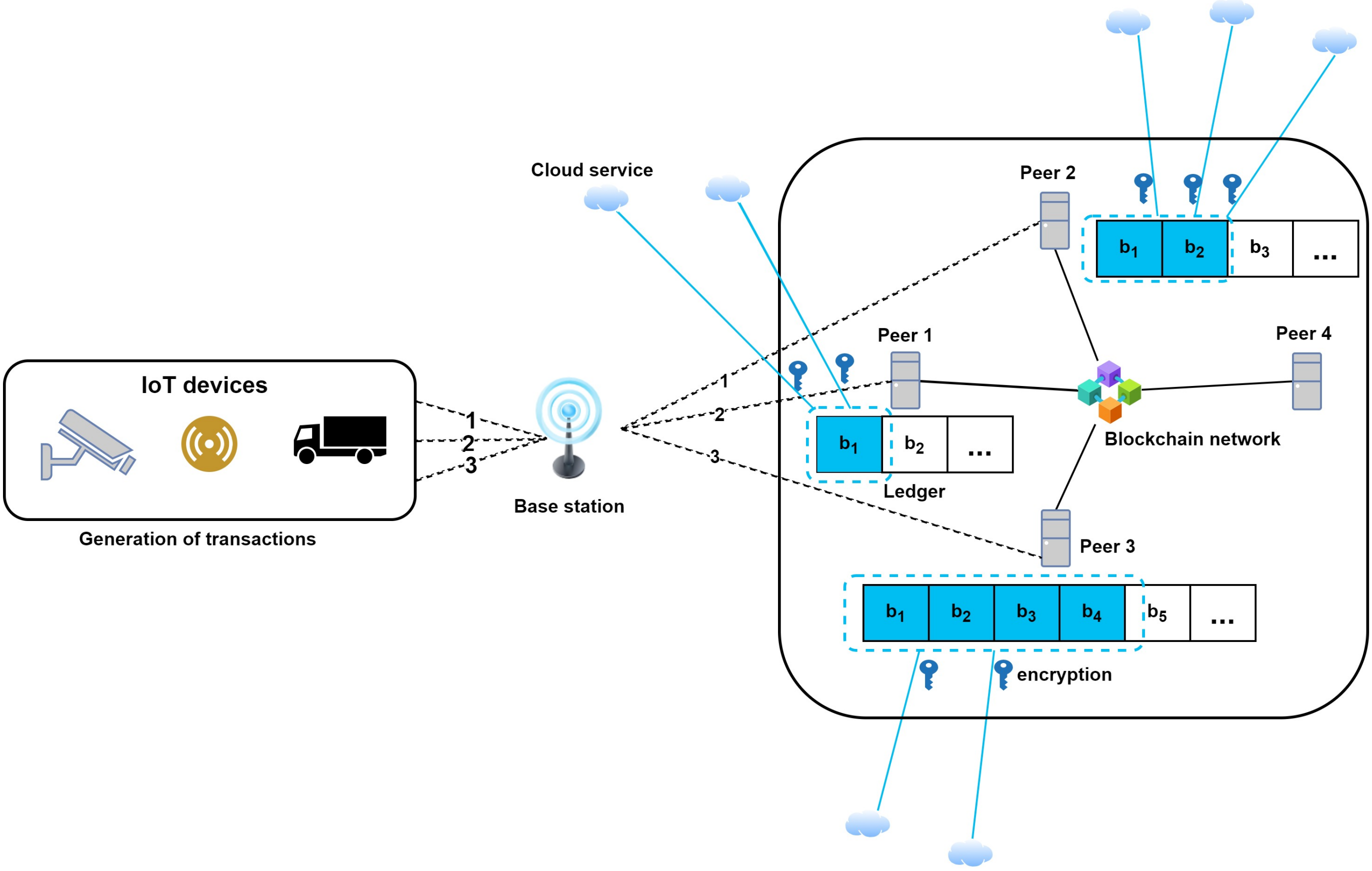

Figure 6. Storage optimization using cloud storage [52].

Nartey et al. [53] improved the work in [52] with the proposal of an advanced time-variant multi-objective particle swarm optimization algorithm to solve the block selection problem. They proposed a blockchain-IIoT framework with fog nodes running side-chains using containers for the block selection, as shown in Figure 7. Their results showed a relatively shorter runtime than the NSGA-C and other benchmark algorithms and lower power consumption compared to the NSGA-C. Despite showing a marked improvement in runtime and energy efficiency, the AT-MOPSO performed relatively worse than NSGA-C for the local space occupancy objective, and this may be reflected in a lower storage overhead reduction for peers. The metaheuristic algorithms proposed in these works to solve the block selection problem could be improved through machine learning and, more specifically, deep reinforcement learning (DRL). The strong generalization ability and faster runtime of a trained DRL model [64] compared to the evolutionary algorithms that are widely used in solving multi-objective optimization problems could prove pivotal for obtaining better solutions.

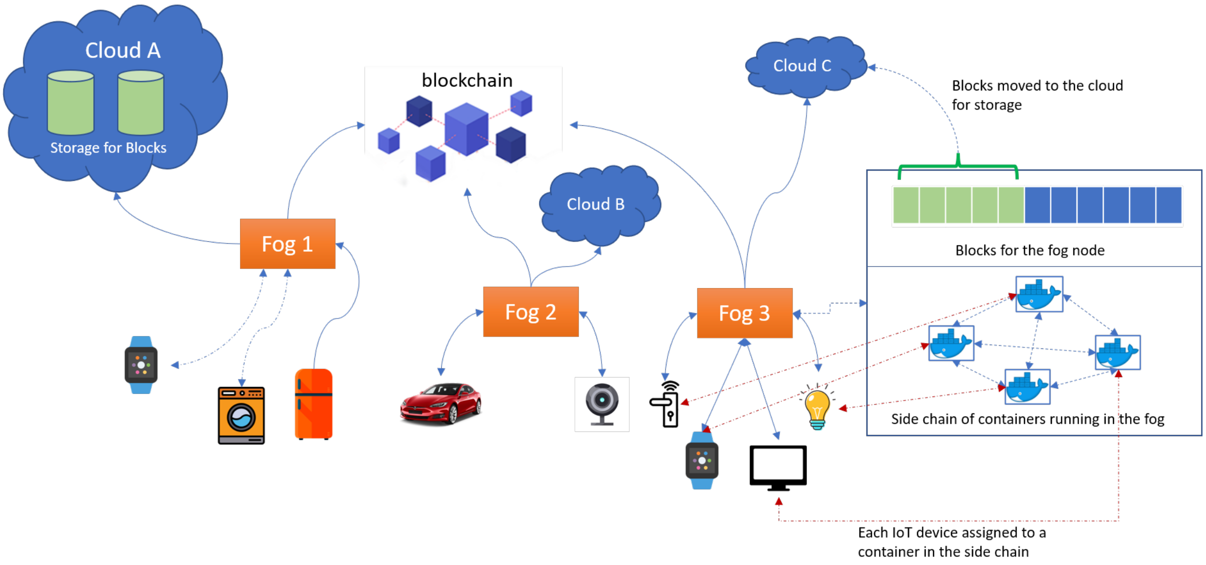
Cloud A
Storage for Blocks
blockchain
Cloud B
Cloud C
Blocks moved to the cloud for storage
Blocks for the fog node
Fog 1
Fog 2
Fog 3
Side chain of containers running in the fog
Each IoT device assigned to a container in the side chain

Figure 7. Storage optimization using side-chain and cloud storage [53].

Zheng et al. [63] proposed a blockchain data storage model that uses the IPFS (interplanetary file system) [65] to store transaction data while the hash of the data is stored in the block on the blockchain, as shown in Figure 8. The storage of all transaction data on the IPFS may result in high latency for queries as the blockchain network grows, especially in IIoT systems. Further research could explore the determination of which data may be stored locally, particularly in IIoT systems.

4.3.2. On-Chain Storage

The immutability of the blockchain ledger has a great appeal for organizations that intend to integrate this technology into their operations. However, this feature of the blockchain is a factor contributing to its storage inefficiency for systems such as IIoT. One of the interesting ideas that arose to combat this is providing flexibility when it comes to the generation of transactions. Table 5 shows a comparison of these works.

Table 5. Comparison of on-chain storage scheme optimization works.

Proposed Work	Approach	Algorithm	Storage Reduction	Query Efficiency	Latency	Limitations
Dorri et al. [66]	Transaction flexibility	MOF-BC	25%	-	max 6.5 min	High transaction processing time
Pyoung et al. [36]	Transaction flexibility	LitiChain	Average storage of 100%–142% of baseline storage	-	-	Undermines traceability and integrity of blockchain through unrecorded hashes of deleted transactions and blocks; high retention cost; complexity in determining expiry time of blocks
Qi et al. [67]	Partial storage	BFT-Store	86.8%	-	-	Long repair time for decoding, leading to longer processing time
Yu et al. [68]	Partial storage	VBG	-	0.19 s	-	Increased query cost on remote block data
Xu et al. [69]	Partial storage	Consensus Unit	75%–95%	Increased query cost	3% higher than benchmark	High latency on off-node queries
Matzutt et al. [70]	Block pruning	CoinPrune	86.98%	-	-	Limited by UTXO-based design
Wang et al. [71]	Block pruning	ESS	82.14%	-	9.21 s	Limited by UTXO-based design

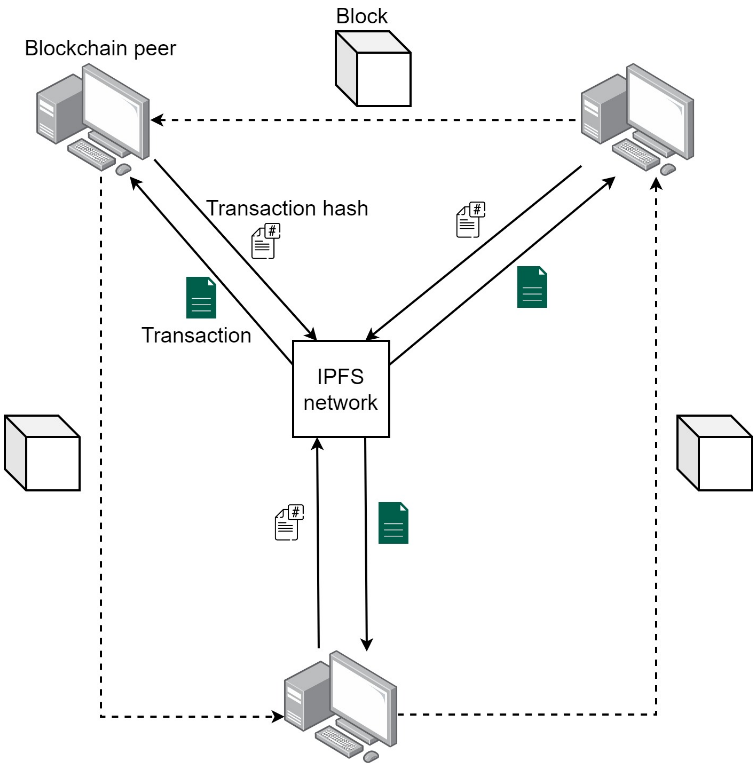

Figure 8. IPFS-based blockchain storage model [63].

Dorri et al. [66] proposed MOF-BC, a memory-optimized and flexible blockchain for large-scale IoT networks. MOF-BC provides more flexibility with its transactions and the storage of IoT data by introducing different transactions that allow IoT users and service providers to remove, summarize, or compress transaction data. As shown in Figure 9, MOF-BC introduces another layer that comprises multiple agents that handle memory optimization. The authors implemented their prototype using C++ integrated with the Crypto++ library and an SQLite database. Their results indicate a storage reduction of up to 25%. The cumulative processing times for temporary and summarizable transactions are 1.5 and 6.5 min, respectively. MOF-BC is suited for permissionless blockchain systems. It has a high impact on blockchain operations; it changes the transaction structure and introduces multiple agents that interact with the blockchain. To improve the processing time of transactions in future research, it may be prudent to reduce the intermediate steps for transaction processing and the number of agents with which peers on the network interact.

Pyoung et al. [36] proposed LitiChain, which is a scalable and lightweight blockchain architecture for edge-based IoT. The authors proposed transactions and blocks with finite lifetimes. The expired blocks are removed from the chain. While this solution may seem suitable for IIoT systems that generate a large number of transactions, LitiChain does not record the hashes of the deleted transactions or blocks, which could undermine the traceability and integrity of the blockchain.

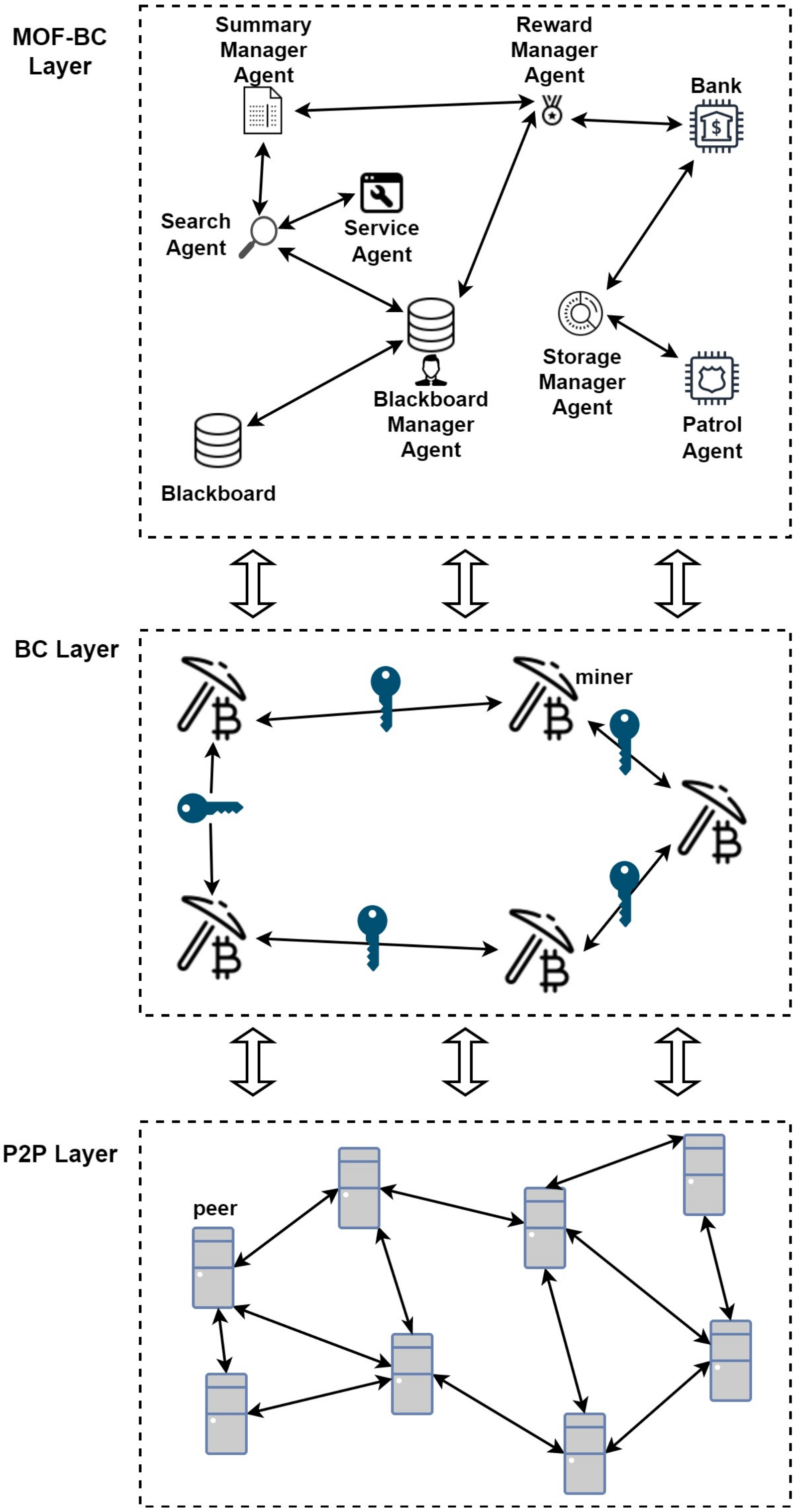

Figure 9. An overview of MOF-BC [66].

The partitioning of the blockchain network into groups such that peers store a part of the blockchain ledger that is relevant to the peer's operation and not the entire ledger radically improves the scalability of the blockchain system and the storage burden on peers.

Qi et al. [28] proposed BFT-Store, a Byzantine fault-tolerant storage engine that uses erasure coding for permissioned blockchain systems, to reduce storage consumption and improve scalability. BFT-Store uses Reed–Solomon erasure coding for storage partitioning, where the original blocks are transformed into coded chunks and stored by the nodes such that no single node stores the entire ledger. Erasure coding allows the recovery of the original data through any subset with a sufficient number of chunks. The authors implemented their prototype on the Tendermint blockchain platform with the PBFT consensus protocol and LevelDB for data persistence. Their results showed that, for a fault factor of 4 and a replica number of 3, they achieved storage savings of 86.38% using 40 nodes and a fixed block size of 1 MB. BFT-Store is suited for permissioned blockchain systems. It has a relatively high complexity and a high impact on core blockchain operations, since it introduces an erasure coding scheme and uses partial node storage instead of the conventional full replication approach. An avenue for improving this approach is to improve the repair pipeline in Byzantine environments to reduce repair time during the decoding process.

Yu et al. [68] proposed a blockchain model named Virtual Block Group (VBG) to address scalability in blockchain systems. This model combines partial node storage and distributed hash table. Multiple successive blocks in the blockchain are combined as a VBG. The nodes may store the VBG with block data or the VBG with only metadata. The hash value of the VBG and the list of storage nodes are stored in a distributed hash table among all the participating nodes. When a request for data that are not stored locally on a node is made, the request is routed to the node that contains the storage index of the data, and then the data are acquired from nodes close to the originating node. The authors implemented their prototype in C#. The VBG model is suited for either public and permissioned blockchain systems. It has a high impact on core blockchain operations and has a relatively high complexity, since it introduces partial node storage and a distributed hash table for queries. There is also increased latency when querying for block data that are not stored locally.

Xu et al. [69] proposed a consensus-unit-based storage scheme to reduce storage overhead on blockchain systems. In this approach, different nodes are organized into units called consensus units. Each consensus unit stores a full copy of the ledger, and its member nodes contribute part of their storage to store the ledger. In this manner, member nodes can query within their consensus unit, rather than the entire network, for blocks that are not stored locally. This storage scheme is demonstrated in Figure 10. Based on this idea, the authors defined a block assignment optimization problem to determine the optimal assignment of blocks in a consensus unit to ensure efficient storage and queries. They proposed three heuristic algorithms for solving this problem. They evaluated their algorithms using Blockbench, a benchmarking framework for blockchain systems. Their approach allowed a space-saving percentage to be preset at 75%, 80%, 85%, 90%, and 95%. Their results indicated a 3% lower throughput than the benchmark Ethereum design and a 3% higher latency than the benchmark Ethereum design. It also had a higher query cost when fetching data from other nodes. This approach is suited for both private and public blockchains. It has a relatively high complexity and would not be easy to implement due to the partial node storage model. Queries for block data not stored on the peer also incur more latency. Edge caching could be used to improve the performance of the schemes proposed in [68,69] by reducing the latency in fetching frequently queried data.

Another approach to storage scheme optimization is block pruning, where older blockchain data can be pruned or removed by individual nodes on the network to reduce the storage burden on those nodes [72,73]. In UTXO-based blockchains, bloating of the UTXO set with non-spendable transactions and unwanted content may be ameliorated through pruning [74].

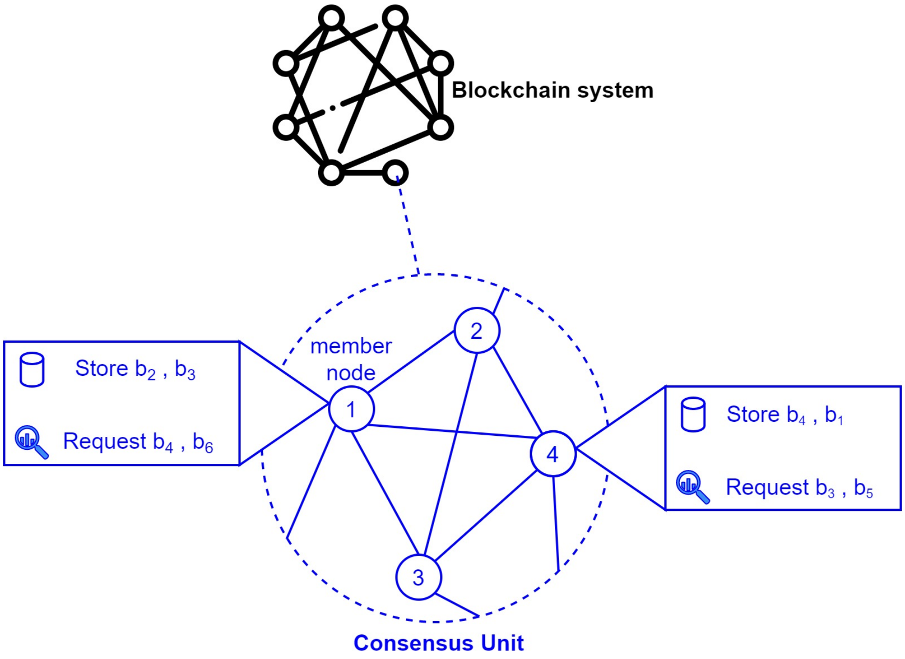

Figure 10. Consensus unit [69].

Matzutt et al. [70] proposed CoinPrune, which is a snapshot-based block pruning scheme for the Bitcoin blockchain that allows joining nodes to use only the small snapshots of the ledger for bootstrapping. This enables established nodes to prune obsolete data without affecting the network. This would help reduce the storage burden on the nodes. Wang et al. [71] proposed ESS, an efficient storage scheme for Bitcoin network scalability. They proposed the concept of UTXO-weight, which sets a weight for the unspent transactions in each block and is used to determine which blocks should be pruned. Their results showed 82.14% storage space savings on full nodes. The application of these block pruning schemes in IIoT is limited by their design, which is focused on UTXO-based blockchains, which are mainly cryptocurrency networks [75].

4.4. Machine-Learning-Enabled Schemes

As two of the innovative technologies driving advancements in information technology research, blockchain and artificial intelligence are widely used in many fields [76]. The rapid development of AI shows its ability to effectively learn from massive amounts of data. However, there have not been many works on using machine learning to optimize blockchain storage directly. In an IIoT application, the massive amounts of data generated could provide useful insights into how to better store and access transactions on the blockchain through machine learning.

Jia et al. [76] proposed an optimized data storage method for sharding-based blockchain systems that utilizes an extreme learning machine (ELM) classifier. In a sharding-based blockchain, certain nodes on the network can form a shard, which stores and processes transactions separately from other non-shard nodes. Sharding improves the scalability of blockchain systems. In this approach, nodes are separated based on three different roles: user nodes, storage nodes, and verification nodes. The user nodes submit transactions on the network, the storage nodes store the shard blockchain data, and the verification nodes ensure the reliability of the storage nodes. Based on the historical query records, number of transactions, transaction contents, and the security of a block, five features—the objective feature of a block, the objective feature of the block associated with the node, the historical popularity, the hidden popularity, and the storage requirements—are extracted and fed to the ELM classifier. The ELM classifier classifies the blocks into hot and non-hot blocks—hot blocks are frequently accessed by the node, while non-hot blocks are not frequently accessed. Hot blocks are stored locally on the node to improve the query efficiency of the block. The results of their experiments showed a better performance in terms of classification and query efficiency compared to the baseline ElasticChain system and an SVM-based model. It may be difficult to extract features for classification. Future research could look into replacing the ELM method with other effective machine learning techniques.

To the best of our knowledge, the work of Jia et al. [76] is the only work to directly optimize blockchain storage by combining sharding and the extreme learning machine (ELM) algorithm.

5. Machine Learning for Blockchain Optimization

Various works have explored the possibility of incorporating machine learning into blockchain systems to improve the performance of these systems and mitigate some of their drawbacks. This section discusses some of the works that have explored the optimization of blockchain systems with machine learning. Though these works do not focus on the optimization of storage, this review shows the potential of machine learning to improve not just the transactional throughput and latency in blockchain systems, but also how the blockchain ledger is stored and accessed.

5.1. Selection Criteria

The works explored in this section were obtained from a total of 46,912 query results. These query results were then filtered through selection criteria that determined which works best suited the theme of the discussion, which is the optimization of blockchain systems through the use of machine learning techniques. Emphasis was placed on works that did not simply incorporate machine learning techniques into blockchain-based systems, but rather used these techniques to improve the blockchain's performance. It should be noted that the purpose of this section is to show the potential for machine learning techniques to improve storage efficiency on blockchain systems by reviewing works on machine learning techniques used for blockchain optimization; thus, few works are reviewed. The criteria used are given in Table 6.

Table 6. Inclusion criteria for selected works.

No.	Criterion
1	Publication date between 2018 and 2022
2	Emphasis on blockchain performance optimization using machine learning techniques
3	Detailed descriptions of the proposed schemes and the underlying technologies

5.2. Review of Selected Works

Al-Marridi et al. [77] proposed a multi-objective optimization blockchain framework called Healthchain-RL based on deep reinforcement learning in order to optimize the real-time behavior of blockchain networks for health systems while considering medical data requirements, such as urgency and security. Healthchain-RL adaptively configures the blockchain parameters, such as the number of transactions per block, and the miners' selection based on the application requirements of urgency, security, and transaction age. The authors proposed an intelligent Blockchain Manager (BM) based on three deep reinforcement learning approaches—Deep Q-Network (DQN), Double Deep Q-Network (DDQN), and Dueling Double Deep Q-Network (D3QN) —which considers the blockchain requirements and dynamics and optimizes the latency, security, and cost of transactions in real-time. The authors noted that the D3QN approach favors unstable environments with frequent changes even though it has a higher action time. They also noted that the DQN approach is the best choice for systems where time is critical to their operation and little latency is required.

Nguyen at al. [78] proposed a cooperative task offloading and block mining (TOBM) scheme for optimizing system utility in blockchain-powered mobile edge computing (MEC). In this scheme, they proposed a cooperative deep reinforcement learning approach using a multi-agent deep deterministic policy gradient (MA-DDPG) algorithm to optimize task offloading of edge devices to the MEC server and block mining on the edge devices. The MA-DDPG scheme produced a higher system reward compared to the baseline algorithms. The proposed scheme, which uses a Proof-of-Reputation (PoR) consensus, achieved a lower block verification latency than the baseline scheme using a Delegated Proof-of-Stake (DPoS) even though the latency increased for both schemes as the number of mining nodes increased.

Fan et al. [79] proposed a DDQN-based computation offloading and resource allocation algorithm (DDQN-CORA) for blockchain-enabled mobile edge computing systems. In this work, the aim was to reduce the energy consumption and processing delay while ensuring a high transaction throughput rate of the blockchain network. The proposed work addresses the problem by optimizing offloading decisions, block size, and block interval. Their results showed that the proposed algorithm performs better than the baseline pre-fixed offloading approach and the random mapping offloading approach. The proposed algorithm showed a lower average system cost with an increase in time slots, a lower average delay cost at a low total computation capability, and a relatively lower energy cost. However, the DDQN-based approach had an extremely large training loss at first and gradually approached zero as the number of training episodes increased.

Li et al. [80] proposed a decentralized federated learning framework based on blockchain called the Blockchain-based Federated Learning framework with Committee consensus (BFLC) to address security concerns with federated learning, which involves cooperatively training a shared machine learning algorithm across multiple devices or servers. The authors proposed a storage overhead reduction scheme where nodes with inadequate storage capacity can delete older blocks locally and store the latest model and updates. This method, however, reduces the security of the blockchain.

Qiu et al. [81] proposed an adaptive blockchain architecture based on deep reinforcement learning to improve the scalability of blockchain networks. Their architecture has three layers, which are, from bottom to top: the resource layer, the coordination layer, and the application layer. The coordinator in the coordination layer dynamically allocates resources (consensus protocols, computational resources, and bandwidth) in the resource layer to different applications in the application layer based on the various QoS requirements. The authors proposed a dueling deep reinforcement learning algorithm to enable the adaptive management of the blockchain. Their results showed a convergence of the QoS value of their proposed scheme to a relatively stable and high value compared to the baseline scheme without dynamic selection of consensus protocol, computing resources, and bandwidth, respectively.

Hu et al. [82] proposed a mobile edge computing (MEC)-enabled blockchain framework for IoT networks that relies on deep-reinforcement-learning-based joint performance optimization. The authors developed an MEC and blockchain joint optimization algorithm to maximize the computational efficiency of the MEC and the transaction throughput of the blockchain. The algorithm is based on the deep deterministic policy gradient (DDPG) learning algorithm. Their results showed a higher blockchain transaction throughput and MEC computational efficiency compared to the other DQN-based and greedy approaches. However, they noted that the algorithm compromises on the average block interval to reduce power consumption.

Table 7 provides a summary of the works described in this section. The ability of machine learning agents to find relationships in large sets of data and through interactions in an environment to make useful inferences can be capitalized upon to improve the storage efficiency of blockchain–IIoT systems. By developing optimization frameworks based on different parameters of the blockchain and of the constituent nodes, such as latency, storage capabilities, and computing power, these systems can adapt to different requirements and configurations to improve the storage and query efficiency, as well as the transaction throughput.

Table 7. Summary of works using machine learning for blockchain optimization.

Proposed Work	Machine Learning Algorithm	Focus	Contributions
Al-Marridi et al. [77] Healthchain-RL	DQN, DDQN, D3QN	Transaction throughput, security	Relatively high average accumulated rewards compared to greedy and random selection algorithms
Nguyen et al. [78] TOBM	MA-DDPG	Consensus algorithm	Relatively high system reward and lower block mining latency compared to the baseline algorithms of DDPG, DQN, and Actor–Critic
Fan et al. [79] DDQN-CORA	DDQN	Transaction throughput	Lower average system, delay, and energy costs under different computation capabilities
Li et al. [80] BFLC	Federated learning	Security, storage efficiency	Demonstrates stability by achieving high accuracy under malicious attacks
Qiu et al. [81] Adaptive blockchain	Dueling deep reinforcement learning	Consensus algorithm, computation, and bandwidth	Converges to a higher QoS value than the baselines
Hu et al. [82] MEC-enabled blockchain framework	DDPG-based	Transaction throughput	Higher average rewards compared to baseline schemes implying enhanced transaction processing of MEC-enabled blockchain network

6. Open Issues and Future Directions

Many solutions have been proposed over the years to mitigate the storage concerns of blockchain technology and its application in large-scale networks, such as IIoT. These solutions have great merit and can provide great results. Compression-based solutions generally allow the reduction of storage overhead without relying on other technologies, such as cloud computing, which have a more centralized model. This aspect facilitates decentralization, which is a major facet and security feature of blockchain systems. A major drawback of these solutions lies in their long-term feasibility. IIoT networks generate many transactions and would ensure the rapid growth of the blockchain ledger. The accumulation of the compressed results over time might not provide enough storage saving on peers. The effect of compression on the processing of transactions should also be considered. IIoT transactions may have certain latency requirements that could be adversely affected by the compression techniques used. Summarization solutions generate summary blocks that depend on redundancies in block data and transactions that transfer entities between parties. These solutions suit financial systems and cryptocurrency systems, but may be difficult to implement for IIoT systems. Storage optimization solutions focus on making the optimal use of storage space for blockchain systems and sometimes combine several paradigms and technologies to improve the storage overhead on blockchain systems. Generally, these solutions are usually proposed as part of a new blockchain architecture, as they tend to be complex and difficult to incorporate into existing blockchain systems. In terms of long-term feasibility, these solutions may provide the best outlook. A major drawback of these solutions includes the introduction of centralized models, such as cloud storage, to save space on local peers or blockchain shards, which separate the blockchain network into organizations that store separate ledgers and process transactions separately. These solutions may also result in longer processing times for transactions due to the computational needs of some algorithms.

There are several possible future directions for researchers looking to improve the storage efficiency of blockchain–IIoT applications:

- Most proposed blockchain storage solutions are specific to certain blockchain platforms and certain types of blockchains, or they require a new blockchain architecture. This limits the application of these solutions in most cases. Solutions that aim at reducing storage overhead on any type of blockchain should be considered. Standard summary blocks should be designed for various blockchain systems to improve the applicability of summarization-based schemes in different systems.
- Another issue of note is that many storage solutions have a short-sighted approach, especially in the case of compression- and summarization-based methods. This is due to the fact that rapid growth of the blockchain ledger, particularly in large-scale networks such as IIoT, would accumulate compression results and, thus, result in unmitigated storage overhead over time.
- Since the volume of data and rate of data generation in IIoT systems are much greater in comparison to those in other applications, compression techniques should be designed to achieve an effective compression ratio regardless of file size and formats. The representation of data before compression should also be examined, since the appropriate representation of data can affect the performance of a compression technique [83]. Efficient representation of data would yield better compression ratios.
- Storage scheme optimization can still be improved for blockchain–IIoT systems. Increased query times due to the remote storage of block data on the cloud or on other blockchain peers, such as in [68,69], can be alleviated through the use of edge caching, where edge servers store block data that may be queried but are not necessarily vital to the operation of the peer. The security of such an approach must then be taken into consideration to avoid malicious attacks. The transaction flexibility offered in [66] can be implemented while ensuring short processing times. This can be achieved by reducing the intermediate steps that the transactions go through and by providing a decentralized approach to the agents that handle memory optimization on the network.
- The use of machine learning can provide great benefits to blockchain–IIoT systems. As in [52,53], where evolutionary algorithms were used to determine which blocks should be offloaded to a cloud service to save space on local peers, machine learning techniques such as deep reinforcement learning could be used for the classification or determination of less frequently queried blocks. This might be favorable in comparison to the long iterative process of evolutionary algorithms. Machine learning could also be combined with summarization- and compression-based schemes, where different transactions may be compressed or summarized using specific techniques to ensure the best possible compression ratio.

7. Conclusions

The rapid growth of blockchain technology in recent years suggests that it will emerge in even more applications in the coming years. The integration of blockchain into IoT systems brings many benefits and certain drawbacks as the nodes on the blockchain grow in number. One of the major drawbacks is the storage overhead incurred with the growth of the blockchain network. The immutability of the blockchain ledger, though a great security feature, ensures that the application of the blockchain technology in high transaction environments such as IIoT would incur high storage overheads on participating peers.

This paper explores the storage and scalability concerns of the blockchain technology and how these concerns affect its integration into IIoT. The high storage demand of blockchain systems restricts the participation of resource-constrained IIoT devices, affects decentralization as fewer nodes are able to join the network, and reduces the storage efficiency of underlying databases.

Several solutions have emerged through research to address these concerns. These solutions, which include summarization-based, compression-based, and storage scheme optimization methods, are necessary to enable the further development of blockchain–IIoT integration. However, these solutions have shortcomings that reduce their effectiveness. Compression-based schemes produce compressed blocks or data that accumulate over time and may not ensure enough storage savings on peers. This can be alleviated by designing compression techniques that provide an efficient representation of data for IIoT systems to yield better compression ratios. Summarization-based schemes reduce redundancy in block data by using the net change in transferring entities between parties and, thus, are better suited for financial systems than for IIoT systems. Standard summary blocks that suit different blockchain applications would be beneficial in reducing the storage overhead. Storage optimization schemes combine different technologies to ensure the optimal use of storage for blockchain systems but introduce centralization and weaken the inherent security of the blockchain. This work proposes the use of machine learning techniques in addressing the problem of storage efficiency. This approach, which has rarely been explored by researchers, represents the next step in evolving these schemes to achieve even greater results in terms of storage efficiency.

It is essential that research into solving the storage overhead problem of the blockchain technology continues to ensure seamless integration with other technologies, such as IoT and edge computing.

Author Contributions: Conceptualization, N.K.A.-M., E.T.T., A.S. and A.S.A.; methodology, N.K.A.-M., H.N.-M., A.-R.A., D.W. and E.K.; validation, A.S., D.W. and E.K.; formal analysis, N.K.A.-M., E.T.T., A.S., A.S.A., H.N.-M., A.-R.A., D.W. and E.K.; investigation, E.T.T.; resources, D.W. and E.K.; writing—original draft preparation, N.K.A.-M.; writing—review and editing, N.K.A.-M., E.T.T., A.S., A.S.A., H.N.-M., A.-R.A., D.W. and E.K.; visualization, A.S.A. and H.N.-M.; supervision, E.T.T., A.S., A.S.A., H.N.-M., and A.-R.A.; project administration, H.N.-M. and A.-R.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from the German Federal Ministry of Research and Education (Bundesministerium für Bildung und Forschung, BMBF) and the German Academic Exchange Service (Deutscher Akademischer Austauschdienst, DAAD). This paper was written as part of the Distributed IoT Platforms for Safe Food Production in Education, Research, and Industry (DIPPER) project, which is co-financed by the BMBF (Förderkennzeichen: 01DG21017) and DAAD (Projekt-ID: 57557211).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The article processing charge was funded by the Baden-Wuerttemberg Ministry of Science, Research and Culture and the Offenburg University in the funding programme Open Access Publishing.

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

References

1. Novo, O. Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT. *IEEE Internet Things J.* **2018**, *5*, 1184–1195. [CrossRef]
2. Guo, S.; Dai, Y.; Guo, S.; Qiu, X.; Qi, F. Blockchain Meets Edge Computing: Stackelberg Game and Double Auction Based Task Offloading for Mobile Blockchain. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5549–5561. [CrossRef]
3. Jeong, Y.S. Blockchain Processing Technique Based on Multiple Hash Chains for Minimizing Integrity Errors of IoT Data in Cloud Environments. *Sensors* **2021**, *21*, 4679. [CrossRef] [PubMed]
4. Al-Rakhami, M.S.; Al-Mashari, M. A Blockchain-Based Trust Model for the Internet of Things Supply Chain Management. *Sensors* **2021**, *21*, 1759. [CrossRef] [PubMed]
5. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Available online: http://bitcoin.org/bitcoin.pdf (accessed on 11 December 2021).
6. Wood, G. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Proj. Yellow Pap.* **2014**, *151*, 1–32.
7. Li, X.; Jiang, P.; Chen, T.; Luo, X.; Wen, Q. A survey on the security of blockchain systems. *Future Gener. Comput. Syst.* **2020**, *107*, 841–853. [CrossRef]

8. Pilares, I.C.A.; Azam, S.; Akbulut, S.; Jonkman, M.; Shanmugam, B. Addressing the Challenges of Electronic Health Records Using Blockchain and IPFS. *Sensors* **2022**, *22*, 4032. [CrossRef]
9. Mani, V.; Manickam, P.; Alotaibi, Y.; Alghamdi, S.; Khalaf, O.I. Hyperledger Healthchain: Patient-Centric IPFS-Based Storage of Health Records. *Electronics* **2021**, *10*, 3003. [CrossRef]
10. Bhat, S.A.; Huang, N.F.; Sofi, I.B.; Sultan, M. Agriculture-Food Supply Chain Management Based on Blockchain and IoT: A Narrative on Enterprise Blockchain Interoperability. *Agriculture* **2022**, *12*, 40. [CrossRef]
11. Nabeeh, N.A.; Abdel-Basset, M.; Gamal, A.; Chang, V. Evaluation of Production of Digital Twins Based on Blockchain Technology. *Electronics* **2022**, *11*, 1268. [CrossRef]
12. Song, T.; Li, R.; Mei, B.; Yu, J.; Xing, X.; Cheng, X. A Privacy Preserving Communication Protocol for IoT Applications in Smart Homes. *IEEE Internet Things J.* **2017**, *4*, 1844–1852. [CrossRef]
13. Yu, S.; Jho, N.; Park, Y. Lightweight Three-Factor-Based Privacy- Preserving Authentication Scheme for IoT-Enabled Smart Homes. *IEEE Access* **2021**, *9*, 126186–126197. [CrossRef]
14. Friha, O.; Ferrag, M.A.; Shu, L.; Maglaras, L.; Wang, X. Internet of Things for the Future of Smart Agriculture: A Comprehensive Survey of Emerging Technologies. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 718–752. [CrossRef]
15. Ayaz, M.; Ammad-Uddin, M.; Sharif, Z.; Mansour, A.; Aggoune, E.H.M. Internet-of-Things (IoT)-Based Smart Agriculture: Toward Making the Fields Talk. *IEEE Access* **2019**, *7*, 129551–129583. [CrossRef]
16. Qazi, S.; Khawaja, B.A.; Farooq, Q.U. IoT-Equipped and AI-Enabled Next Generation Smart Agriculture: A Critical Review, Current Challenges and Future Trends. *IEEE Access* **2022**, *10*, 21219–21235. [CrossRef]
17. Aivaliotis, V.; Tsantikidou, K.; Sklavos, N. IoT-Based Multi-Sensor Healthcare Architectures and a Lightweight-Based Privacy Scheme. *Sensors* **2022**, *22*, 4269. [CrossRef]
18. Shaban, M.; Alsharekh, M.F. Design of a Smart Distribution Panelboard Using IoT Connectivity and Machine Learning Techniques. *Energies* **2022**, *15*, 3658. [CrossRef]
19. Alfalouji, Q.; Schranz, T.; Kümpel, A.; Schraven, M.; Storek, T.; Gross, S.; Monti, A.; Müller, D.; Schweiger, G. IoT Middleware Platforms for Smart Energy Systems: An Empirical Expert Survey. *Buildings* **2022**, *12*, 526. [CrossRef]
20. Costantini, A.; Di Modica, G.; Ahouangonou, J.C.; Duma, D.C.; Martelli, B.; Galletti, M.; Antonacci, M.; Nehls, D.; Bellavista, P.; Delamarre, C.; et al. IoTwins: Toward Implementation of Distributed Digital Twins in Industry 4.0 Settings. *Computers* **2022**, *11*, 67. [CrossRef]
21. Rudenko, R.; Pires, I.M.; Oliveira, P.; Barroso, J.; Reis, A. A Brief Review on Internet of Things, Industry 4.0 and Cybersecurity. *Electronics* **2022**, *11*, 1742. [CrossRef]
22. Senadeera, S.D.A.P.; Kyi, S.; Sirisung, T.; Pongsupan, W.; Taparugssanagorn, A.; Dailey, M.N.; Wai, T.A. Cost-Effective and Low Power IoT-Based Paper Supply Monitoring System: An Application Modeling Approach. *J. Low Power Electron. Appl.* **2021**, *11*, 46. [CrossRef]
23. Rahman, L.F.; Alam, L.; Marufuzzaman, M.; Sumaila, U.R. Traceability of Sustainability and Safety in Fishery Supply Chain Management Systems Using Radio Frequency Identification Technology. *Foods* **2021**, *10*, 2265. [CrossRef]
24. What Is Industrial IoT (IIoT)? Available online: https://www.cisco.com/c/en/us/solutions/internet-of-things/what-is-industrial-iot.html (accessed on 19 March 2022).
25. Nartey, C.; Tchao, E.T.; Gadze, J.D.; Keelson, E.; Klogo, G.S.; Kommey, B.; Diawuo, K. On Blockchain and IoT Integration Platforms: Current Implementation Challenges and Future Perspectives. *Wirel. Commun. Mob. Comput.* **2021**, 1–25 . [CrossRef]
26. Zaman, U.; Imran; Mehmood, F.; Iqbal, N.; Kim, J.; Ibrahim, M. Towards Secure and Intelligent Internet of Health Things: A Survey of Enabling Technologies and Applications. *Electronics* **2022**, *11*, 1893. [CrossRef]
27. Benefits of Blockchain. Available online: https://www.ibm.com/topics/benefits-of-blockchain (accessed on 19 March 2022).
28. Qi, S.; Lu, Y.; Zheng, Y.; Li, Y.; Chen, X. Cpds: Enabling Compressed and Private Data Sharing for Industrial Internet of Things over Blockchain. *IEEE Trans. Ind. Inform.* **2021**, *17*, 2376–2387. [CrossRef]
29. Guru, D.; Perumal, S.; Varadarajan, V. Approaches towards blockchain innovation: A survey and future directions. *Electronics* **2021**, *10*, 1219. [CrossRef]
30. Hepp, T.; Sharinghousen, M.; Ehret, P.; Schoenhals, A.; Gipp, B. On-chain vs. off-chain storage for supply-and blockchain integration. *IT - Inf. Technol.* **2021**, *60*, 283–291. [CrossRef]
31. Zhang, C.; Ni, Z.; Xu, Y.; Luo, E.; Chen, L.; Zhang, Y. A trustworthy industrial data management scheme based on redactable blockchain. *J. Parallel Distrib. Comput.* **2021**, *152*, 167–176. [CrossRef]
32. Antwi, R.; Gadze, J.D.; Tchao, E.T.; Sikora, A.; Nunoo-Mensah, H.; Agbemenu, A.S.; Obour Agyekum, K.O.B.; Agyemang, J.O.; Welte, D.; Keelson, E. A Survey on Network Optimization Techniques for Blockchain Systems. *Algorithms* **2022**, *15*, 193. [CrossRef]
33. Blockchain Challenges and Opportunities: A Survey. *Int. J. Web Grid Serv.* **2018**, *14*, 352–375. [CrossRef]
34. IBM. Storage Needs for Blockchain Technology—Point of View. Available online: https://www.ibm.com/downloads/cas/LA8XBQGR (accessed on 19 March 2022) .
35. Sundarakani, B.; Ajaykumar, A.; Gunasekaran, A. Big data driven supply chain design and applications for blockchain: An action research using case study approach. *Omega* **2021**, *102*, 102452. [CrossRef]
36. Pyoung, C.K.; Baek, S.J. Blockchain of Finite-Lifetime Blocks with Applications to Edge-Based IoT. *IEEE Internet Things J.* **2020**, *7*, 2102–2116. [CrossRef]

37. Dai, H.N.; Zheng, Z.; Zhang, Y. Blockchain for Internet of Things: A Survey. *IEEE Internet Things J.* **2019**, *6*, 8076–8094. [CrossRef]
38. Chen, X.; Zhang, K.; Liang, X.; Qiu, W.; Zhang, Z.; Tu, D. HyperBSA: A high-performance consortium blockchain storage architecture for massive data. *IEEE Access* **2020**, *8*, 178402–178413. [CrossRef]
39. Cao, B.; Wang, X.; Zhang, W.; Song, H.; Lv, Z. A Many-Objective Optimization Model of Industrial Internet of Things Based on Private Blockchain. *IEEE Netw.* **2020**, *34*, 78–83. [CrossRef]
40. Yang, F.; Qiao, Y.; Qi, Y.; Bo, J.; Wang, X. BACS: Blockchain and AutoML-based technology for efficient credit scoring classification. *Ann. Oper. Res.* **2022**, 1–21. [CrossRef] [PubMed]
41. Jiang, S.; Li, J.; Gong, S.; Yan, J.; Yan, G.; Sun, Y.; Li, X. BZIP: A compact data memory system for UTXO-based blockchains. *J. Syst. Archit.* **2020**, *109*, 101809. [CrossRef]
42. Zhou, Q.; Huang, H.; Zheng, Z.; Bian, J. Solutions to Scalability of Blockchain: A Survey. *IEEE Access* **2020**, *8*, 16440–16455. [CrossRef]
43. Croman, K.; Decker, C.; Eyal, I.; Gencer, A.E.; Juels, A.; Kosba, A.; Miller, A.; Saxena, P.; Shi, E.; Gün Sirer, E.; et al. On Scaling Decentralized Blockchains. In *Proceedings of the Financial Cryptography and Data Security*; Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; pp. 106–125.
44. Khan, D.; Jung, L.T.; Hashmani, M.A. Systematic Literature Review of Challenges in Blockchain Scalability. *Appl. Sci.* **2021**, *11*, 9372. [CrossRef]
45. Zhang, J.; Zhong, S.; Wang, J.; Wang, L. An systematic study on blockchain transaction databases storage and optimization. In Proceedings of the 2020 IEEE International Conference on Parallel and Distributed Processing with Applications, Big Data and Cloud Computing, Sustainable Computing and Communications, Social Computing and Networking (ISPA/BDCloud/SocialCom/SustainCom), Exeter, UK, 17–19 December 2020; pp. 298–304. [CrossRef]
46. Guo, Z.; Gao, Z.; Mei, H.; Zhao, M.; Yang, J. Design and optimization for storage mechanism of the public blockchain based on redundant residual number system. *IEEE Access* **2019**, *7*, 98546–98554. [CrossRef]
47. Matzutt, R.; Hohlfeld, O.; Henze, M.; Rawiel, R.; Ziegeldorf, J.H.; Wehrle, K. POSTER: I Don't Want That Content! On the Risks of Exploiting Bitcoin's Blockchain as a Content Store. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, Vienna, Austria, 24–28 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 1769–1771. [CrossRef]
48. Matzutt, R.; Hiller, J.; Henze, M.; Ziegeldorf, J.H.; Müllmann, D.; Hohlfeld, O.; Wehrle, K. A Quantitative Analysis of the Impact of Arbitrary Blockchain Content on Bitcoin. In *Financial Cryptography and Data Security*; Meiklejohn, S., Sako, K., Eds.; Springer: Berlin/Heidelberg, Germany, 2018; pp. 420–438.
49. Sward, A.; Vecna, I.; Stonedahl, F. Data Insertion in Bitcoin's Blockchain. *Ledger* **2018**, *3* , 1–23. [CrossRef]
50. Liu, T.; Wu, J.; Li, J.; Li, J. Secure and balanced scheme for non-local data storage in blockchain network. In Proceedings of the 2019 IEEE 21st International Conference on High Performance Computing and Communications, IEEE 17th International Conference on Smart City, IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Zhangjiajie, China, 10–12 August 2019 ; pp. 2424–2427. [CrossRef]
51. Hafid, A.; Hafid, A.S.; Samih, M. Scaling Blockchains: A Comprehensive Survey. *IEEE Access* **2020**, *8*, 125244–125262. [CrossRef]
52. Xu, M.; Feng, G.; Ren, Y.; Zhang, X. On Cloud Storage Optimization of Blockchain with a Clustering-Based Genetic Algorithm. *IEEE Internet Things J.* **2020**, *7*, 8547–8558. [CrossRef]
53. Nartey, C.; Tchao, E.T.; Gadze, J.D.; Yeboah-Akowuah, B.; Nunoo-Mensah, H.; Welte, D.; Sikora, A. Blockchain-IoT peer device storage optimization using an advanced time-variant multi-objective particle swarm optimization algorithm. *EURASIP J. Wirel. Commun. Netw.* **2022**, *2022*, 1–27. [CrossRef]
54. Kim, T.; Lee, S.; Kwon, Y.; Noh, J.; Kim, S.; Cho, S. SELCOM: Selective Compression Scheme for Lightweight Nodes in Blockchain System. *IEEE Access* **2020**, *8*, 225613–225626. [CrossRef]
55. Spataru, A.L.; Pungila, C.P.; Radovancovici, M. A high-performance native approach to adaptive blockchain smart-contract transmission and execution. *Inf. Process. Manag.* **2021**, *58*, 1–15. [CrossRef]
56. Chen, X.; Lin, S.; Yu, N. Bitcoin Blockchain Compression Algorithm for Blank Node Synchronization. In Proceedings of the 2019 11th International Conference on Wireless Communications and Signal Processing, WCSP 2019, Xi'an, China, 23–25 October 2019; pp. 10–15. [CrossRef]
57. Marsalek, A.; Zefferer, T.; Fasllija, E.; Ziegler, D. Tackling data inefficiency: Compressing the bitcoin blockchain. In Proceedings of the 2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering, TrustCom/BigDataSE 2019, Rotorua, New Zealand, 5–8 August 2019; pp. 626–633. [CrossRef]
58. Yu, B.; Li, X.; Zhao, H. PoW-BC: A PoW consensus protocol based on block compression. *KSII Trans. Internet Inf. Syst.* **2021**, *15*, 1389–1408. [CrossRef]
59. Ding, D.; Jiang, X.; Wang, J.; Wang, H.; Zhang, X.; Sun, Y. Txilm: Lossy Block Compression with Salted Short Hashing. *arXiv* **2019**, arXiv:1906.06500 .
60. Gai, F.; Niu, J.; Ali Tabatabaee, S.; Feng, C.; Jalalzai, M. Cumulus: A Secure BFT-based Sidechain for Off-chain Scaling. In Proceedings of the 2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS), Tokyo, Japan, 25–28 June 2021; pp. 1–6. [CrossRef]

61. Palai, A.; Vora, M.; Shah, A. Empowering Light Nodes in Blockchains with Block Summarization. In Proceedings of the 2018 9th IFIP International Conference on New Technologies, Mobility and Security, NTMS 2018, Paris, France, 26–28 February 2018; pp. 1–5. [CrossRef]
62. Nadiya, U.; Mutijarsa, K.; Rizqi, C.Y. Block Summarization and Compression in Bitcoin Blockchain. In Proceedings of the 2018 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 23–24 October 2018; pp. 1–4. [CrossRef]
63. Zheng, Q.; Li, Y.; Chen, P.; Dong, X. An Innovative IPFS-Based Storage Model for Blockchain. In Proceedings of the 2018 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2018, Santiago, Chile, 3–6 December 2018; pp. 704–708. [CrossRef]
64. Li, K.; Zhang, T.; Wang, R. Deep Reinforcement Learning for Multiobjective Optimization. *IEEE Trans. Cybern.* **2021**, *51*, 3103–3114. [CrossRef]
65. IPFS. Available online: https://ipfs.io/ (accessed on 19 March 2022).
66. Dorri, A.; Kanhere, S.S.; Jurdak, R. MOF-BC: A memory optimized and flexible blockchain for large scale networks. *Future Gener. Comput. Syst.* **2019**, *92*, 357–373. [CrossRef]
67. Qi, X.; Zhang, Z.; Jin, C.; Zhou, A. A reliable storage partition for permissioned blockchain. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 14–27. [CrossRef]
68. Yu, B.; Li, X.; Zhao, H. Virtual block group: A scalable blockchain model with partial node storage and distributed hash table. *Comput. J.* **2021**, *63*, 1524–1536. [CrossRef]
69. Xu, Z.; Han, S.; Chen, L. CUb, a consensus unit-based storage scheme for blockchain system. In Proceedings of the IEEE 34th International Conference on Data Engineering, ICDE 2018, Paris, France, 16–19 April 2018; pp. 173–184. [CrossRef]
70. Matzutt, R.; Kalde, B.; Pennekamp, J.; Drichel, A.; Henze, M.; Wehrle, K. CoinPrune: Shrinking Bitcoin's Blockchain Retrospectively. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 3064–3078. [CrossRef]
71. Wang, X.; Wang, C.; Zhou, K.; Cheng, H. ESS: An Efficient Storage Scheme for Improving the Scalability of Bitcoin Network. *IEEE Trans. Netw. Serv. Manag.* **2021**, *4537*, 1–12. [CrossRef]
72. Florian, M.; Henningsen, S.; Beaucamp, S.; Scheuermann, B. Erasing Data from Blockchain Nodes. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS PW), Stockholm, Sweden, 17–19 June 2019; pp. 367–376. [CrossRef]
73. Bitcoin Core Version 0.11.0 Released. Available online: https://bitcoin.org/en/release/v0.11.0 (accessed on 19 March 2022).
74. Matzutt, R.; Henze, M.; Ziegeldorf, J.H.; Hiller, J.; Wehrle, K. Thwarting Unwanted Blockchain Content Insertion. In Proceedings of the 2018 IEEE International Conference on Cloud Engineering (IC2E), Orlando, FL, USA, 17–20 April 2018; pp. 364–370. [CrossRef]
75. UTXO vs. Account Model. Available online: https://academy.horizen.io/technology/expert/utxo-vs-account-model/ (accessed on 19 March 2022).
76. Jia, D.; Xin, J.; Wang, Z.; Wang, G. Optimized Data Storage Method for Sharding-Based Blockchain. *IEEE Access* **2021**, *9*, 67890–67900. [CrossRef]
77. Al-Marridi, A.Z.; Mohamed, A.; Erbad, A. Reinforcement learning approaches for efficient and secure blockchain-powered smart health systems. *Comput. Netw.* **2021**, *197*, 108279. [CrossRef]
78. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Poor, H.V. Utility Optimization for Blockchain Empowered Edge Computing with Deep Reinforcement Learning. In Proceedings of the IEEE International Conference on Communications, Virtual, 14–23 June 2021; pp. 1–6. [CrossRef]
79. Fan, W.; Zhang, W.; Wang, L.; Liu, T.; Zhang, G. Joint Offloading and Resource Allocation in Cooperative Blockchain-Enabled MEC System. In Proceedings of the ACM TURC 2021, Hefei, China, 30 July 2021; pp. 136–140. [CrossRef]
80. Li, Y.; Chen, C.; Liu, N.; Huang, H.; Zheng, Z.; Yan, Q. A Blockchain-Based Decentralized Federated Learning Framework with Committee Consensus. *IEEE Netw.* **2021**, *35*, 234–241. [CrossRef]
81. Qiu, C.; Ren, X.; Cao, Y.; Mai, T. Deep Reinforcement Learning Empowered Adaptivity for Future Blockchain Networks. *IEEE Open J. Comput. Soc.* **2020**, *2*, 99–105. [CrossRef]
82. Hu, Z.; Gao, H.; Wang, T.; Han, D.; Lu, Y. Joint Optimization for Mobile Edge Computing-Enabled Blockchain Systems: A Deep Reinforcement Learning Approach. *Sensors* **2022**, *22*, 3217. [CrossRef]
83. Jayasankar, U.; Thirumal, V.; Ponnurangam, D. A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications. *J. King Saud Univ. Comput. Inf. Sci.* **2021**, *33*, 119–140. [CrossRef]