Technologies for Improving Storage Efficiency in Blockchain-Based IIoT

Subjects: Computer Science, Artificial Intelligence | Engineering, Electrical & Electronic

Contributor: Nana Kwadwo Akrasi-Mensah , Eric Tutu Tchao , Axel Sikora , Andrew Selasi Agbemenu , Henry Nunoo-Mensah , Abdul-Rahman Ahmed , Dominik Welte , Eliel Keelson

The Internet of Things (IoT) and blockchain have contributed to massive advancements in the fields to which they have been applied. The benefits of the blockchain, which include enhanced security, transparency, and greater traceability, make it a promising technology for integration with IIoT, which has long had issues with security. However, there are several issues that limit the integration of blockchain into Industrial Internet of Things (IIoT) systems. One of these issues is the huge storage requirement of the blockchain. There are several solutions 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.

blockchain | IIoT | scalability | storage efficiency | storage optimization | compression

1. 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 [1]; thus, the throughput of transactions is much lower compared to more traditional database-based systems [2][3][4].

The Industrial Internet of Things (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 [5][6][7][8][9].

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 [10]. 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 1** [4].

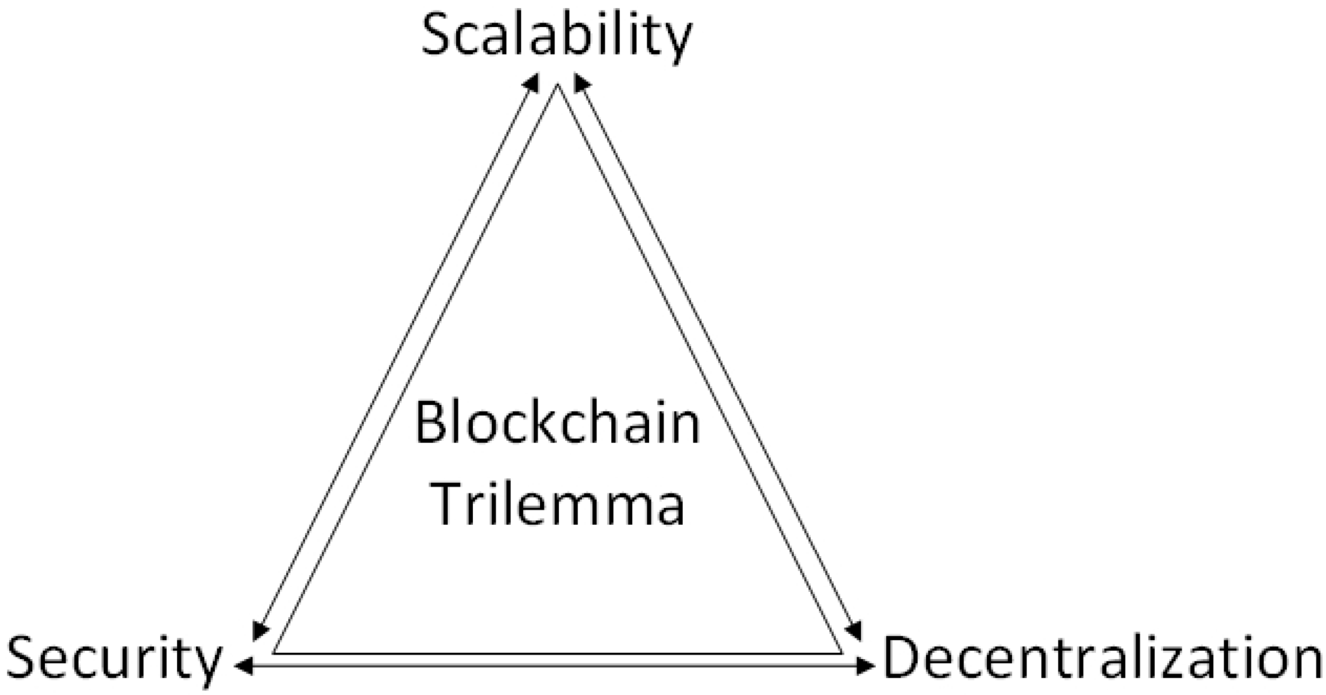

Figure 1. The blockchain trilemma.

The blockchain trilemma proposes that tradeoffs among the decentralization, scalability, and security of a blockchain system are inevitable [4][11]. 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 [4].

2. 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.

2.1. 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 1** shows a comparison of these schemes.

Table 1. Comparison of compression-based schemes.

Proposed Work	Approach	Algorithm	Compression Ratio/ Storage Reduction	Limitations
Qi et al. [12]	Data Compression	Tree-based key-value compression	4–9×	May have a low compression ratio for large product record data
Kim et al. [13]	Block Compression	Block Merkle Tree	76.02% reduction	Sidechain requires synchronization between nodes
Spataru et al. [14]	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. [15]	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. [16]	Block Compression	Snapshot block	93% reduction	Accumulation of compression results over time, suitable for UTXO-based blockchains
Yu et al. [17]	Block Compression	Deflate algorithm	30.53%–42.16% of original block	Increased mining difficulty
Ding et al. [18]	Block Compression	Txilm Protocol	8	Increased latency

2.1.1. 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. [13] proposed SELCOM, a selective compression scheme using a Block Merkle Tree, for lightweight nodes in blockchain systems. As shown in **Figure 2**, SELCOM allows nodes to maintain blocks selectively through a second chain called a checkpoint chain. It uses BMT to

References

1. 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.
2. Zhou, Q.; Huang, H.; Zheng, Z.; Bian, J. Solutions to Scalability of Blockchain: A Survey. IEEE Access 2020, 8, 16440–16455.

compress several blocks into a checkpoint. The compressed blocks can then be selectively removed or maintained depending on each node. The results indicated an average storage reduction of 76.92%. The maintenance of a security chain [illegible] complexity, as synchronization between [illegible] for this [illegible] is required. Unlike other works, the scholars proposed an update mechanism to continually accumulate 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 [19].

3. 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.
4. Khan, D.; Jung, L.T.; Hashmani, M.A. Systematic Literature Review of Challenges in Blockchain Scalability. Appl. Sci. 2021, 11, 9372.
5. 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.
6. 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.
7. 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.
8. 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.
9. Sward, A.; Vecna, I.; Stonedahl, F. Data Insertion in Bitcoin's Blockchain. Ledger 2018, 3, 1–23.
10. 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.
11. Hafid, A.; Hafid, A.S.; Samih, M. Scaling Blockchains: A Comprehensive Survey. IEEE Access 2020, 8, 125244–125262.
12. 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.
13. 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.

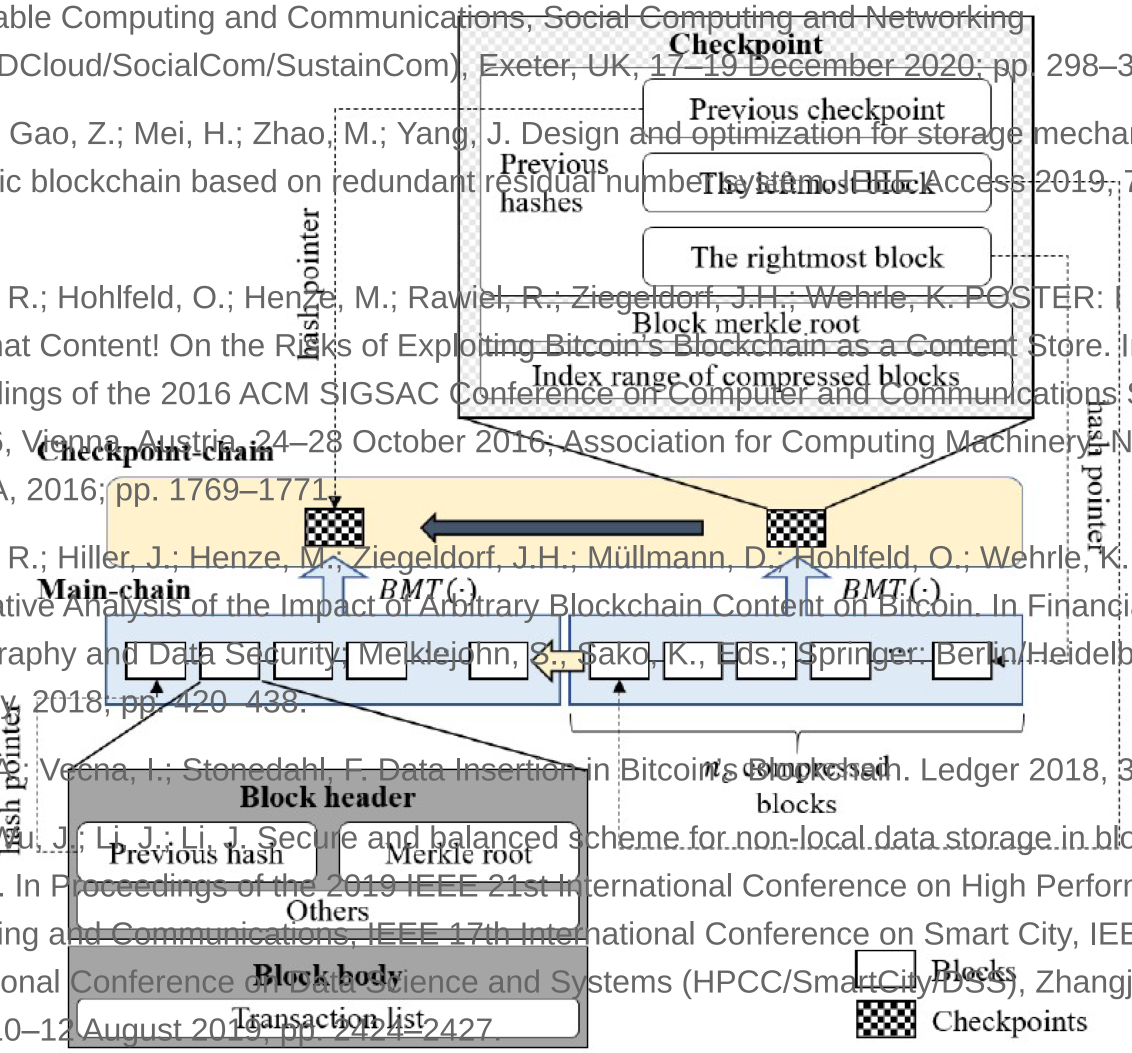

Figure 2. SELCOM [13].

2.1.2. Data Compression

Some works have proposed the compression of product data before they are encapsulated in blockchain transactions. Qi et al. [12] 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 3**, the scholars employ an off-chain architecture, 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 scholars 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.

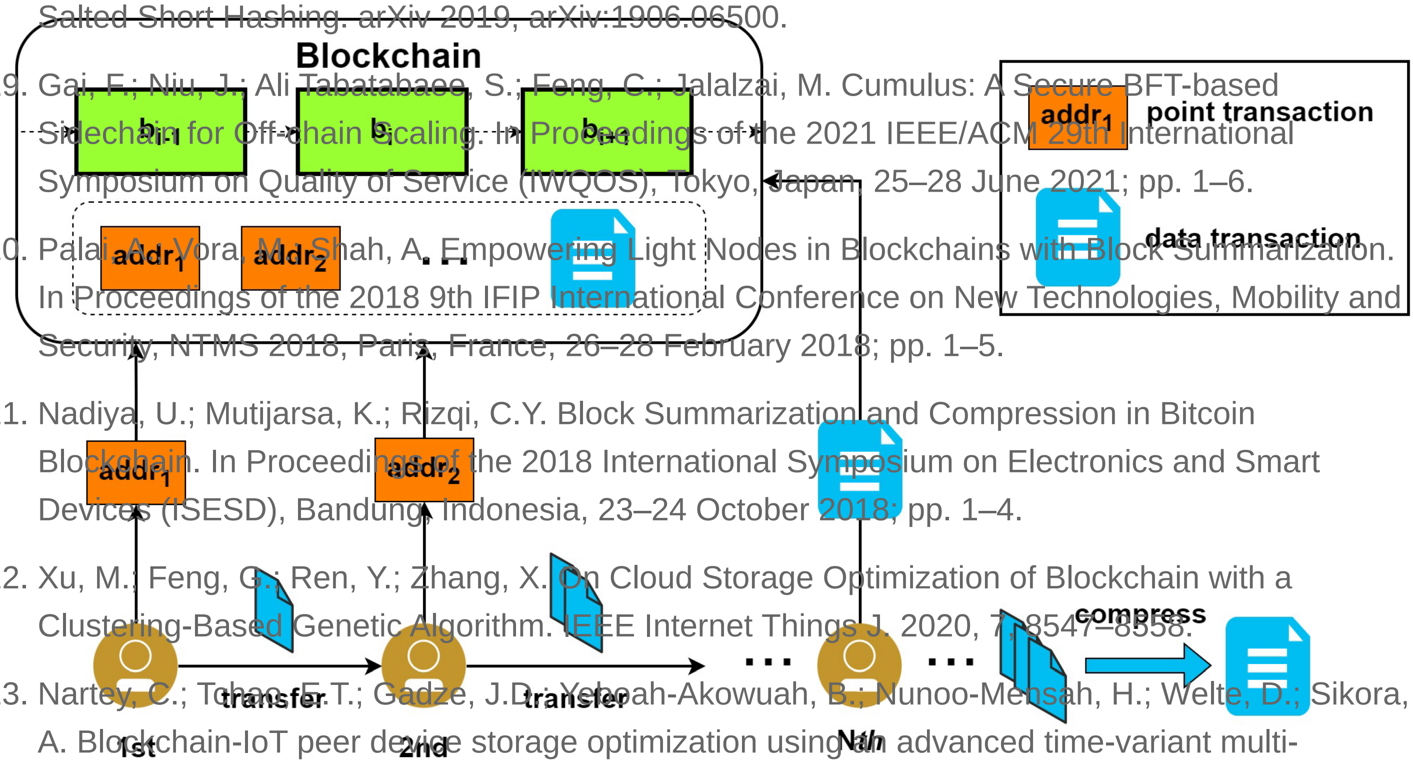

Figure 3. Compressed and private data sharing (Cpds) [12].

2.2. 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 2**.

14. 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.

15. 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.

16. 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.

17. 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.

18. 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.

19. 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.

20. 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.

21. 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.

22. 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.

23. 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.

24. 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.

25. 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.

26. 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.
2. IEEE
2. storage
2. stem. In Proceedings of the IEEE 34th International Conference on Data Engineering, ICDE 2018, Paris, France, 16–19 April 2018; pp. 173–184.
30. 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.
31. 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.

Table 2. Comparison of summarization-based schemes.

Proposed Work	Approach	Algorithm	Storage Reduction	Limitations
Palai et al. [20]	Summarization	Recursive summarization tree	54%	Huge block summary size
Nadiya et al. [21]	Summarization	Recursive summarization tree and deflate compression algorithm	78.1%	Designed for bitcoin blockchain, lack of standard summary block for other blockchains

2.3. 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.

2.3.1. 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 3** shows a comparison of these works.

Table 3. Comparison of off-chain storage scheme optimization works.

Proposed Work	Approach	Algorithm	Storage Reduction	Runtime	Limitations
Xu et al. [22]	Cloud storage	NSGA-C	30%	872.4 s	Long runtime
Nartey et al. [23]	Cloud storage	AT-MOPSO	-	384.2 s	Relatively poor solution for local space occupancy compared to NSGA-C
Zheng et al. [24]	Distributed data storage	IPFS-based storage	91.83%	-	Increased latency due to queries to IPFS network

2.3.2. 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 4** shows a comparison of these works.

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

Proposed Work	Approach	Algorithm	Storage Reduction	Query Efficiency	Latency	Limitations
Dorri et al. [25]	Transaction flexibility	MOF-BC	25%	-	max 6.5 min	High transaction processing time
Pyoung et al. [26]	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. [27]	Partial storage	BFT-Store	86.8%	-	-	Long repair time for decoding, leading to longer processing time
Yu et al. [28]	Partial storage	VBG	-	0.19 s	-	Increased query cost on remote block data
Xu et al. [29]	Partial storage	Consensus Unit	75%–95%	Increased query cost	3% higher than benchmark	High latency on off-node queries
Matzutt et al. [30]	Block pruning	CoinPrune	86.98%	-	-	Limited by UTXO-based design
Wang et al. [31]	Block pruning	ESS	82.14%	-	9.21 s	Limited by UTXO-based design