Internet of Things 26 (2024) 101217

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot

Review article

Latency performance modelling in hyperledger fabric blockchain: Challenges and directions with an IoT perspective

Jummai Enare Abang [a,*], Haifa Takruri [a], Rabab Al-Zaidi [a], Mohammed Al-Khalidi [b]

[a] School of Science, Engineering and Environment, University of Salford, Salford, Manchester, UK
[b] Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, UK

ARTICLE INFO

Keywords:
Blockchain
Hyperledger Fabric
Internet of Things
Latency

ABSTRACT

Blockchain is a decentralized and distributed ledger technology that enables secure and transparent recording of transactions across multiple participants. Hyperledger Fabric (HLF), a permissioned blockchain, enhances performance through its modular design and pluggable consensus. However, integrating HLF with enterprise applications introduces latency challenges. Researchers have proposed numerous latency performance modelling techniques to address this issue. These studies contribute to a deeper understanding of HLF's latency by employing various modelling approaches and exploring techniques to improve network latency. However, existing HLF latency modelling studies lack an analysis of how these research efforts apply to specific use cases. This paper examines existing research on latency performance modelling in HLF and the challenges of applying these models to HLF-enabled Internet of Things (IoT) use cases. We propose a novel set of criteria for evaluating HLF latency performance modelling and highlight key HLF parameters that influence latency, aligning them with our evaluation criteria. We then classify existing papers based on their focus on latency modelling and the criteria they address. Additionally, we provide a comprehensive overview of latency performance modelling from various researchers, emphasizing the challenges in adapting these models to HLF-enabled IoT blockchain within the framework of our evaluation criteria. Finally, we suggest directions for future research and highlight open research questions for further exploration.

1. Introduction

Blockchain is a decentralized and distributed ledger technology that enables the secure and transparent recording of transactions across multiple participants [1–3]. It is designed to be transparent, secure, tamper-resistant and consists of a chain of blocks, each containing a list of transactions [4–6]. Blockchain technology gained prominence with the advent of Bitcoin, the first cryptocurrency, but its potential applications extend far beyond digital currencies [7]. The technology has gained significant importance due to its ability to address several challenges in different sectors [8–11]. From the perspective of IoT, Blockchain provides enhanced security by utilizing cryptographic techniques to ensure the integrity and immutability of data [12,13] transactions between IoT devices, reducing vulnerabilities and unauthorized access. Additionally, it offers transparency and auditability by providing a transparent and traceable record of transactions, facilitating real-time auditing and accountability within IoT networks [12–15]. Blockchain eliminates the need for a central authority, reducing intermediary costs. One of the key features of blockchain is its immutability, which means that once a transaction is recorded, it cannot be altered or tampered with [16–22]. This attribute makes blockchain a reliable and trustworthy system for various IoT applications. Blockchain has the potential to streamline

[*] Corresponding author.
E-mail address: j.o.enareabang@edu.salford.ac.uk (J.E. Abang).

https://doi.org/10.1016/j.iot.2024.101217
Received 28 February 2024; Received in revised form 19 April 2024; Accepted 6 May 2024
Available online 11 May 2024

processes, automate workflows, and create new business models in various IoT industries [23,24]. These attributes make blockchain suitable for several applications, including supply chain management, financial services, and healthcare applications. Blockchain can revolutionize supply chain management by tracking and verifying the movement of goods, enhancing transparency and efficiency. It can also integrate with IoT devices in smart cities to improve urban infrastructure management, such as traffic control and waste management. Moreover, blockchain secures and shares medical data among IoT devices in healthcare applications, ensuring patient privacy and interoperability across healthcare systems. In industrial IoT (IIoT) settings, blockchain enhances security and transparency in industrial processes, optimizing manufacturing operations and predictive maintenance. Additionally, blockchain technology manages the digital identities of IoT devices securely, enabling seamless authentication and trust between devices. This capability can also be applied in agriculture, where blockchain-enabled IoT devices monitor crop conditions, automate farming processes, and track food supply chains for improved efficiency and traceability [17]. In summary, integrating blockchain with IoT adds a layer of security, transparency, and automation to various industries, improving data integrity, operational efficiency, and trust among interconnected devices.

The significance of blockchain has led to a surge in research and development activities. Researchers are exploring ways to improve blockchain systems' scalability, privacy, interoperability, latency and energy efficiency. They are developing new consensus algorithms, optimizing smart contract execution, enhancing privacy features, and exploring new use cases to make blockchain more practical and adaptable to different use cases beyond cryptocurrency [25–27]. Additionally, researchers are exploring integrating blockchain with other emerging technologies, such as artificial intelligence and the Internet of Things, to unlock new possibilities [28–30]. Integrating blockchain and IoT is driven by needs across industries to enhance security, transparency, and efficiency in interconnected environments. There are two primary types of blockchains: permissionless and permissioned, further divided into public, private, and consortium. Public Blockchains, like Bitcoin and Ethereum, are permissionless and open to anyone, allowing anyone to participate in the network and validate transactions. Private Blockchains are permissioned and restricted to a specific entity or organization, where participants are known and trusted, providing higher privacy and control over the network. Consortium Blockchains like HLF are a hybrid model where organizations collaboratively maintain the network with shared control and decision-making, making it suitable for industries requiring shared governance [31].

The Permissioned blockchains have received much attention recently because of their fast transaction processing and privacy preservation [32,33]. Hyperledger blockchain, a consortium open-source collaborative project hosted by the Linux Foundation, aims to advance cross-industry blockchain technologies by providing a modular framework for building enterprise-grade blockchain solutions [34]. Hyperledger incorporates multiple blockchain frameworks, tools, and libraries to support diverse business use cases. Some Hyperledger frameworks include HLF, Hyperledger Sawtooth, Hyperledger Indy, Hyperledger Iroha, and Hyperledger Besu [35]. These frameworks differ in architecture, consensus mechanisms, and features, allowing organizations to choose the most suitable one for their needs. The Hyperledger tools facilitate the development and deployment of blockchain applications. Some notable tools include Hyperledger Composer, which simplifies the creation of smart contracts; Hyperledger Caliper for performance benchmarking; Hyperledger Explorer for blockchain visualization; and Hyperledger Cello for blockchain infrastructure management [35,36].

HLF is one of the most widely adopted frameworks within the Hyperledger umbrella [35]. It is the first open-source distributed system designed for permissioned blockchain deployment [37]. HLF is specifically designed for enterprise use cases and provides features like scalability, privacy, and flexibility [38–41]. Fabric supports a modular architecture that enhances performance and allows organizations to plug in different components according to their requirements. It also offers channel-based privacy, enabling selective sharing of data among network participants [39,42]. HLF is gaining popularity among enterprises due to its suitability for building permissioned and private blockchain networks. It provides a high level of control over access and permissions, making it ideal for industries that require strict privacy and compliance measures [43–46]. HLF has various configurable parameters, including block size, channels, endorsement policy, and state databases, which must be optimally set to obtain the best performance [37]. The consensus mechanism in HLF is deterministic, meaning that the HLF network's process of agreeing on the order and validity of transactions is predictable and follows a specific set of rules. This attribute enables fast consensus among authenticated users, making HLF suitable for enterprise applications with high transaction volumes [47]. However, integrating HLF with enterprise applications presents a significant challenge: latency.

Latency is the time delay that occurs in the execution and validation of transactions within a blockchain network. It impacts the overall performance and user experience of applications running on the network. Transaction latency is a vital metric for blockchain systems as it directly affects task Quality of Service (QoS). Low latency is crucial in applications like smart transportation systems, industries, and E-health services. The challenge lies in minimizing latency to ensure smooth and efficient operations for latency-sensitive enterprise applications [47].

The overall performance of HLF blockchain systems is extensively researched [37,48–52] through several recent studies focusing on analysing the latency of HLF [47,53–55]. These studies, among others, contribute to a deeper understanding of HLF latency by utilizing different modelling approaches. Latency performance modelling research focuses on optimizing one or more HLF parameters to minimize latency. These works explore various techniques to enhance the latency of HLF networks. However, the existing studies on HLF latency modelling lack analysis regarding how these research collectively impact a use case, highlighting the limitations and future research directions. This paper addresses this gap by analysing existing research on latency performance modelling in HLF and its impact on IoT use cases. HLF is significant in IoT because it offers a secure, scalable, and permissioned blockchain framework. Fabric's features, like access control, identity management, and privacy protection, are crucial for maintaining the integrity and security of the network in IoT applications where numerous devices exchange sensitive data. Additionally, its modular architecture allows for customization to fit various IoT use cases, making it an ideal platform for deploying and managing IoT ecosystems. The HLF latency modelling survey is crucial for IoT because IoT systems rely on efficient communication among numerous devices, and optimizing latency in HLF ensures smooth data transmission. Understanding the resource needs of HLF nodes aids in conserving resources, which is crucial given the limitations of IoT devices; this also benefits IoT real-time decision-making by reducing transaction latency in HLF and enhancing system responsiveness. Evaluating HLF scalability helps accommodate IoT network growth without compromising performance. Security is paramount in IoT, and understanding HLF latency ensures secure and transparent transaction processing. The customization of HLF for diverse IoT deployments minimizes latency while meeting specific application needs. Ultimately, the HLF latency survey guides the design and optimization of blockchain solutions for IoT, enhancing efficiency and reliability. This survey is essential for fostering trust and improving security in IoT ecosystems.

This paper aims to offer a comprehensive analysis of latency optimization research in the HLF blockchain network, identify challenges in their application to IoT use cases, and highlight future directions. The main contributions of this paper are summarized as follows:

1. We propose a novel set of evaluation criteria to assess the research conducted in latency performance modelling in the HLF network. We also outline the HLF parameters identified by various researchers to affect latency, showing their relationship to the evaluation criteria.
2. We create novel latency modelling focus areas, and based on these, we categorize the research papers and outline the evaluation criteria each paper in each category satisfies.
3. We present a summary of the latency performance modelling outlined by various researchers. Under the evaluation criteria umbrella, we identify challenges in their application to HLF-enabled IoT blockchain and provide possible future research direction.

   The subsequent sections of the paper are organized as follows: Section 2 explains the HLF architecture, transaction flow and latency in HLF. Section 3 presents the evaluation criteria and HLF latency performance modelling parameters. The taxonomy of current research on latency performance modelling in HLF is presented in Section 4. Section 5 summarizes research on latency performance modelling and challenges in its application to HLF-enabled IoT use case, while Section 6 presents the future research directions and open research question. Finally, we conclude the paper in Section 7.

2. HLF architecture, transaction flow and latency

HLF is attributed to its modularity, flexibility, and focus on meeting the needs of enterprise blockchain applications. These characteristics are driven by the HLF architecture that underpins its transaction flow and latency. Understanding HLF architecture, transaction flow, and latency is crucial for integrating IoT networks with this blockchain framework. This knowledge helps design efficient systems that accommodate diverse IoT devices and handle large volumes of data generated by IoT networks. Optimizing transaction flow ensures the timely processing of IoT data and efficient execution of smart contracts, which is essential for real-time IoT applications. Managing latency within HLF ensures prompt data processing and minimizes resource usage on IoT devices with limited capabilities. Exploring these aspects allows for scalable and flexible solutions that can adapt to the evolving needs of IoT applications, ensuring optimal performance and reliability in HLF-enabled IoT networks. This section discusses these main pillars of HLF in detail.

2.1. HLF architecture

HLF is a sophisticated enterprise blockchain framework where various components collaborate seamlessly to create a secure and scalable network. The architecture is designed to address the requirements of scalability, privacy, and permissioned access control. The primary components of the architecture include peer nodes, ordering nodes and client applications. Fig. 1 illustrates the layout of different components in the HLF blockchain network and their interconnections. This network consists of a single channel and two organizations (ORG A and B), each having two peers, a client application, and a Fabric Certificate Authority (CA). Each CA is linked with a distinct Membership Service Provider (MSP) associated with its respective organization. The Membership Service Provider (MSP) manages identities and authentication, while the Certificate Authority (CA) issues digital certificates for secure communication. Peer nodes execute smart contracts (chaincode), maintain ledgers, and validate transactions, with the Ordering Service arranging transactions into blocks and enforcing transaction orders. The Consensus Protocol ensures agreement on transaction validity among network nodes. Channels enable data privacy and confidentiality, restricting visibility to specific participants. Chaincode defines business logic for transactions and interacts with the ledger, while the ledger maintains an immutable record of all transactions and asset states. State databases store the current state of assets managed by chaincode, allowing for efficient querying and access. This cohesive architecture enables Hyperledger Fabric to support diverse enterprise blockchain applications efficiently and securely.

2.1.1. Organization

The HLF network comprises multiple organizations that are collaborating to form a consortium. Each organization represents an entity that participates in the network. Organizations have control over their membership and can define their policies. The following are the different members of an organization [56]:

Peer Nodes: Peer nodes maintain a copy of the shared ledger and execute smart contracts (chaincode). There are two types of peer nodes: endorsing and committing peers. Endorsing peers simulate and endorse transactions by executing the chaincode

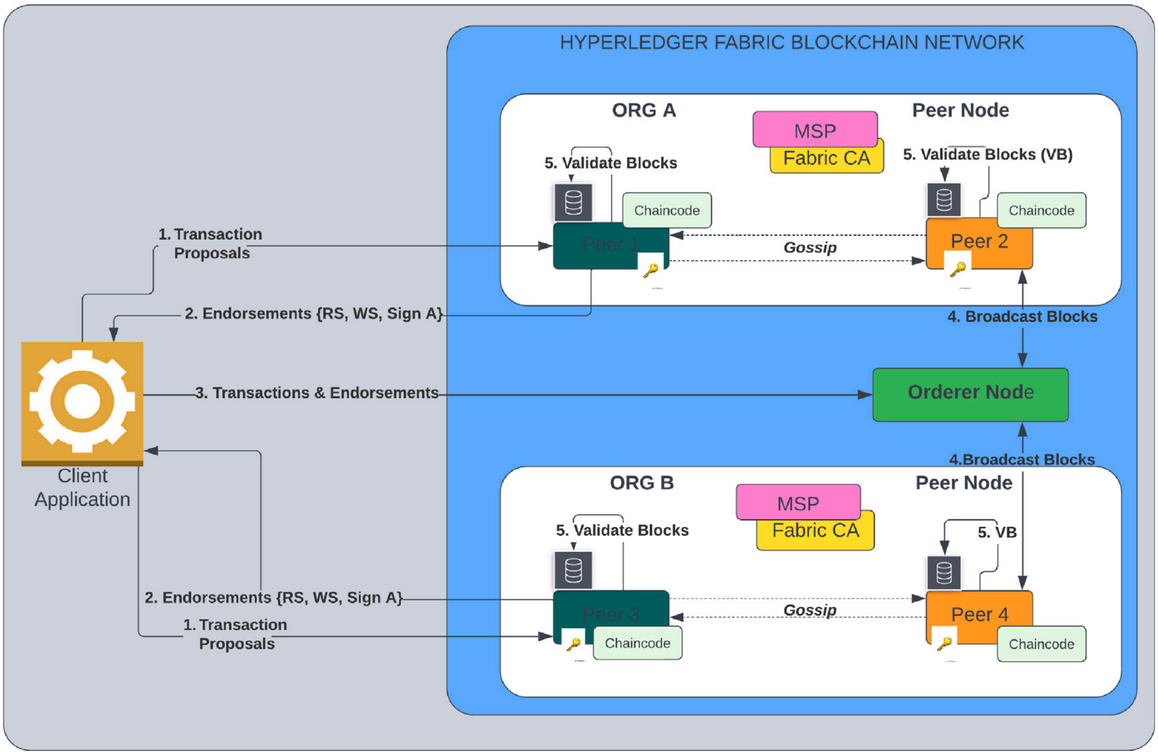

Fig. 1. Visualizing HLF components and transaction flow within the HLF blockchain network.

and generating transaction endorsements [57]. Committing peers validate endorsed transactions and commit them to the shared ledger [58]. In HLF, endorsing peers serves dual roles as endorsing and committing peers [56].

Ledger: The ledger in HLF maintains a tamper-resistant record of all transactions. There are two types of ledgers: the **world state** and the **transaction log** (blockchain) [57]. The world state represents the current state of the network, represented as a key–value pair, allowing access to an object's value without searching the entire blockchain. The transaction log contains a history of all transactions. It records all the changes that result in the current value in the world state and saves them in blocks joined to the block on top of the other to form a chain [58]. HLF supports two types of state databases, namely **CouchDB** and **GoLevelDB** [59].

Chaincode: Chaincode, also known as smart contracts, encapsulates the business logic of the network. Chaincode is written in programming languages like Go, JavaScript, or Java. It defines the rules for validating and modifying the ledger state [5,60].

Certificate Authority (CA): The Certificate Authority generates digital certificates for network nodes. These certificates contain comprehensive information about each node and function as their identity within the network. Each organization's CA administrator is responsible for generating certificates for the organization's peers [56].

Membership Service Provider (MSP): The MSP manages identities and permissions within the network. It ensures that participants are authenticated and authorized, and predefined policies govern their access to the network. MSP provides a hierarchical structure for managing certificates and identity validation [61]. The default MSP implementation in Fabric supports commercial Certification Authority (CA) and standard PKI techniques for authentication based on digital signatures. Alternatively, Fabric provides Fabric-CA, a standalone certification authority [62].

2.1.2. Client application

A client is an end-user who does not store any blockchain data. Instead, the client interacts with the network through an application. The client sends query or update requests to multiple peers within the same network channel using this application. The client application is not part of the HLF network [56].

2.1.3. Ordering service

The ordering service is responsible for receiving endorsed transactions from endorsing peers, ordering them into a consistent sequence, and packaging them into blocks. The nodes that assume this responsibility are called orderer or ordering nodes. Like other network nodes, ordering nodes belong to a specific organization and receive their identities through the same process as the peer nodes [56]. HLF supports pluggable consensus mechanisms. It allows participants to choose their preferred consensus algorithm based on their specific requirements. Some algorithms include Practical Byzantine Fault Tolerance (PBFT), Raft and Kafka. The ordering service is implemented using these consensus algorithms [56]. The ordering service ensures the total order of transactions in the network and delivers blocks to committing peers for validation and inclusion in the ledger [5,60].

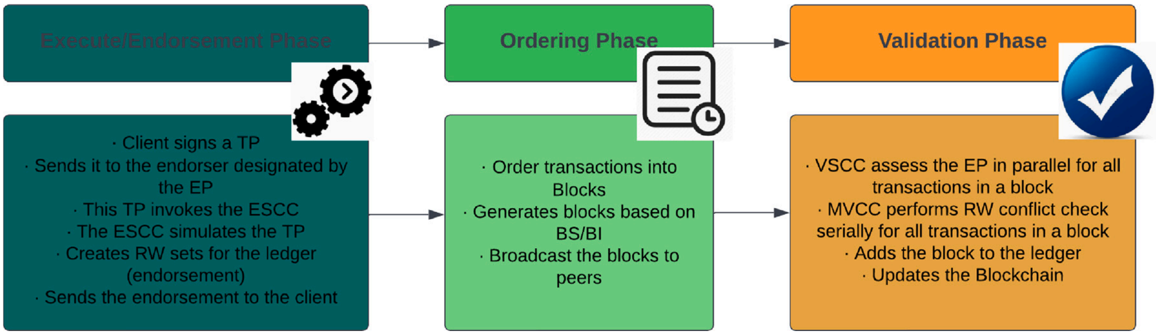
Execute/Endorsement Phase
· Client signs a TP
· Sends it to the endorser designated by the EP
· This TP invokes the ESCC
· The ESCC simulates the TP
· Creates RW sets for the ledger (endorsement)
· Sends the endorsement to the client
Ordering Phase
· Order transactions into Blocks
· Generates blocks based on BS/BI
· Broadcast the blocks to peers
Validation Phase
· VSCC assess the EP in parallel for all transactions in a block
· MVCC performs RW conflict check serially for all transactions in a block
· Adds the block to the ledger
· Updates the Blockchain

Fig. 2. Summary of Processes in the transaction flow of HLF.

2.1.4. Channels

Channels in HLF allow participants to create private sub-networks within the overall network. Each channel has its ledger and smart contracts, providing privacy and confidentiality for the participants involved. Transactions within a channel are only visible to the channel members. The architecture promotes scalability, high performance, and confidentiality, making HLF a popular choice for building private and permissioned blockchain networks. Fabric nodes communicate via the gRPC framework [5,63].

2.2. Transaction flow

In HLF, the transaction flow architecture follows an execute-order-validate approach, which consists of three essential phases: **Execution/Endorsement Phase, Ordering Phase** and **Validation/Commitment Phase** [64]. A summary of the processes in each phase of the transaction flow is depicted in Fig. 2. Through the transaction flow, HLF ensures a robust and secure environment for enterprise blockchain applications.

Execution Phase: In the Execution phase, a client signs a transaction proposal (TP) with its credentials and sends it to the designated endorser for endorsement (point 1 in Fig. 1). The transaction proposal invokes the Endorsement System Chaincode (ESCC) in the endorser, which simulates the transaction proposal and creates read/write (RW) sets for the ledger, a process called endorsement [65]. The read set comprises the keys accessed along with their corresponding version numbers, whereas the write set includes the keys that need to be updated along with their new values [66]. The endorsement is sent back to the client (point 2 in Fig. 1), who collects enough transactions and endorsements based on the Endorsement Policy (EP) and submits them to the orderer. The endorsement phase is significant because it allows for distributed validation, preventing malicious or unauthorized transactions from entering the network. The client application ensures that the transactions and endorsements collected from various endorsers are consistent; otherwise, they will be deemed invalid. It also enables parallelism and scalability, as multiple endorsing peers can execute transactions simultaneously.

Ordering Phase: In the Ordering Phase, the orderer uses a pluggable consensus protocol to create a series of endorsed transactions in blocks. The blocks are cut by the block size or generation time limit and broadcasted to peers directly or indirectly through gossip using epidemic multicast (points 3 & 4 in Fig. 1). The ordering phase is vital for achieving consensus on the transaction order across all network participants. It guarantees the integrity and consistency of the blockchain by employing consensus algorithms.

Validation Phase: In the Validation phase, the endorsement policy is assessed in parallel for all transactions in the block. The Validation System Chaincode (VSCC) evaluates the endorsement based on the chaincode's policy, and Multi-Version Concurrency Control (MVCC) performs a check for read–write conflicts on each transaction in the block. It ensures that the versions of the keys in the read set field match those in the locally kept current state of the ledger for each transaction. This check ensures that valid transactions do not have read–write conflicts, preventing double-spending [66]. Finally, the block is committed to the locally stored ledger, and the blockchain is updated [5,63] (point 5 in Fig. 1). The validation/commitment phase is crucial for maintaining the integrity of the blockchain by verifying that the endorsed transactions are valid and consistent with the network's rules. It also ensures that the state of the blockchain accurately reflects the executed transactions.

2.3. Latency in HLF

Latency in HLF refers to the time to commit a transaction to the ledger. A high-level view of the transaction flow in HLF consists of three latency phases: **Execution/Endorsement Phase Latency, Ordering Phase Latency**, and **Validation/Commit Phase Latency**. These latency phases are crucial to understanding the overall transaction processing time and optimizing the network's performance.

Execution/Endorsement Phase Latency refers to the time it takes the endorsing peers to validate and simulate the transaction against the smart contracts or chaincode. This is depicted in points 1 and 2 in Fig. 1 above. This latency is influenced by factors such as the complexity of the endorsement policy, the number of endorsing peers, the resources available to execute the transaction and the complexity of the chaincode.

Ordering Phase Latency refers to the time it takes the ordering service to aggregate transactions into blocks and establish the order in which they will be included in the blockchain. This is shown in points 3 and 4 of Fig. 1. This latency is influenced by factors such as the number of transactions to be ordered, the block size, Block timeout and the type of consensus algorithm utilized.

Validation/Commit Phase Latency refers to the time the validating peers take to perform the RW conflict and validation checks and commit the transactions to the ledger. Point 5 in Fig. 1 above represents this latency. This latency is influenced by factors such as the complexity of the validation rules, the complexity of the endorsement policy, the number of validating peers, and the resources available to execute the validation process [32].

Lower latency contributes to higher network scalability, allowing the blockchain network to handle more transactions within a given time frame. Hence, minimizing latency in each phase of the transaction flow is essential for maximizing the performance and efficiency of the HLF network.

3. The evaluation criteria and HLF latency performance modelling parameters

3.1. The evaluation criteria

The evaluation criteria used in this study to analyse latency in HLF are essential for assessing the performance and effectiveness of different research approaches. The authors propose the criteria in 1 to compare and evaluate the methodologies and tools employed in HLF latency studies. These novel evaluation criteria provide a comprehensive framework for assessing the strengths and limitations of different latency modelling approaches in HLF. From an IoT perspective, these evaluation criteria are essential for assessing the effectiveness and efficiency of latency models in IoT systems. HLF blockchain must scale efficiently to handle the high volume of IoT transactions and ensure timely data processing. It must minimize latency to support real-time or near-real-time communication in IoT applications. Resource efficiency is crucial, considering the limited computational resources of IoT devices. Interoperability is necessary for seamless integration with diverse IoT technologies. Flexibility in architecture and smart contract functionalities allows adaptation to various IoT scenarios. Scalable data storage is required to manage the large volume of historical IoT data. Integrating IoT protocols facilitates data exchange between blockchain networks and IoT devices, ensuring effective support for IoT applications. By considering these criteria, researchers can make informed decisions and identify areas for improvement in future studies.

The proposed set of evaluation criteria for assessing latency performance represents a notable contribution compared to existing approaches. It offers a comprehensive coverage of latency factors. This comprehensive coverage provides a holistic view of latency performance, considering multiple aspects contributing to overall system responsiveness. Unlike existing works that may focus on general blockchain performance metrics, the proposed criteria are tailored to the specific requirements and constraints of IoT applications. This relevance ensures that the evaluation criteria address IoT environments' unique challenges. The proposed criteria offer specific and well-defined metrics for measuring latency. This specificity enhances clarity and facilitates precise evaluation of latency performance in different contexts. The evaluation criteria are designed to be practical and applicable in real-world scenarios, considering the practical implications of latency on IoT device interactions and system responsiveness. This practicality enables stakeholders to assess and optimize latency performance for IoT deployments effectively. This study establishes common benchmarks for measuring and comparing latency performance across blockchain platforms and implementations by proposing standardized evaluation criteria. Standardization facilitates meaningful comparisons and promotes best practices in latency optimization for blockchain-enabled IoT networks. In summary, the proposed evaluation criteria's main differences and significance compared to existing works lie in their comprehensive coverage, specificity, relevance to IoT applications, practical applicability, and potential for standardization. These aspects collectively enhance the effectiveness and utility of the evaluation criteria in assessing and improving latency performance in blockchain-based IoT systems.

3.2. HLF parameters used to measure latency and how they relate to the evaluation criteria

This subsection highlights key parameters influencing the performance of HLF systems. The parameters shown in Table 2 are crucial in measuring and evaluating the latency of HLF networks. **TAR/TSR** signifies the pace at which transactions arrive or are sent into the network. **BS** denotes the maximum number of transactions a block can hold. **BI** is the time interval between successive block creation. **NOT** encompasses the number of transactions a client application presents for execution on the shared ledger. **RU** measures the effective utilization of computational resources across nodes within the HLF network to perform various operations. **CEP** is the intricacy or sophistication of policy conditions, including the combination of logical operators such as AND, OR, and NOT. **TC** characterizes the depth of intricacy and resource demands linked to executing a specific transaction type, be it a read, write or a combination of transaction types. **CM** references the protocol or mechanism employed to achieve consensus among multiple nodes regarding transaction validity and order. **CDB** signifies the fundamental database technology employed to store HLF's asset state and other pertinent data. **C/CPL** encompasses the chaincode delineating the business logic governing network interactions and the programming languages utilized for scripting the chaincode. **NOC** designates the count of private communication pathways or subnetworks within a larger blockchain network. **NOO** indicates the count of entities in the blockchain network as distinct units, each with its identity, assets, and roles. **NOP** represents the number of nodes actively participating in the blockchain network, retaining ledger copies, and executing chaincode (smart contracts) to process transactions.

Table 1
The evaluation criteria.

Evaluation criteria	Meaning and relationship to latency in the context of IoT
Scalability (SC)	This criterion evaluates how well the latency model can handle increasing transaction volumes and network sizes without significant degradation in performance. In IoT systems, scalability is crucial as the number of connected devices and data transactions can grow rapidly. Low latency ensures quick processing of these transactions, allowing the system to handle increasing workloads without delays, thereby maintaining scalability.
Flexibility (FL)	Flexibility assesses the adaptability of the latency model to different IoT configurations and scenarios. IoT environments can vary significantly in terms of device types, data formats, and communication protocols. Low latency enables rapid adaptation to these diverse requirements, ensuring timely responses to changing conditions and varied data formats, thus enhancing flexibility.
Complexity (CO)	Complexity measures the intricacy involved in implementing and utilizing the latency model. Complex processes introduce latency in IoT systems. Evaluating latency in terms of complexity involves simplifying deployment and integration processes to minimize transaction confirmation times, contributing to streamlined execution and reduced latency in data transmission.
Computational Efficiency (CE)	CE evaluates the computational resources and time required to execute the latency model. In IoT systems, efficient resource utilization is critical for minimizing latency. Evaluating latency in terms of computational efficiency involves optimizing the utilization of computing resources to achieve low latency in data processing and consensus algorithms, thereby improving overall system performance.
Reproducibility (RE)	RE assesses the ability to replicate results obtained from the latency model. Consistent and reliable latency performance is essential in IoT systems to ensure predictable behaviour across various conditions. Evaluating latency in terms of reproducibility involves testing and analysing the system's behaviour to ensure consistent and reliable performance, regardless of the environment or scenario.
Validation (VA)	VA examines the extent to which the latency model has been validated against real-world measurements or empirical data. Validating the latency model against real-world data is crucial for ensuring its suitability for IoT applications and identifying potential bottlenecks. Latency metrics serve as a basis for validation to objectively measure and verify the system's performance, ensuring it meets defined criteria and requirements in IoT environments.

Table 2
Key HLF latency-related parameters and their alignment with the evaluation criteria.

Key HLF latency-related parameters	Evaluation criteria					
	SC	FL	CO	CE	RE	VA
Transaction Arrival Rate/Transaction Sending Rate (TAR/TSR)	✓	✓	✓	✓	✓	✓
Block Size (BS)	✓			✓		
Block Interval (BI)	✓			✓		
Number of Transactions(NOT)	✓			✓		
Resource Utilization (RU)	✓			✓	✓	
Complexity of the Endorsement Policy (CEP)	✓	✓	✓	✓	✓	✓
Transaction Complexity (TC)	✓	✓		✓	✓	✓
Consensus Mechanism (CM)	✓	✓	✓	✓	✓	✓
Choice of Database (CDB)	✓	✓		✓	✓	✓
Chaincode/Chaincode Programming Language (C/CPL)		✓	✓		✓	✓
Number of Channels (NOC)	✓			✓		
Number of Organizations (NOO)	✓		✓	✓		
Number of Peers (NOP)	✓		✓	✓		

4. Taxonomy of current research on latency performance modelling in HLF

This section categorizes recent studies regarding latency performance modelling in HLF. We utilized the SPIDER research methodology, which stands for Sample population, Phenomenon of interest, Design of study, Evaluation, and Research type [67].

Sample population: This research sample consists of papers on latency performance modelling in HLF. Specifically, the sample includes papers retrieved from academic databases such as Google Scholar, IEEE Xplore, Elsevier, MDPI, ACM, and Scopus library. We found 69 related research papers published between 2017 and 2023. The sample is narrowed down to 35 papers selected based on quality assessment factors explained in the evaluation category of SPIDER.

Phenomenon of interest: The phenomenon of interest is latency optimization in HLF. It involves a comprehensive search of existing literature on latency optimization in HLF, followed by the selection, appraisal, and synthesis of relevant studies. The aim is to provide a thorough and unbiased summary of the existing research. The research objectives guided the development of the targeted search strategy, utilizing relevant keywords and phrases in academic databases, conference proceedings, and reputable journals focused on latency in HLF.

Design of study: We define the inclusion criteria, emphasizing papers that propose or evaluate latency optimization techniques in HLF, published in credible conferences or journals related to HLF blockchain technology. We excluded papers that do not specifically address latency optimization in HLF, followed by a further selection process with an initial screening of titles and abstracts to identify

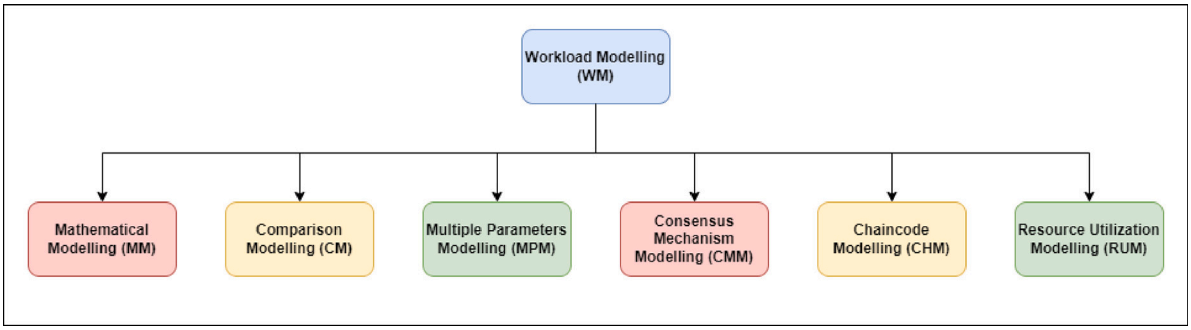
Workload Modelling (WM)
Mathematical Modelling (MM)
Comparison Modelling (CM)
Multiple Parameters Modelling (MPM)
Consensus Mechanism Modelling (CMM)
Chaincode Modelling (CHM)
Resource Utilization Modelling (RUM)

Fig. 3. Categories of workload modelling.

potentially relevant papers, followed by a thorough review of the full texts of selected papers. Information extraction from these papers focused on capturing details specific to HLF, particularly the proposed latency optimization techniques and key findings.

Evaluation: Evaluation encompasses the quality assessment of the selected papers, focusing on factors such as the rigor of experimental design, methodological appropriateness, and significance of results. The synthesized information from the selected papers is used to provide a comprehensive overview of the current state of latency optimization in HLF.

Research type: We categorized the latency focus into six distinct areas of workload modelling based on the modelling technique used by the study and outlined the corresponding evaluation criteria satisfied by papers within each category. Fig. 2 shows the categories of the workload modelling. The categorization organizes and classifies different aspects of workload modelling based on the specific modelling techniques employed in the study. This categorization helps present a structured and comprehensive overview of how various aspects of workload are being modelled and analysed in the context of latency. Breaking down workload modelling into these areas allows a more detailed examination of different factors influencing latency. Each category represents a unique angle or method applied to understand and analyse workload in the context of latency.

The specific choice of six categories for workload modelling is influenced by the complexity and diversity of factors involved in understanding and analysing latency. This choice reflects the study's multidimensional nature of workload modelling, where different aspects must be considered to understand holistically the factors influencing latency in the given context. Each category provides a unique lens through which researchers can examine and analyse specific aspects of workload and its impact on system performance.

4.1. Workload modelling (WM)

Various factors influence the performance of the HLF blockchain system, and optimizing one metric may affect others. Designing a practical HLF blockchain system is currently a major research challenge. Workload modelling has emerged as a crucial approach to address this challenge. Workload modelling involves testing different parameters of the HLF network, explicitly focusing on critical metrics such as latency. This modelling provides a standardized platform for optimizing performance. The effectiveness of a system can be better evaluated by testing a higher workload in benchmarking [32]. We classify workload modelling into six categories based on their latency modelling area of focus and outline the evaluation criteria each paper in each category satisfies. The modelling categories are interconnected and work together to optimize system performance. Mathematical Modelling establishes theoretical frameworks for performance analysis, offering precise quantitative insights into how parameters affect system behaviour. Comparison Modelling aids in making strategic decisions by comparing Hyperledger Fabric with other systems or platforms identifying strengths and weaknesses to guide improvements. Multiple Parameters Modelling provides a comprehensive view of system interactions and dependencies by considering the combined impact of various parameters on latency and performance. Consensus Mechanism Modelling focuses on ensuring efficient data consistency and transaction processing within decentralized systems using consensus protocols. Chaincode Modelling targets smart contract performance optimization, improving transaction execution efficiency within the Fabric platform. Resource Utilization Modelling analyzes resource usage under different scenarios, guiding operational optimizations to maximize system efficiency and performance. Together, these modelling approaches contribute to a holistic understanding and enhancement of HLF's performance and efficiency, addressing various aspects of latency to optimize blockchain implementations for practical applications. Each category is crucial in improving overall system performance and operational effectiveness within the HLF ecosystem. Most reviewed papers do not fully address the impact of all the parameters in each evaluation criterion. Where that be the case, we score the paper "Partially meets the criteria". Where all the parameters are addressed, we scored "Meets the criteria"; where no parameter is addressed, we scored "Does not meet the criteria".

1. *Mathematical Modelling (MM):* This category involves using mathematical techniques to model and analyse the impact of various parameters on latency. Mathematical models provide precise and quantitative insights into how parameter changes affect system performance. Several theoretical models, including the Stochastic Reward Net (SRN) model [53,54] Generalized

Table 3
Mathematical modelling evaluation criteria.

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
[54]	Performance Modelling of Hyperledger Fabric (Permissioned Blockchain Network)	Harish Sukhwani et al. used a stochastic modelling approach to capture the performance behaviour of HLF	2018	●	●	●	●	●	●
[68]	Performance modelling and analysis of a Hyperledger-based system using GSPN	Pu Yuan et al. proposed a model using Generalized Stochastic Petri Nets (GSPN) to analyse the performance of a HLF -based system.	2020	◒	●	●	○	●	●
[32]	Facing to Latency of Hyperledger Fabric for Blockchain-enabled IoT: Modelling and Analysis	Sungho Lee et al. presented a latency model for HLF-enabled IoT using probability distributions, specifically the Gamma distribution	2022	◒	○	●	○	●	●

● Meets the criteria. ◒ Partially meets the criteria. ○ Does not meet the criteria.

Table 4
Comparison modelling evaluation criteria.

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
[69]	Performance Comparison of IoT Based Metering System with Different Blockchain Platforms	Chathura Edirimanna et al. evaluated the performance of Ethereum and HLF blockchain platforms for an electricity billing scenario.	2020	○	○	◒	○	◒	◒
[50]	Performance Analysis of Blockchain Platforms:Empirical Evaluation of Hyperledger Fabric and Ethereum	Mohammad Dabbagh et al. evaluated the performance of two prominent blockchain platforms HLF and Ethereum using a performance benchmarking tool called Hyperledger Caliper.	2020	◒	◒	◒	●	●	◒
[70]	Performance Analysis of Private Blockchain Platforms in Varying Workloads	Suporn Pongnumkul et al. provided a detailed methodology and experimental results to analyse the performance of HLF and Ethereum in varying workloads.	2017	●	●	●	●	●	◒
[71]	Performance Analysis of Hyperledger Fabric Platforms	Qassim Nasir et al. evaluated the performance of two versions of HLF (v0.6 and v1.0) regarding execution time, latency, and throughput.	2018	◒	○	●	●	●	◒

● Meets the criteria. ◒ Partially meets the criteria. ○ Does not meet the criteria.

Stochastic Petri Nets (GSPNs) [68], a hierarchical model based on transaction execution and validation [71], and a queueing-based model [32,47], are utilized to define and measure latency. In modelling the performance of Fabric, authors employ Stochastic Petri Networks modelling to provide a simple definition and automated generation/solution of the underlying stochastic process that captures the network system's performance characteristics. SRNs enable the investigation of different scenarios by quickly adding or removing system details [54]. GSPN offers a graphical approach to divide the request processing flow of the HLF-based system into various stages. This is then simulated to derive performance metrics such as latency for each phase, even under unstable conditions [68]. Latencies at each transaction phase are captured, and the total latency is modelled as a Gamma distribution [32]. At the time of this research, only three papers modelled latency in HLF using mathematical models. As shown in Table 3, the primary limitations in this category are scalability and computational efficiency. These limitations arise from factors such as the large size of the stochastic model, the inability to generate the underlying state-space, implementation constraints, focusing only on a specific transaction type, and deploying HLF on a single physical machine and Local Area Network (LAN). These factors collectively affect the ability to scale and achieve optimal computational efficiency.

2. *Comparison Modelling (CM):* Comparison modelling allows for a better understanding of various systems' potential applications and performance analysis across different platforms, particularly in latency [69]. Despite the evident advantages of blockchain technology, technical challenges still need to be addressed before widespread adoption can occur. One of the major barriers to blockchain adoption is the performance aspect, as it needs to provide a superior and practical alternative to centralized solutions currently in use [50,70].

   A comprehensive study of blockchain literature reveals that performance is a significant issue that the academic community has yet to tackle fully [50,72]. Comparing different models or approaches is essential for identifying best practices or understanding the trade-offs between different methods. This category involves benchmarking different workload models to assess their relative performance. Consequently, evaluating the performance of different blockchain platforms and versions can provide practitioners with valuable insights into how their integration into real-world applications is affected, enabling a better understanding of limitations and assisting in the selection of the most suitable platform for their specific needs [50,70,71]. From the information provided in Table 4, the main limitations in this category include scalability, flexibility, and validation. The lack of consideration for different consensus protocols, the potential effects of multiple orderers on network performance, and the impact of node scalability on each platform's performance contribute to this limitation. Additionally, there was no comparison of the maximum number of transactions achievable during the blockchain platform's peak performance. Furthermore, focusing on a specific phase in the transaction flow negatively affects scalability, flexibility, and validation.
3. *Multiple Parameters Modelling (MPM):* This category considers the combined impact of multiple parameters on latency. The HLF systems often have multiple interacting components, and modelling these interactions can be complex but necessary for a comprehensive understanding of latency. Conducting a performance analysis to determine the applicability and usefulness of HLF-based systems is essential [73]. Factors such as block size, number of peers, and hardware limitations, to mention but a few, influence the performance of the Fabric network; some authors focused mainly on evaluating the different configurable network components affecting performance [74] to optimize the Fabric network's performance [75]. From the information presented in 5, the primary limitations in this category involve complexity, computational efficiency, and validation. The benchmark experiments were conducted and implemented on a single-host virtual machine and local area network (LAN), which differs from a distributed production environment. Inefficiencies in utilizing multiple virtual CPUs contribute to system bottlenecks. Additionally, comparing the proposed work with similar studies is a key aspect of validation that is lacking in this category.
4. *Consensus Mechanism Modelling (CMM):* Consensus methods are crucial in decentralized systems, ensuring data consistency across multiple nodes. The effectiveness of the consensus algorithm is essential, particularly for IoT devices with limited resources [76]. Modelling the consensus process can help understand how it influences latency. Consensus modelling involves evaluating the performance of different consensus mechanisms or protocols, specifically analysing their impact on transaction latency using the HLF network workload [77]. Byzantine Fault Tolerance (BFT) [76] and Practical BFT [76] are among the commonly used consensus mechanisms in consortium blockchains [76,78], while Raft and Kafka are the existing consensus mechanisms in use. From the information in Table 6, the main challenges in this modelling category are scalability, flexibility, and validation. The scalability of the Raft consensus algorithm was not thoroughly investigated on larger networks or with varying network sizes, except for batch sizes and sending rates. The discussion on how Practical Byzantine Fault Tolerance (PBFT) addresses scalability is unclear. An ideal algorithm for IoT should possess the flexibility to be adaptable to different IoT scenarios or network configurations. However, scaling up the number of peers and analysing system performance in larger-scale IoT scenarios pose logistical and resource challenges. The reliance on specific assumptions and conditions limits the flexibility to explore alternative consensus mechanisms or blockchain platforms. Limiting the validation scope by excluding consensus protocols and analysing only current platform versions contributes to the limitation in this category.
5. *Chaincode Modelling (CHM):* HLF supports smart contracts called chaincodes [37], which can be written in popular programming languages like Go, Java, and Node.js. In the Fabric platform, a chaincode represents the software responsible for managing and updating assets in the ledger [79]. Chaincode modelling focuses on analysing the impact of workload on transaction latency, with a specific emphasis on the chaincode component. From the information in Table 7, the main limitation in this category is scalability. Evaluating all parameters that impact the scalability of a HLF network is crucial. The scalability analysis is limited to the number of nodes and does not account for additional scalability factors like network load or transaction volume. While there is mention of increased transaction processing speed and network scalability, more comprehensive information about scalability aspects, such as the number of participants, resource requirements, and other parameters, is required.
6. *Resource Utilisation Modelling (RUM):* The measurements conducted for resource usage involved analysing CPU processing power, memory utilization, and network usage. These measurements are performed under different load scenarios. The process of storing various data volumes in the network is carried out with various transactions to determine resource usage [80]. The evaluation of the RUM is presented in Table 8, highlighting scalability and validation as notable constraints in RUM. The evaluation of QiOi lacks detailed insights into its scalability limits because it uses different ordering services and chaincodes but does not cover a wide range of scenarios or transaction combinations. Furthermore, it does not compare QiOi with other existing approaches or benchmarks, which limits the comprehensive assessment of its performance and suggests the need for comparative evaluations with alternative techniques to strengthen validation.

Table 5
Multiple parameter modelling evaluation criteria.

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
[74]	Optimal blockchain network construction methodology based on analysis of configurable components for enhancing Hyperledger Fabric performance.	Lei Hang et al. provide a detailed description of the proposed blockchain network construction methodology and the configurable components that can impact the performance of HLF.	2021	●	●	◒	◒	●	●
[81]	Performance Characterization and Bottleneck Analysis of Hyperledger Fabric	Canhui Wang et al. conducted a performance evaluation on the first long term support release of HLF, specifically focusing on its execute, order and validate phases.	2020	●	○	●	◒	●	○
[82]	Performance Analysis of a Hyperledger Fabric Blockchain Framework: Throughput, Latency and Scalability	Murat Kuzlu et al. provided detailed information about the performance analysis of a specific HLF blockchain implementation.	2019	●	○	○	○	●	◒
[83]	Performance Characterization of Hyperledger Fabric	Arati Baliga et al. examined the performance and scalability features of production release of HLF (v1.0)	2018	●	●	●	●	●	●
[59]	Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform	Parth et al. conducted an empirical study to evaluate the performance of HLF and identify potential bottlenecks.	2018	●	●	●	●	●	●
[84]	Research of the Modular Operational Performance Analysis for Consortium Blockchain	Yean-Fu Wen et al. focused on analysing the performance of a consortium blockchain system and developing modular deployment strategies to achieve the desired transaction throughput.	2021	●	●	●	●	●	●
[56]	Impact of Block Data Components on the Performance of Blockchain Based VANET Implemented on Hyperledger Fabric	Priyanka Gaba et al. provided a detailed exploration of HLF Private Blockchain Network and its components, as well as implementing a Vehicular Ad-hoc Network (VANET) case study.	2022	●	●	●	●	●	●
[37]	Experimental Performance Analysis of a Scalable Distributed Hyperledger Fabric for a Large-Scale IoT Testbed	Houshyar Honar Pajooh et al. focused on evaluating HLF performance in large-scale IoT application	2022	●	●	●	●	●	◒
[85]	Performance Optimization for Blockchain-Enabled Industrial Internet of Things systems: Deep Reinforcement Learning Approach	Mengting Liu et al. proposed a Deep Reinforcement Learning (DRL)-based performance optimization framework for blockchain-enabled IIoT systems.	2019	●	●	●	◒	◒	◒
[47]	Latency performance modelling and analysis for hyperledger fabric blockchain network	Xiaoqiong et al. proposed a theoretical model to calculate the transaction latency in HLF based on various network configurations.	2021	●	●	●	○	●	◒
[86]	Adaptive Blocksize for IoT Payload Data on Fabric Blockchain	Chuan-Ming Liu et al. proposed a blockchain network for IoT data access.	2021	◒	●	○	◒	◒	◒
[87]	Mitigating Conflicting Transactions in Hyperledger Fabric-Permissioned Blockchain for Delay-Sensitive IoT Applications	Xiaoqiong et al. proposed a blockchain system called CATP-Fabric to address the issue of conflicting transactions in HLF	2021	○	◒	◒	◒	●	◒
[88]	Hyperledger Fabric	Takuya Nakaike et al. provided a detailed	2020	●	●	●	●	●	●

(*continued on next page*)

Table 5 (*continued*).

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
	Performance Characterization and Optimization Using GoLevelDB Benchmark	performance characterization of HLF using the GoLevelDB benchmark.							
[79]	Performance modelling and analysis of Hyperledger Fabric	Zuqiang Ke et al. presented a detailed analysis of the performance of HLF using queuing models.	2022	◒	◒	◒	○	○	○

● Meets the criteria. ◒ Partially meets the criteria. ○ Does not meet the criteria.

Table 6
Consensus mechanism modelling evaluation criteria.

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
[77]	Latency Analysis for Raft Consensus on Hyperledger Fabric	Xuefeng Piao et al. provided a theoretical model for the latency of the Raft consensus algorithm in HLF.	2022	◒	○	◒	○	●	◒
[89]	Performance Evaluation of Blockchain Based Agricultural Input Voucher System	Isakwisa Gaddy Tende et al. evaluated the performance of the network based on resources (CPU and memory) consumption metrics of Raft and Kafka consensus protocol.	2021	◒	○	◒	●	◒	◒
[53]	Performance Modelling of PBFT Consensus Process for Permissioned Permissioned Blockchain Network (Hyperledger Fabric)	Harish Sukhwani et al. modelled the mean time to complete consensus for the PBFT consensus process in HLF using Stochastic Reward Nets (SRN).	2017	●	◒	◒	◒	○	◒
[80]	Hyperledger Fabric Blockchain for Securing the Edge Internet of Things	Houshyar Honar Pajooh et al. implemented a permissioned blockchain using HLF for securing Internet of Things edge devices.	2021	●	●	●	●	●	●
[90]	Impact of network delays on Hyperledger Fabric	Thanh Son Lam Nguyen et al. evaluated network delay impact on HLF	2019	○	○	●	●	●	◒
[91]	PLEDGE: A Proof-of-Honesty based Consensus Protocol for Blockchain-based IoT Systems	Imran Makhdoom et al. described the PLEDGE consensus protocol and its implementation in the context of blockchain-based IoT systems.	2020	●	●	●	●	◒	◒
[76]	Weighted RAFT: An Improved Blockchain Consensus Mechanism for Internet of Things Application	Xianjun Xu et al. introduced a weighted RAFT consensus algorithm for IoT applications.	2021	◒	○	◒	○	○	◒
[92]	A Permissioned Blockchain Framework for Supporting Instant Transaction and Dynamic Block Size	Xinping Min et al. provided a Permissioned Blockchain Framework (PBF) and its components to achieve trusted trading and support instant transactions in E-commerce.	2016	◒	○	◒	◒	○	○
[61]	A Byzantine Fault-Tolerant Ordering Service for the Hyperledger Fabric Blockchain Platform	João Sousa et al. assessed a Byzantine Tolerant (BFT) ordering service Fault for the HLF system.	2018	◒	●	●	◒	●	●

● Meets the criteria. ◒ Partially meets the criteria. ○ Does not meet the criteria.

Table 7
Chaincode modelling evaluation criteria.

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
[93]	5G Network Slice Brokering: A Distributed Blockchain-based Market	Nima Afraz et al. evaluated the performance of a blockchain-based slice brokering market regarding transaction latency, throughput, and computing intensity.	2020	◒	○	●	◒	●	◒
[94]	Performance Evaluation of NFT Trading Platform Based on Hyperledger Fabric Blockchain	Ju Won Kim et al. provided detailed information about the proposed NFT trading platform based on HLF.	2022	○	◒	○	●	●	●
[58]	Hyperledger Fabric Blockchain: Chaincode Performance Analysis	Luca Foschini et al. analysed how transaction latency is affected by the programming language adopted for implementing the chaincode.	2020	◒	●	●	◒	●	●

● Meets the criteria. ◒ Partially meets the criteria. ○ Does not meet the criteria.

Table 8
Resource utilization modelling evaluation criteria.

Ref	Title	Authors and their contribution	Year	Evaluation criteria					
				SC	FL	CO	CE	RE	VA
[95]	QiOi: Performance Isolation for Hyperledger Fabric	Jeongsu Kim et al. analysed the stochastic modelling approach to performance interference in HLF caused by co-located services in cloud data centres.	2021	◒	●	◒	●	●	◒
[96]	Performance and availability evaluation of the blockchain platform in hyperledger fabric	Carlos Melo et al. provided a detailed evaluation of the performance and availability of HLF in a private environment managed by a single entity.	2022	◒	○	●	●	●	◒

● Meets the criteria. ◒ Partially meets the criteria. ○ Does not meet the criteria.

5. A summary of research on latency performance modelling, challenges in its application to HLF-enabled IoT use case and future research directions

HLF can address the unique requirements of IoT applications while ensuring privacy for various implementation techniques [97]. The incorporation of HLF with IoT (HLF-enabled IoT) applications, as discovered in literature [98–100], include Industrial IoT (IIoT), healthcare, wireless monitoring, and Unmanned Aerial Vehicles (UAVs). For example, A. Dixit et al. propose a decentralized IoT data marketplace using HLF [99] while F. Jamil et al. used HLF for smart healthcare to disseminate monitored patient vital signs [98], and Aggarwal et al. used UAVs to protect medical data privacy and security in the Healthcare 4.0 sector [100]. Consequently, HLF will be utilized more frequently to manage large IoT data securely [32]. However, the latency problem remains a significant obstacle. Most applications, such as smart industrial, smart transportation, and e-health services, are particularly time-sensitive. In this section, we summarize the latency performance modelling presented by various research, present challenges in applying this performance modelling to HLF-Enabled IoT use cases, and provide future research directions. Fig. 3 below depicts the limitations based on the evaluation criteria and the future research directions.

5.1. A summary of the latency performance modelling by various researchers

Latency: At a fixed BI and varying TSR/TAR and BS, the impact on latency depends on the TSR/TAR and BS values. For low TSR/TAR, the BS values slightly affect overall latency but significantly impact high TSR/TAR latency. When TSR/TAR is below the

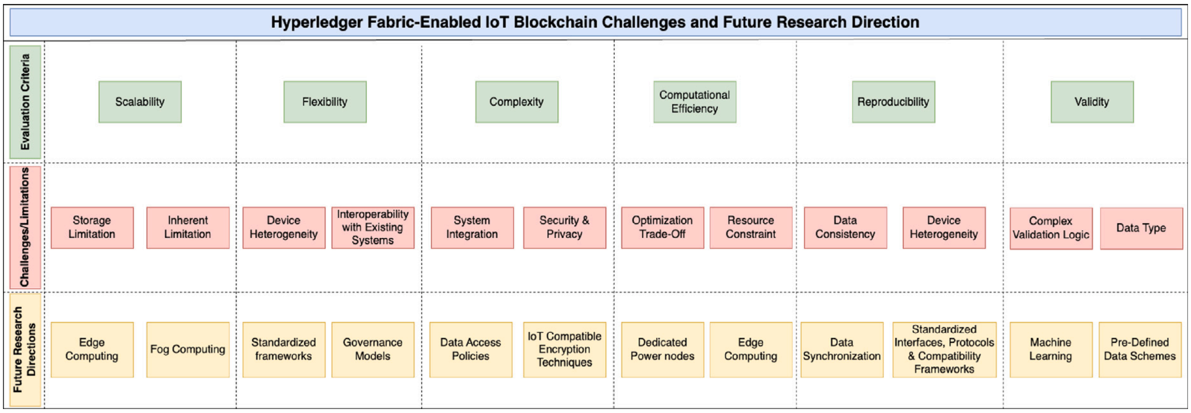

Fig. 4. HLF-Enabled IoT blockchain limitations and future research directions.

BS, latency increases with increasing TSR/TAR. This is because the NOT cannot reach the BS within the BI, causing blocks to be generated due to the BI, leading to increased overall network latency during the ordering phase [47]. On the other hand, for high TSR/TAR, latency increases with increasing BS and decreases with increasing TSR/TAR above the BS. When TSR/TAR is above the BS, the NOT within the BI exceeds the BS, and the ordering service batches pending transactions into a new block once the NOT reaches the block size. This leads to an increase in block generation, resulting in reduced waiting time during each transaction's ordering phase. In this case, the overall latency is primarily due to the validation phase, as blocks queue up for validation [80,87]. With varying TSR/TAR and fixed BS and BI, when TSR/TAR is below the BS, the ordering phase contributes the most to the overall transaction latency as transactions queue for batching during the ordering phase. As TSR/TAR increases, the number of ordered transactions in the validation phase queue also increases, impacting both validation and commit latency. The validation stage becomes the performance bottleneck for the system [47]. Before reaching the system's saturation point, an increase in BS leads to increased latency for the same value of TSR/TAR. This is because the time it takes to generate a block increases, causing latency to increase during the ordering phase. At lower TSR/TAR values below the saturation point, an increase in the number of peers for different block sizes also increases latency [54]. At the system's saturation point, latency increases significantly for all block sizes. The saturation point of TSR/TAR has a strong impact, with latency increasing with an increase in BS for TSR/TAR values below the saturation point [56]. However, for TSR/TAR values greater than the saturation point, latency decreases in the ordering phase with an increase in the BS. In the validation phase, latency increases significantly when TSR/TAR is at or above the saturation point due to the increased number of ordered transactions in the VSCC queue during validation, impacting commit latency [59]. Furthermore, the number of organizations involved also affects latency [37]. More endorsers are required to endorse the transactions, including additional information like certificates, public keys, and signatures of each endorsing peer, leading to increased block size and latency. The number of endorsements sent to the orderer increases, resulting in increased latency during the execution phase [47]. As for the consensus delay in transaction latency, utilizing the Raft consensus protocol, it grows when TSR/TAR increases, regardless of whether the overall transaction latency increases or decreases. Raft includes a transaction block in the "AppendEntries" message during the consensus process, and higher block size results in a longer consensus time [77,89]. PBFT, while resilient, has performance issues due to the time-consuming five steps of the consensus process, leading to long consensus latency [85,90]. The competition for CPU resources causes total latency to increase [37] as the number of channels increases. At moderate loads, performance degrades when the number of channels exceeds the number of vCPUs allocated. Each channel maintains its chain of blocks and is independent of others, resulting in increased CPU utilization and contention during the validation and final ledger update phase, which affects average endorsement and commit latency [47,59]. The endorsement policy "AND" leads to higher transaction latency compared to "OR" because more endorsers are required to endorse transactions in the "AND" policy, causing latency during the execution phase [54,81]. For the validation phase, VSCC latency increases linearly with the number of signatures to verify and rapidly with increased sub-policies. The increase in sub-policies also leads to increased CPU consumption and VSCC latency due to the verification process. Transaction complexity also affects latency. Read-only transactions have extremely low latency, indicating that the blockchain network can handle more transactions. Latency increases with the write workload, and the MVCC latency increases with the number of reads per transaction. The number of GET REST API requests made during the MVCC validation phase increases as the number of items in the read set increases [82]. For RW transactions, the blockchain network can support a certain number of transactions without noticeable network latency. When the transaction rate exceeds the system's threshold, latency increases. Compared to GoLevelDB, CouchDB has more extended endorsement, VSCC, MVCC, and ledger update latencies. Write-only transactions in CouchDB also experience increased endorsement and ledger update latencies with an increasing number of writes per transaction. The endorsement phase in CouchDB obtains a shared read lock on the whole database. In contrast, the ledger update phase obtains an exclusive write lock, leading to resource competition and increased latencies. Additionally, CouchDB performs three tasks for each key–value write in a transaction's write-set, increasing ledger update latency with more writes per transaction. The committers maintain the lock on the database for longer due to these tasks, leading to increased endorsement latency [59].

Throughput: In the "OR" endorsement policy, the throughput does not necessarily increase linearly with the increase in TAR; instead, it reaches a saturation point where it becomes constant. Initially, the throughput increases linearly with TAR until it reaches the saturation point, then flattens out [37,56]. The peak throughput achieved using the "OR" policy is limited by the number of transactions per second and the endorsing peers defined in the endorsement policy. On the other hand, when using the "AND" policy, the peak throughput is influenced by the number of transactions per second, the endorsing peers defined in the endorsement policy and the computing capacity of validation peers. As TAR increases, the throughput increases until it reaches saturation [81]. The Block Size (BS) has little impact on throughput until saturation. For TAR values greater than the saturation point, an increase in the BS results in higher throughput. Until saturation, the throughput remains constant at a fixed TAR value for all BSs. After the saturation point, higher BS values yield slightly higher throughput than smaller BS values [56,59]. Increasing the number of peer nodes leads to lower throughput for TAR values below the saturation point. Similarly, increasing the number of peers for varying BSs lowers the throughput. More peers result in a lower TAR/TSR peak point, affecting overall throughput. If the system operates at its maximum limit, the throughput remains relatively flat as concurrent transactions increase. For varying values of TAR using LevelDB, read throughput scales linearly across the entire range. Reads are efficiently served locally by the peer machine from its optimized LevelDB database. However, for the write workload, the throughput increases almost linearly until it reaches the highest achievable throughput. Beyond this point, performance starts to degrade [83]. With one read and one write per transaction, the blockchain network can handle a specific number of transactions without significant network latency, and the throughput decreases as TAR/TSR increases beyond this point [82]. The throughput decreases with multiple writes using CouchDB, whereas this impact is not observed with GoLevelDB. This significant performance difference between CouchDB and GoLevelDB is because GoLevelDB is an embedded database-to-peer process, whereas CouchDB access is via REST APIs over secure HTTP. As a result, the transaction throughput with GoLevelDB as a state database is greater than CouchDB on a single channel [59].

5.2. Challenges in the application of the latency performance modelling to HLF-enabled IoT use case

Scalability: Scalability presents challenges in HLF-enabled IoT due to processing large volumes of data and managing numerous IoT devices [101]. HLF operates on a peer-to-peer network model, where multiple nodes validate transactions and maintain the blockchain ledger. However, as the network expands, managing peer nodes becomes increasingly intricate. Ensuring the scalability of consensus algorithms becomes critical as the number of IoT devices and associated transaction load grows. IoT devices typically possess limited computational resources, memory, and energy supply, compounding the issue (see Fig. 4).

Flexibility: Flexibility in HLF-enabled IoT challenges arises in the system's ability to adapt and accommodate various IoT devices, data sources, and use cases. Device heterogeneity poses a significant challenge, as IoT devices exhibit variations in communication protocols, data formats, and capabilities. Integrating and managing such diverse devices within the HLF network is complex and requires careful consideration. Another aspect of flexibility challenges lies in interoperability with existing systems. HLF enables the development and execution of smart contracts, which automate processes and transactions within the blockchain network. However, accommodating different IoT use cases and scenarios requires defining and deploying interoperable smart contracts with existing systems, ensuring seamless integration and compatibility [101,102].

Complexity: Integrating diverse components like IoT devices, gateways, and external systems in IoT deployments can be complex and time-consuming due to different protocols, interfaces, and data formats. Ensuring smooth communication and interoperability among these components is a significant challenge. IoT systems face unique security and privacy challenges due to device interconnectivity and sensitive data.

Computational Efficiency: IoT devices often have limited computational resources such as processing power, memory, and energy supply. HLF requires computational resources to process and validate transactions, execute consensus algorithms, and store data. Achieving computational efficiency in HLF-enabled IoT systems often involves trade-offs between performance, resource utilization, and security. Optimizing one aspect, such as transaction processing speed, may come at the expense of increased resource consumption or reduced security. Finding the right balance and optimizing the system for the specific requirements of the IoT application can be challenging.

Reproducibility: Ensuring consistent data across distributed IoT devices and HLF nodes can be challenging. The decentralized nature of IoT systems and the potential for data inconsistencies due to communication failures, device malfunctions, or network disruptions can impact reproducibility. IoT systems often have various devices with different capabilities, software versions, and configurations. Achieving reproducibility across heterogeneous devices can be challenging due to differences in data formats, communication protocols, and operating environments.

Validation: Validating IoT data often requires intricate computations, rules, and conditions based on specific use cases and business logic. Implementing and executing these validation rules efficiently within the HLF framework can be challenging, as it demands careful design and implementation of complex validation algorithms. Maintaining data consistency in a distributed system like HLF is complex, particularly when dealing with large volumes of IoT data generated by multiple devices.

6. Future research directions and open research questions

This section presents the future research directions to the challenges identified in Section 5 above and outlines open research questions. Integrating HLF blockchain with IoT is a promising paradigm driven by potential benefits. HLF can provide enhanced security through a decentralized and tamper-resistant ledger, ensuring data integrity in IoT systems. Its transparent, decentralized nature fosters trust and reduces the reliance on centralized systems. The decentralization aligns well with IoT's distributed architecture, minimizing the risk of a single point of failure. The smart contracts automate processes, improving efficiency and reducing the need for intermediaries. HLF empowers users with greater control over their data, reducing fraud and cyber-attack risks.

6.1. Future research directions

Scalability: Tackling scalability challenges include a comprehensive approach involving architectural design, optimization techniques, and infrastructure scaling strategies. This approach includes implementing data partitioning or sharding techniques to distribute and manage data effectively [103–105]. Employing lightweight protocols and algorithms for resource-constrained devices helps optimize their performance [106–109] while leveraging edge computing, off-chain storage options, and fog computing paradigms allows for efficient processing and storage closer to IoT devices [110–114]. Additionally, utilizing the Decentralized Intelligent Network Edges (DINEs) and Social Federated Edge Learning framework (SFEL) leverage edge computational intelligence to process large-scale sensing data [115,116]. Advanced network technologies such as 5G can also enhance the overall scalability of HLF-enabled IoT. It is also beneficial to incorporate efficient data compression techniques and load balancing strategies to optimize resource utilization and ensure smooth operation [117–120].

Flexibility: Addressing the flexibility challenges requires a combination of technical advancements, industry collaboration, and standardization efforts. Achieving flexibility in HLF-enabled IoT involves establishing governance models, standardized frameworks, and best practices [121,122]. This entails defining guidelines for various aspects, such as device onboarding, data sharing, smart contract development, and system interoperability.

Complexity: Alleviating complexity requires a comprehensive approach, combining technical expertise, architectural design, and industry collaboration. The emphasis should be on modularity in system design and establishing interoperability standards. Breaking down the system into modular components makes integrating and maintaining various elements of the IoT ecosystem easier. By defining standardized protocols and interfaces, different components within the system can communicate seamlessly, reducing integration challenges and enhancing overall interoperability [123–125]. Designing robust data access policies and encryption techniques is crucial for ensuring security and privacy in IoT systems [126–128].

Computational Efficiency:A comprehensive approach is necessary to tackle and enhance computational efficiency in resource-constrained IoT environments. This involves implementing optimization techniques [129,130], improving algorithms, and advancing hardware capabilities. Additionally, leveraging edge computing, offloading computations to more capable nodes, and utilizing specialized hardware accelerators can further enhance efficiency [131,132]. Combining these strategies can effectively address the challenges, improving computational performance in IoT systems with limited resources.

Reproducibility: Reproducibility challenges require establishing mechanisms for data synchronization, error detection and recovery, and ensuring consistency across devices and nodes. Developing standardized interfaces, protocols, and compatibility frameworks can help address reproducibility challenges in IoT systems [123–125].

Validation: To address validation challenges, models can be trained to learn IoT data's expected patterns and behaviours using machine learning algorithms [133–138]. These models can then validate incoming data and identify deviations that significantly differ from the learned patterns—also, ensuring that the data received from IoT devices adhere to the expected data types. This involves verifying if the data is in the correct format and validating it against predefined data schemas.

6.2. Open research questions

The papers examined in this study exclusively concentrated on modelling latency. They modelled various HLF parameters using either the "AND" or "OR" endorsement policy, significantly impacting latency and security. The choice of endorsement policy greatly influences latency in the execution, ordering and validation phases of transaction flow. When an endorsement policy is set to "AND", all designated endorsing peers must successfully execute and validate the transaction. This means that transactions must pass through all the selected endorsers before they can be considered valid. As a result, the latency of transactions tends to be higher because they need to wait for all endorsements to complete. The more endorsers involved, the longer the transaction might take. The "OR" endorsement policy allows a transaction to be considered valid if any of the selected endorsing peers endorse it. This approach reduces latency, as transactions do not need to wait for all endorsing peers to respond. Consequently, transactions can be processed more quickly. The "AND" policy, while increasing latency, provides a higher level of security. Since all endorsing peers must validate and approve the transaction, it is less susceptible to fraudulent or unauthorized transactions. This is particularly important in applications where security is a top priority. On the contrary, the "OR" policy, by allowing a transaction to be valid with just one endorsement, reduces the network's overall security. It opens the door for potentially fraudulent activities, as a malicious endorsing peer could approve a transaction. The choice between "AND" and "OR" endorsement policies involves a trade-off between transaction latency and security.

At the time of this research, there is a noticeable gap in exploring the ideal endorsement policy that could minimize latency while ensuring security. One promising avenue to address this challenge is leveraging artificial intelligence to enable dynamic endorsements [139,140]. Dynamic endorsement is the ability to dynamically determine the number of endorsers required for a transaction based on specific criteria. In a typical HLF network, the endorsement policy specifies a fixed number or percentage of endorsing peers who must sign off on a transaction to be considered valid. However, dynamic endorsement allows more flexibility and adaptability in determining the required endorsers. Dynamic endorsement defines rules and conditions that determine the number of endorsers based on transaction-specific attributes, participant identities, or other criteria. This capability enables fine-grained control over the endorsement process and allows for less complex endorsement policies. Dynamic endorsement in HLF adds a layer of customization and adaptability to the endorsement process, allowing organizations to tailor the endorsement policy to their specific use cases and transaction requirements. It enhances the flexibility and scalability of the network by providing more control over the endorsement process. By using dynamic endorsement, the performance and probable security of the HLF network are optimized by adjusting the number of endorsers based on the specific needs of different transactions [141]. For example, a higher number of endorsers may be required for high-value or critical transactions, while a lower number for less critical or low-value transactions may be allowed. This flexibility can help balance latency and security, adapting to the varying requirements of different types of transactions. The following are the questions open for further research;

1. How do organizations currently navigate latency and security trade-offs when selecting between the "AND" and "OR" endorsement policies in HLF?
2. How can artificial intelligence be leveraged to dynamically determine the optimal number of endorsers required for a transaction in HLF, considering specific transaction attributes, participant identities, and other criteria?
3. What are the technical challenges and considerations in implementing dynamic endorsement in HLF, and how do these challenges impact the overall performance and security of the network?
4. How does dynamic endorsement enhance the flexibility and scalability of HLF networks, and what are the potential implications for transaction processing efficiency and security?
5. What are the practical implications of dynamic endorsement for organizations deploying HLF, and how does it enable customization and adaptability in the endorsement process?

7. Conclusions

Recent studies have examined the performance of HLF blockchain systems, emphasizing analysing latency. However, there is a lack of analysis on how these research efforts collectively impact real-world use cases. This paper proposes a novel set of evaluation criteria to assess the existing research and identify challenges in their application to the IoT use case under the evaluation criteria. IoT infrastructures are dynamic and diverse. For IoT systems, which usually involve massive devices generating data at high rates, it can be challenging to scale blockchain networks to handle such transactions efficiently. IoT nodes are spread out, differ in characteristics, and can be unpredictable. This poses a flexibility challenge, primarily due to device heterogeneity and interoperability challenges with existing systems. Integrating IoT components is complex due to varied protocols and interfaces, presenting communication challenges. The constraints of limited computing resources impact computational efficiency. Also, some energy-intensive consensus mechanisms in HLF networks may not align with the energy-efficient requirements of IoT devices. Differences in data formats, communication protocols, and configurations complicate reproducibility and validation across heterogeneous devices.

In addressing these challenges, we suggest a holistic approach for the scalability challenge in HLF-enabled IoT, involving architectural design, optimization, and infrastructure scaling with data partitioning and lightweight protocols. Utilizing DINEs, SFEL, and 5G enhances scalability while efficient data compression, load balancing, and off-chain storage optimize resources. Flexibility challenges can be addressed through technical advancements, industry collaboration, and standardization, including governance models and guidelines for device onboarding and smart contract development. Alleviating complexity involves a comprehensive approach focusing on modularity, interoperability, and robust data access policies. Enhancing computational efficiency includes optimization, edge computing, and hardware advancements in resource-constrained IoT, utilizing strategies like offloading computations. Reproducibility challenges can be tackled with mechanisms for data synchronization and standardized interfaces, aided by compatibility frameworks, while the validation challenges are addressed by training models using machine learning to recognize expected IoT data patterns, ensuring adherence to predefined schemas.

In considering the evaluation criteria for assessing latency performance in IoT systems, it is important to acknowledge potential limitations and biases that can impact the objectivity and comprehensiveness of these assessments. The selection of criteria may inadvertently reflect personal preferences or specific areas of expertise, potentially overlooking other critical aspects of latency optimization. Assumptions underlying these criteria must be carefully scrutinized to ensure they accurately represent real-world system behaviour and workload characteristics, avoiding the introduction of sensitivity or bias. A narrow focus on specific dimensions of latency, such as scalability or efficiency, might inadvertently neglect broader considerations like security and usability, leading to incomplete evaluations. Complex interactions among evaluation criteria can obscure the holistic impacts of latency optimization strategies, requiring careful analysis to avoid biased conclusions. Standardization of definitions and methodologies is crucial to promoting consistency and comparability across different studies, reducing variability and potential biases in interpretations. Validation methods must be carefully chosen to accurately reflect real-world deployment scenarios, avoiding biases introduced by reliance on simulated environments alone. Additionally, contextual factors such as network topology and user requirements should be integrated into criteria development to ensure evaluations are relevant to practical deployment scenarios. Through transparency, rigour, and continuous refinement based on feedback and emerging challenges, the validity and applicability of evaluation criteria can be enhanced, driving meaningful advancements in IoT latency optimization research.

CRediT authorship contribution statement

Jummai Enare Abang: Writing – original draft. **Haifa Takruri:** Writing – review & editing, Supervision. **Rabab Al-Zaidi:** Writing – review & editing, Supervision. **Mohammed Al-Khalidi:** Writing – review & editing, Visualization, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] M. Uddin, Blockchain Medledger: Hyperledger fabric enabled drug traceability system for counterfeit drugs in pharmaceutical industry, Int. J. Pharm. 597 (2021) 120235, http://dx.doi.org/10.1016/j.ijpharm.2021.120235.

[2] R. Sekaran, R. Patan, A. Raveendran, F. Al-Turjman, M. Ramachandran, L. Mostarda, Survival study on blockchain based 6G-enabled mobile edge computation for IoT automation, IEEE Access 8 (2020) 143453–143463, http://dx.doi.org/10.1109/ACCESS.2020.3013946.

[3] Y. Sharma, B. Balamurugan, Preserving the privacy of electronic health records using blockchain, Procedia Comput. Sci. 173 (2020) 171–180, http://dx.doi.org/10.1016/j.procs.2020.06.021.

[4] Z. Tian, M. Li, M. Qiu, Y. Sun, S. Su, Block-DEF: A secure digital evidence framework using blockchain, Inform. Sci. 491 (2019) 151–165, http://dx.doi.org/10.1016/j.ins.2019.04.011.

[5] A. Pawar, D. Barthare, N. Rawat, M. Yadav, M. Shirole, BlockAudit 2.0: PoA blockchain based solution for secure audit logs, in: 2021 5th International Conference on Information Systems and Computer Networks, ISCON, IEEE, 2021, pp. 1–6, http://dx.doi.org/10.1109/ISCON52037.2021.9702378.

[6] A. Hari, T. Lakshman, The internet blockchain: A distributed, tamper-resistant transaction framework for the internet, in: Proceedings of the 15th ACM Workshop on Hot Topics in Networks, 2016, pp. 204–210, http://dx.doi.org/10.1145/3005745.3005771.

[7] H.M.M. Khan, W. Saeed, M.W. Iqbal, A. Ali, M. Zuraiz, M.N. Shahzad, M. Ahmed, The promises of blockchain and cryptocurrencies technology for architecture and interaction design, Int. J. Adv. Trends Comput. Sci. Eng. 10 (3) (2021) http://dx.doi.org/10.30534/ijatcse/2021/1601032021.

[8] A.P. Joshi, M. Han, Y. Wang, A survey on security and privacy issues of blockchain technology, Math. Found. Comput. 1 (2) (2018) 121–147, http://dx.doi.org/10.3934/mfc.2018007.

[9] A.A. Siyal, A.Z. Junejo, M. Zawish, K. Ahmed, A. Khalil, G. Soursou, Applications of blockchain technology in medicine and healthcare: Challenges and future perspectives, Cryptography 3 (1) (2019) 3, http://dx.doi.org/10.3390/cryptography3010003.

[10] D. Marbouh, T. Abbasi, F. Maasmi, I.A. Omar, M.S. Debe, K. Salah, R. Jayaraman, S. Ellahham, Blockchain for COVID-19: review, opportunities, and a trusted tracking system, Arabian J. Sci. Eng. 45 (2020) 9895–9911, http://dx.doi.org/10.1007/s13369-020-04950-4.

[11] P. Sandner, A. Lange, P. Schulden, The role of the CFO of an industrial company: an analysis of the impact of blockchain technology, Future Internet 12 (8) (2020) 128, http://dx.doi.org/10.3390/fi12080128.

[12] B. Arunkumar, G. Kousalya, Blockchain-based decentralized and secure lightweight e-health system for electronic health records, in: Intelligent Systems, Technologies and Applications: Proceedings of Fifth ISTA 2019, India, Vol. 1148, Springer, 2020, pp. 273–289, http://dx.doi.org/10.1007/978-981-15-3914-5_21.

[13] B. Sowmiya, E. Poovammal, K. Ramana, S. Singh, B. Yoon, Linear elliptical curve digital signature (LECDS) with blockchain approach for enhanced security on cloud server, IEEE Access 9 (2021) 138245–138253, http://dx.doi.org/10.1109/ACCESS.2021.3115238.

[14] R.W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, M. Omar, Blockchain in oil and gas industry: Applications, challenges, and future trends, Technol. Soc. 68 (2022) 101941, http://dx.doi.org/10.1016/j.techsoc.2022.101941.

[15] R.W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, S. Ellahham, M. Omar, The role of blockchain technology in telehealth and telemedicine, Int.. J. Med. Inf. 148 (2021) 104399, http://dx.doi.org/10.1016/j.ijmedinf.2021.104399.

[16] M.J.M. Chowdhury, M.S. Ferdous, K. Biswas, N. Chowdhury, A. Kayes, M. Alazab, P. Watters, A comparative analysis of distributed ledger technology platforms, IEEE Access 7 (2019) 167930–167943, http://dx.doi.org/10.1109/ACCESS.2019.2953729.

[17] O.J. Ajayi, J. Rafferty, J. Santos, M. Garcia-Constantino, Z. Cui, BECA: A blockchain-based edge computing architecture for internet of things systems, IoT 2 (4) (2021) 610–632, http://dx.doi.org/10.3390/iot2040031.

[18] M. Niranjanamurthy, B. Nithya, S. Jagannatha, Analysis of blockchain technology: pros, cons and SWOT, Cluster Comput. 22 (2019) 14743–14757, http://dx.doi.org/10.1007/s10586-018-2387-5.

[19] P. Helo, Y. Hao, Blockchains in operations and supply chains: A model and reference implementation, Comput. Ind. Eng. 136 (2019) 242–251, http://dx.doi.org/10.1016/j.cie.2019.07.023.

[20] U. Rahardja, A.N. Hidayanto, T. Hariguna, Q. Aini, Design framework on tertiary education system in Indonesia using blockchain technology, in: 2019 7th International Conference on Cyber and IT Service Management, CITSM, Vol. 7, IEEE, 2020, pp. 1–4, http://dx.doi.org/10.1109/CITSM47753.2019.8965380.

[21] K.K. Vaigandla, R. Karne, M. Siluveru, M. Kesoju, Review on blockchain technology: Architecture, characteristics, benefits, algorithms, challenges and applications, Mesopotamian J. CyberSecur. 2023 (2023) 73–85, http://dx.doi.org/10.58496/MJCS/2023/012.

[22] M.D. Borah, V.B. Naik, R. Patgiri, A. Bhargav, B. Phukan, S.G. Basani, Supply chain management in agriculture using blockchain and IoT, Adv. Appl. Blockchain Technol. 60 (2020) 227–242, http://dx.doi.org/10.1007/978-981-13-8775-3_11.

[23] W. Viriyasitavat, L. Da Xu, Z. Bi, V. Pungpapong, Blockchain and internet of things for modern business process in digital economy—the state of the art, IEEE Trans. Comput. Soc. Syst. 6 (6) (2019) 1420–1432, http://dx.doi.org/10.1109/TCSS.2019.2919325.

[24] A. Mohammed, A. Almousa, A. Ghaithan, L.A. Hadidi, The role of blockchain in improving the processes and workflows in construction projects, Appl. Sci. 11 (19) (2021) 8835, http://dx.doi.org/10.3390/app11198835.

[25] V. Dedeoglu, R. Jurdak, A. Dorri, R. Lunardi, R. Michelin, A. Zorzo, S. Kanhere, Blockchain technologies for iot, Adv. Appl. Blockchain Technol. 60 (2020) 55–89, http://dx.doi.org/10.1007/978-981-13-8775-3_3.

[26] A. Miglani, N. Kumar, V. Chamola, S. Zeadally, Blockchain for Internet of Energy management: Review, solutions, and challenges, Comput. Commun. 151 (2020) 395–418, http://dx.doi.org/10.1016/j.comcom.2020.01.014.

[27] S.-Y. Lin, L. Zhang, J. Li, L.-l. Ji, Y. Sun, A survey of application research based on blockchain smart contract, Wirel. Netw. 28 (2) (2022) 635–690, http://dx.doi.org/10.1007/s11276-021-02874-x.

[28] S.S. Gill, S. Tuli, M. Xu, I. Singh, K.V. Singh, D. Lindsay, S. Tuli, D. Smirnova, M. Singh, U. Jain, et al., Transformative effects of IoT, Blockchain and Artificial Intelligence on cloud computing: Evolution, vision, trends and open challenges, Internet Things 8 (2019) 100118, http://dx.doi.org/10.1016/j.iot.2019.100118.

[29] S. Zhao, S. Li, Y. Yao, Blockchain enabled industrial Internet of Things technology, IEEE Trans. Comput. Soc. Syst. 6 (6) (2019) 1442–1453, http://dx.doi.org/10.1109/TCSS.2019.2924054.

[30] S. Singh, P.K. Sharma, B. Yoon, M. Shojafar, G.H. Cho, I.-H. Ra, Convergence of blockchain and artificial intelligence in IoT network for the sustainable smart city, Sustain. Cities Soc. 63 (2020) 102364, http://dx.doi.org/10.1016/j.scs.2020.102364.

[31] B. Zhong, H. Wu, L. Ding, H. Luo, Y. Luo, X. Pan, Hyperledger fabric-based consortium blockchain for construction quality information management, Front. Eng. Manag. 7 (4) (2020) 512–527, http://dx.doi.org/10.1007/s42524-020-0128-y.

[32] S. Lee, M. Kim, J. Lee, R.-H. Hsu, M.-S. Kim, T.Q. Quek, Facing to latency of hyperledger fabric for blockchain-enabled IoT: Modeling and analysis, IEEE Netw. (Early Access) (2023) 1–8, http://dx.doi.org/10.1109/MNET.120.2200064.

[33] M. Jo, K. Hu, R. Yu, L. Sun, M. Conti, Q. Du, Private blockchain in industrial IoT, IEEE Netw. 34 (5) (2020) 76–77, http://dx.doi.org/10.1109/MNET.2020.9199796.

[34] A. Sarma, Smart contracts: A way to modern digital world, in: Blockchain and Deep Learning: Future Trends and Enabling Technologies, Vol. 105, Springer, 2022, pp. 67–106, http://dx.doi.org/10.1007/978-3-030-95419-2_4.

[35] D. Li, W.E. Wong, J. Guo, A survey on blockchain for enterprise using hyperledger fabric and composer, in: 2019 6th International Conference on Dependable Systems and their Applications, DSA, IEEE, 2020, pp. 71–80, http://dx.doi.org/10.1109/DSA.2019.00017.

[36] S. Dalla Palma, R. Pareschi, F. Zappone, What is your distributed (hyper) ledger? in: 2021 IEEE/ACM 4th International Workshop on Emerging Trends in Software Engineering for Blockchain, WETSEB, IEEE, 2021, pp. 27–33, http://dx.doi.org/10.1109/WETSEB52558.2021.00011.

[37] H. Honar Pajooh, M.A. Rashid, F. Alam, S. Demidenko, Experimental performance analysis of a scalable distributed hyperledger fabric for a large-scale IoT testbed, Sensors 22 (13) (2022) 4868, http://dx.doi.org/10.3390/s22134868.

[38] D. Ravi, S. Ramachandran, R. Vignesh, V.R. Falmari, M. Brindha, Privacy preserving transparent supply chain management through Hyperledger Fabric, Blockchain: Res. Appl. 3 (2) (2022) 100072, http://dx.doi.org/10.1016/j.bcra.2022.100072.

[39] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, et al., Hyperledger fabric: a distributed operating system for permissioned blockchains, in: Proceedings of the Thirteenth EuroSys Conference, Vol. 30, ACM Digital Library, 2018, pp. 1–15, http://dx.doi.org/10.1145/3190508.3190538.

[40] N.O. Nawari, Blockchain technologies: Hyperledger fabric in BIM work processes, in: Proceedings of the 18th International Conference on Computing in Civil and Building Engineering, ICCCBE 2020, Springer, 2021, pp. 813–823, http://dx.doi.org/10.1007/978-3-030-51295-8_56.

[41] C. Stamatellis, P. Papadopoulos, N. Pitropakis, S. Katsikas, W.J. Buchanan, A privacy-preserving healthcare framework using hyperledger fabric, Sensors 20 (22) (2020) 6587, http://dx.doi.org/10.3390/s20226587.

[42] N.O. Nawari, S. Ravindran, Blockchain technology and BIM process: review and potential applications, J. Inf. Technol. Constr. 24 (2019) 209–238, https://www.itcon.org/2019/12.

[43] A. Dubovitskaya, F. Baig, Z. Xu, R. Shukla, P.S. Zambani, A. Swaminathan, M.M. Jahangir, K. Chowdhry, R. Lachhani, N. Idnani, et al., ACTION-EHR: Patient-centric blockchain-based electronic health record data management for cancer care, J. Med. Internet Res. 22 (8) (2020) e13598, http://dx.doi.org/10.2196/13598.

[44] C. Esposito, M. Ficco, B.B. Gupta, Blockchain-based authentication and authorization for smart city applications, Inf. Process. Manage. 58 (2) (2021) 102468, http://dx.doi.org/10.1016/j.ipm.2020.102468.

[45] S. Rouhani, R. Belchior, R.S. Cruz, R. Deters, Distributed attribute-based access control system using permissioned blockchain, World Wide Web 24 (2021) 1617–1644, http://dx.doi.org/10.1007/s11280-021-00874-7.

[46] H. Liu, D. Han, D. Li, Fabric-IoT: A blockchain-based access control system in IoT, IEEE Access 8 (2020) 18207–18218, http://dx.doi.org/10.1109/ACCESS.2020.2968492.

[47] X. Xu, G. Sun, L. Luo, H. Cao, H. Yu, A.V. Vasilakos, Latency performance modeling and analysis for hyperledger fabric blockchain network, Inf. Process. Manage. 58 (1) (2021) 102436, http://dx.doi.org/10.1016/j.ipm.2020.102436.

[48] T.T.A. Dinh, J. Wang, G. Chen, R. Liu, B.C. Ooi, K.-L. Tan, Blockbench: A framework for analyzing private blockchains, in: Proceedings of the 2017 ACM International Conference on Management of Data, ACM Digital Library, 2017, pp. 1085–1100, http://dx.doi.org/10.1145/3035918.3064033.

[49] C. Fan, S. Ghaemi, H. Khazaei, P. Musilek, Performance evaluation of blockchain systems: A systematic survey, IEEE Access 8 (2020) 126927–126950, http://dx.doi.org/10.1109/ACCESS.2020.3006078.

[50] M. Dabbagh, M. Kakavand, M. Tahir, A. Amphawan, Performance analysis of blockchain platforms: Empirical evaluation of hyperledger fabric and ethereum, in: 2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology, IICAIET, IEEE, 2020, pp. 1–6, http://dx.doi.org/10.1109/IICAIET49801.2020.9257811.

[51] M. Dabbagh, K.-K.R. Choo, A. Beheshti, M. Tahir, N.S. Safa, A survey of empirical performance evaluation of permissioned blockchain platforms: Challenges and opportunities, Comput. Secur. 100 (2021) 102078, http://dx.doi.org/10.1016/j.cose.2020.102078.

[52] Y. Zhou, X. Luo, M. Zhou, Cryptocurrency transaction network embedding from static and dynamic perspectives: An overview, IEEE/CAA J. Autom. Sin. 10 (5) (2023) 1105–1121, http://dx.doi.org/10.1109/JAS.2023.123450.

[53] H. Sukhwani, J.M. Martínez, X. Chang, K.S. Trivedi, A. Rindos, Performance modeling of PBFT consensus process for permissioned blockchain network (hyperledger fabric), in: 2017 IEEE 36th Symposium on Reliable Distributed Systems, SRDS, IEEE, 2017, pp. 253–255, http://dx.doi.org/10.1109/SRDS.2017.36.

[54] H. Sukhwani, N. Wang, K.S. Trivedi, A. Rindos, Performance modeling of hyperledger fabric (permissioned blockchain network), in: 2018 IEEE 17th International Symposium on Network Computing and Applications, NCA, IEEE, 2018, pp. 1–8, http://dx.doi.org/10.1109/NCA.2018.8548070.

[55] L. Jiang, X. Chang, Y. Liu, J. Mišić, V.B. Mišić, Performance analysis of Hyperledger Fabric platform: A hierarchical model approach, Peer-to-Peer Netw. Appl. 13 (2020) 1014–1025, http://dx.doi.org/10.1007/s12083-019-00850-z.

[56] P. Gaba, R.S. Raw, M.A. Mohammed, J. Nedoma, R. Martinek, Impact of block data components on the performance of blockchain-based VANET implemented on hyperledger fabric, IEEE Access 10 (2022) 71003–71018, http://dx.doi.org/10.1109/ACCESS.2022.3188296.

[57] A.S. Yadav, V. Charles, D.K. Pandey, S. Gupta, T. Gherman, D.S. Kushwaha, Blockchain-based secure privacy-preserving vehicle accident and insurance registration, Expert Syst. Appl. (2023) 120651, http://dx.doi.org/10.1016/j.eswa.2023.120651.

[58] L. Foschini, A. Gavagna, G. Martuscelli, R. Montanari, Hyperledger fabric blockchain: Chaincode performance analysis, in: ICC 2020-2020 IEEE International Conference on Communications, ICC, IEEE, 2020, pp. 1–6, http://dx.doi.org/10.1109/ICC40277.2020.9149080.

[59] P. Thakkar, S. Nathan, B. Viswanathan, Performance benchmarking and optimizing hyperledger fabric blockchain platform, in: 2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, MASCOTS, IEEE, 2018, pp. 264–276, http://dx.doi.org/10.1109/MASCOTS.2018.00034.

[60] M.Q. Nguyen, D. Loghin, T.T.A. Dinh, Understanding the scalability of hyperledger fabric, 2021, http://dx.doi.org/10.48550/arXiv.2107.09886, arXiv preprint arXiv:2107.09886.

[61] J. Sousa, A. Bessani, M. Vukolic, A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform, in: 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN, IEEE, 2018, pp. 51–58, http://dx.doi.org/10.1109/DSN.2018.00018.

[62] J. Camenisch, E. Van Herreweghen, Design and implementation of the idemix anonymous credential system, in: Proceedings of the 9th ACM Conference on Computer and Communications Security, ACM Digital Library, 2002, pp. 21–30, http://dx.doi.org/10.1145/586110.586114.

[63] S. Quamara, A.K. Singh, An in-depth security and performance investigation in hyperledger fabric-configured distributed computing systems, Int. J. Comput. Digit. Syst. 13 (1) (2023) 179–191, http://dx.doi.org/10.12785/ijcds/130115.

[64] H. Trabelsi, K. Zhang, Early detection for multiversion concurrency control conflicts in hyperledger fabric, Distrib., Parallel, Cluster Comput. 1 (2023) http://dx.doi.org/10.48550/arXiv.2301.06181.

[65] S. Wang, Performance evaluation of hyperledger fabric with malicious behavior, in: Blockchain–ICBC 2019: Second International Conference, Held As Part of the Services Conference Federation, SCF 2019, San Diego, CA, USA, June 25–30, 2019, Proceedings 2, Vol. 11521, Springer, 2019, pp. 211–219, http://dx.doi.org/10.1007/978-3-030-23404-1_15.

[66] H. Javaid, C. Hu, G. Brebner, Optimizing validation phase of hyperledger fabric, in: 2019 IEEE 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, MASCOTS, IEEE, 2019, pp. 269–275, http://dx.doi.org/10.1109/MASCOTS.2019.00038.

[67] S.O. Ajakwe, D.-S. Kim, J.M. Lee, Drone transportation system: Systematic review of security dynamics for smart mobility, IEEE Internet Things J. 10 (16) (2023) 14462–14482, http://dx.doi.org/10.1109/JIOT.2023.3266843.

[68] P. Yuan, K. Zheng, X. Xiong, K. Zhang, L. Lei, Performance modeling and analysis of a hyperledger-based system using GSPN, Comput. Commun. 153 (2020) 117–124, http://dx.doi.org/10.1016/j.comcom.2020.01.073.

[69] C. Edirimanna, P. Jayasena, Performance comparison of IoT based metering system with different blockchain platforms, in: International Conference on Advances in Computing and Technology (ICACT–2020) Proceedings, 2020, https://fct.kln.ac.lk/media/pdf/proceedings/ICACT-2020/B-9.pdf.

[70] S. Pongnumkul, C. Siripanpornchana, S. Thajchayapong, Performance analysis of private blockchain platforms in varying workloads, in: 2017 26th International Conference on Computer Communication and Networks, ICCCN, IEEE, 2017, pp. 1–6, http://dx.doi.org/10.1109/ICCCN.2017.8038517.

[71] Q. Nasir, I.A. Qasse, M. Abu Talib, A.B. Nassif, et al., Performance analysis of hyperledger fabric platforms, Secur. Commun. Netw. 2018 (2018) http://dx.doi.org/10.1155/2018/3976093.

[72] J. Yli-Huumo, D. Ko, S. Choi, S. Park, K. Smolander, Where is current research on blockchain technology?—a systematic review, PLoS One 11 (10) (2016) e0163477, http://dx.doi.org/10.1371/journal.pone.0163477.

[73] M. Shuaib, N.H. Hassan, S. Usman, S. Alam, N.A.A. Bakar, N. Maarop, Performance evaluation of DLT systems based on hyper ledger fabric, in: 2022 4th International Conference on Smart Sensors and Application, ICSSA, IEEE, 2022, pp. 70–75, http://dx.doi.org/10.1109/ICSSA54161.2022.9870957.

[74] L. Hang, D.-H. Kim, Optimal blockchain network construction methodology based on analysis of configurable components for enhancing hyperledger fabric performance, Blockchain: Res. Appl. 2 (1) (2021) 100009, http://dx.doi.org/10.1016/j.bcra.2021.100009.

[75] S. Zhang, S. Hua, B. Pi, J. Sun, K. Yamashita, Y. Nomura, Performance diagnosis and optimization for hyperledger fabric, in: 2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services, BRAINS, IEEE, 2020, pp. 210–211, http://dx.doi.org/10.1109/BRAINS49436.2020.9223271.

[76] X. Xu, L. Hou, Y. Li, Y. Geng, Weighted raft: An improved blockchain consensus mechanism for internet of things application, in: 2021 7th International Conference on Computer and Communications, ICCC, IEEE, 2021, pp. 1520–1525, http://dx.doi.org/10.1109/ICCC54389.2021.9674683.

[77] X. Piao, M. Li, F. Meng, H. Song, Latency analysis for raft consensus on hyperledger fabric, in: Blockchain and Trustworthy Systems. BlockSys 2022. Communications in Computer and Information Science, Springer, 2022, pp. 165–176, http://dx.doi.org/10.1007/978-981-19-8043-5_12.

[78] M. Castro, B. Liskov, Practical byzantine fault tolerance and proactive recovery, ACM Trans. Comput. Syst. (TOCS) 20 (4) (2002) 398–461, http://dx.doi.org/10.1145/571637.571640.

[79] Z. Ke, N. Park, Performance modeling and analysis of hyperledger fabric, Cluster Comput. 26 (2022) 2681—-2699, http://dx.doi.org/10.1007/s10586-022-03800-2.

[80] H. Honar Pajooh, M. Rashid, F. Alam, S. Demidenko, Hyperledger fabric blockchain for securing the edge internet of things, Sensors 21 (2) (2021) 359, http://dx.doi.org/10.3390/s21020359.

[81] C. Wang, X. Chu, Performance characterization and bottleneck analysis of hyperledger fabric, in: 2020 IEEE 40th International Conference on Distributed Computing Systems, ICDCS, IEEE, 2020, pp. 1281–1286, http://dx.doi.org/10.1109/ICDCS47774.2020.00165.

[82] M. Kuzlu, M. Pipattanasomporn, L. Gurses, S. Rahman, Performance analysis of a hyperledger fabric blockchain framework: throughput, latency and scalability, in: 2019 IEEE International Conference on Blockchain, Blockchain, IEEE, 2019, pp. 536–540, http://dx.doi.org/10.1109/Blockchain.2019.00003.

[83] A. Baliga, N. Solanki, S. Verekar, A. Pednekar, P. Kamat, S. Chatterjee, Performance characterization of hyperledger fabric, in: 2018 Crypto Valley Conference on Blockchain Technology, CVCBT, IEEE, 2018, pp. 65–74, http://dx.doi.org/10.1109/CVCBT.2018.00013.

[84] Y.-F. Wen, C.-M. Hsu, Research of the modular operational performance analysis for consortium blockchain, in: 2021 International Symposium on Networks, Computers and Communications, ISNCC, IEEE, 2021, pp. 1–6, http://dx.doi.org/10.1109/ISNCC52172.2021.9615839.

[85] M. Liu, F.R. Yu, Y. Teng, V.C. Leung, M. Song, Performance optimization for blockchain-enabled industrial Internet of Things (IIoT) systems: A deep reinforcement learning approach, IEEE Trans. Ind. Inform. 15 (6) (2019) 3559–3570, http://dx.doi.org/10.1109/TII.2019.2897805.

[86] C.-M. Liu, M. Badigineni, S.W. Lu, Adaptive blocksize for IoT payload data on fabric blockchain, in: 2021 30th Wireless and Optical Communications Conference, WOCC, IEEE, 2021, pp. 92–96, http://dx.doi.org/10.1109/WOCC53213.2021.9602935.

[87] X. Xu, X. Wang, Z. Li, H. Yu, G. Sun, S. Maharjan, Y. Zhang, Mitigating conflicting transactions in hyperledger fabric-permissioned blockchain for delay-sensitive IoT applications, IEEE Internet Things J. 8 (13) (2021) 10596–10607, http://dx.doi.org/10.1109/JIOT.2021.3050244.

[88] T. Nakaike, Q. Zhang, Y. Ueda, T. Inagaki, M. Ohara, Hyperledger fabric performance characterization and optimization using goleveldb benchmark, in: 2020 IEEE International Conference on Blockchain and Cryptocurrency, ICBC, IEEE, 2020, pp. 1–9, http://dx.doi.org/10.1109/ICBC48266.2020.9169454.

[89] I.G. Tende, K. Aburada, H. Yamaba, T. Katayama, N. Okazaki, Performance evaluation of blockchain based agricultural input voucher system, in: 2021 IEEE 10th Global Conference on Consumer Electronics, GCCE, IEEE, 2021, pp. 637–638, http://dx.doi.org/10.1109/GCCE53005.2021.9622001.

[90] T.S.L. Nguyen, G. Jourjon, M. Potop-Butucaru, K.L. Thai, Impact of network delays on Hyperledger Fabric, in: IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, IEEE, 2019, pp. 222–227, http://dx.doi.org/10.1109/INFCOMW.2019.8845168.

[91] I. Makhdoom, F. Tofigh, I. Zhou, M. Abolhasan, J. Lipman, PLEDGE: A proof-of-honesty based consensus protocol for blockchain-based IoT systems, in: 2020 IEEE International Conference on Blockchain and Cryptocurrency, ICBC, IEEE, 2020, pp. 1–3, http://dx.doi.org/10.1109/ICBC48266.2020.9169406.

[92] X. Min, Q. Li, L. Liu, L. Cui, A permissioned blockchain framework for supporting instant transaction and dynamic block size, in: 2016 IEEE Trustcom/BigDataSE/ISPA, IEEE, 2016, pp. 90–96, http://dx.doi.org/10.1109/TrustCom.2016.0050.

[93] N. Afraz, M. Ruffini, 5G network slice brokering: A distributed blockchain-based market, in: 2020 European Conference on Networks and Communications, EuCNC, IEEE, 2020, pp. 23–27, http://dx.doi.org/10.1109/EuCNC48522.2020.9200915.

[94] J.W. Kim, J.G. Song, T.R. Lee, J.W. Jang, Performance evaluation of NFT trading platform based on hyperledger fabric blockchain, in: 2022 the 8th International Conference on Computing and Data Engineering, ACm Digital Library, 2022, pp. 65–70, http://dx.doi.org/10.1145/3512850.3512855.

[95] J. Kim, K. Lee, G. Yang, K. Lee, J. Im, C. Yoo, Qioi: performance isolation for hyperledger fabric, Appl. Sci. 11 (9) (2021) 3870, http://dx.doi.org/10.3390/app11093870.

[96] C. Melo, F. Oliveira, J. Dantas, J. Araujo, P. Pereira, R. Maciel, P. Maciel, Performance and availability evaluation of the blockchain platform hyperledger fabric, J. Supercomput. 78 (10) (2022) 12505–12527, http://dx.doi.org/10.1007/s11227-022-04361-2.

[97] H. Honar Pajooh, Blockchain for Secured Iot and D2D Applications Over 5G Cellular Networks: a Thesis by Publications Presented in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Computer and Electronics Engineering, Massey University, Albany, New Zealand (Ph.D. thesis), Massey University, 2021, http://hdl.handle.net/10179/17112.

[98] F. Jamil, S. Ahmad, N. Iqbal, D.-H. Kim, Towards a remote monitoring of patient vital signs based on IoT-based blockchain integrity management platforms in smart hospitals, Sensors 20 (8) (2020) 2195, http://dx.doi.org/10.3390/s20082195.

[99] A. Dixit, A. Singh, Y. Rahulamathavan, M. Rajarajan, Fast data: A fair, secure and trusted decentralized iiot data marketplace enabled by blockchain, IEEE Internet Things J. 10 (4) (2021) 2934–2944, http://dx.doi.org/10.1109/JIOT.2021.3120640.

[100] S. Aggarwal, N. Kumar, M. Alhussein, G. Muhammad, Blockchain-based UAV path planning for healthcare 4.0: Current challenges and the way ahead, IEEE Netw. 35 (1) (2021) 20–29, http://dx.doi.org/10.1109/MNET.011.2000069.

[101] Z. Rahman, X. Yi, I. Khalil, A. Kelarev, Blockchain for iot: A critical analysis concerning performance and scalability, in: Quality, Reliability, Security and Robustness in Heterogeneous Systems: 17th EAI International Conference, QShine 2021, Virtual Event, November 29–30, 2021, Proceedings 17, Vol. 402, Springer, 2021, pp. 57–74, http://dx.doi.org/10.1007/978-3-030-91424-0_4.

[102] A. Abdelmaboud, A.I.A. Ahmed, M. Abaker, T.A.E. Eisa, H. Albasheer, S.A. Ghorashi, F.K. Karim, Blockchain for IoT applications: taxonomy, platforms, recent advances, challenges and future research directions, Electronics 11 (4) (2022) 630, http://dx.doi.org/10.3390/electronics11040630.

[103] W. Tong, X. Dong, Y. Shen, X. Jiang, A hierarchical sharding protocol for multi-domain iot blockchains, in: ICC 2019-2019 IEEE International Conference on Communications, ICC, IEEE, 2019, pp. 1–6, http://dx.doi.org/10.1109/ICC.2019.8761147.

[104] M.D. de Assuncao, A. da Silva Veith, R. Buyya, Distributed data stream processing and edge computing: A survey on resource elasticity and future directions, J. Netw. Comput. Appl. 103 (2018) 1–17, http://dx.doi.org/10.1016/j.jnca.2017.12.001.

[105] S.R. Niya, R. Beckmann, B. Stiller, DLIT: a scalable distributed ledger for IoT data, in: 2020 Second International Conference on Blockchain Computing and Applications, BCCA, IEEE, 2020, pp. 100–107, http://dx.doi.org/10.1109/BCCA50787.2020.9274456.

[106] V.K. Sarker, T.N. Gia, H. Tenhunen, T. Westerlund, Lightweight security algorithms for resource-constrained IoT-based sensor nodes, in: ICC 2020-2020 IEEE International Conference on Communications, ICC, IEEE, 2020, pp. 1–7, http://dx.doi.org/10.1109/ICC40277.2020.9149359.

[107] V.A. Thakor, M.A. Razzaque, M.R. Khandaker, Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities, IEEE Access 9 (2021) 28177–28193, http://dx.doi.org/10.1109/ACCESS.2021.3052867.

[108] T.X. Meng, W. Buchanan, Lightweight Cryptographic Algorithms on Resource-Constrained Devices, Researchgate, 2020, http://dx.doi.org/10.20944/preprints202009.0302.v1, Preprints.

[109] S. Rana, S. Hossain, H.I. Shoun, M.A. Kashem, An effective lightweight cryptographic algorithm to secure resource-constrained devices, Int. J. Adv. Comput. Sci. Appl. 9 (11) (2018) 267–275, http://dx.doi.org/10.14569/IJACSA.2018.091137.

[110] H. Xue, D. Chen, N. Zhang, H.-N. Dai, K. Yu, Integration of blockchain and edge computing in internet of things: A survey, Future Gener. Comput. Syst. 144 (2023) 307–326, http://dx.doi.org/10.1016/j.future.2022.10.029.

[111] J. Al-Karaki, D. Pavithran, A. Gawanmeh, Integrating blockchain with fog and edge computing for micropayment systems, in: Security Issues in Fog Computing from 5G To 6G: Architectures, Applications and Solutions, Springer, 2022, pp. 93–112, http://dx.doi.org/10.1007/978-3-031-08254-2_6.

[112] A. Nawaz, J. Peña Queralta, J. Guan, M. Awais, T.N. Gia, A.K. Bashir, H. Kan, T. Westerlund, Edge computing to secure iot data ownership and trade with the ethereum blockchain, Sensors 20 (14) (2020) 3965, http://dx.doi.org/10.3390/s20143965.

[113] M.A. Rahman, M.S. Hossain, G. Loukas, E. Hassanain, S.S. Rahman, M.F. Alhamid, M. Guizani, Blockchain-based mobile edge computing framework for secure therapy applications, IEEE Access 6 (2018) 72469–72478, http://dx.doi.org/10.1109/ACCESS.2018.2881246.

[114] B.W. Nyamtiga, J.C.S. Sicato, S. Rathore, Y. Sung, J.H. Park, Blockchain-based secure storage management with edge computing for IoT, Electronics 8 (8) (2019) 828, http://dx.doi.org/10.3390/electronics8080828.

[115] G. Li, M. Dong, L.T. Yang, K. Ota, J. Wu, J. Li, Preserving edge knowledge sharing among IoT services: A blockchain-based approach, IEEE Trans. Emerg. Top. Comput. Intell. 4 (5) (2020) 653–665, http://dx.doi.org/10.1109/TETCI.2019.2952587.

[116] X. Lin, J. Wu, J. Li, X. Zheng, G. Li, Friend-as-learner: Socially-driven trustworthy and efficient wireless federated edge learning, IEEE Trans. Mob. Comput. 22 (1) (2021) 269–283, http://dx.doi.org/10.1109/TMC.2021.3074816.

[117] M. Vijarania, S. Gupta, A. Agrawal, M.O. Adigun, S.A. Ajagbe, J.B. Awotunde, Energy efficient load-balancing mechanism in integrated IoT–fog–cloud environment, Electronics 12 (11) (2023) 2543, http://dx.doi.org/10.3390/electronics12112543.

[118] Z. Zhao, K.M. Barijough, A. Gerstlauer, Deepthings: Distributed adaptive deep learning inference on resource-constrained iot edge clusters, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 37 (11) (2018) 2348–2359, http://dx.doi.org/10.1109/TCAD.2018.2858384.

[119] M.M.S. Maswood, M.R. Rahman, A.G. Alharbi, D. Medhi, A novel strategy to achieve bandwidth cost reduction and load balancing in a cooperative three-layer fog-cloud computing environment, IEEE Access 8 (2020) 113737–113750, http://dx.doi.org/10.1109/ACCESS.2020.3003263.

[120] W.-C. Chien, C.-F. Lai, H.-H. Cho, H.-C. Chao, A SDN-SFC-based service-oriented load balancing for the IoT applications, J. Netw. Comput. Appl. 114 (2018) 88–97, http://dx.doi.org/10.1016/j.jnca.2018.04.009.

[121] S. Sinche, D. Raposo, N. Armando, A. Rodrigues, F. Boavida, V. Pereira, J.S. Silva, A survey of IoT management protocols and frameworks, IEEE Commun. Surv. Tutor. 22 (2) (2019) 1168–1190, http://dx.doi.org/10.1109/COMST.2019.2943087.

[122] H. Derhamy, J. Eliasson, J. Delsing, P. Priller, A survey of commercial frameworks for the internet of things, in: 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation, Etfa, IEEE, 2015, pp. 1–8, http://dx.doi.org/10.1109/ETFA.2015.7301661.

[123] V.P. Singh, V.T. Dwarakanath, P. Haribabu, N.S.C. Babu, IoT standardization efforts—An analysis, in: 2017 International Conference on Smart Technologies for Smart Nation, SmartTechCon, IEEE, 2017, pp. 1083–1088, http://dx.doi.org/10.1109/SmartTechCon.2017.8358536.

[124] M.A. Jazayeri, S.H. Liang, C.-Y. Huang, Implementation and evaluation of four interoperable open standards for the internet of things, Sensors 15 (9) (2015) 24343–24373, http://dx.doi.org/10.3390/s150924343.

[125] L. Belli, S. Cirani, L. Davoli, A. Gorrieri, M. Mancin, M. Picone, G. Ferrari, Design and deployment of an IoT application-oriented testbed, Computer 48 (9) (2015) 32–40, http://dx.doi.org/10.1109/MC.2015.253.

[126] J. Hao, C. Huang, J. Ni, H. Rong, M. Xian, X.S. Shen, Fine-grained data access control with attribute-hiding policy for cloud-based IoT, Comput. Netw. 153 (2019) 1–10, http://dx.doi.org/10.1016/j.comnet.2019.02.008.

[127] S. Qi, Y. Lu, W. Wei, X. Chen, Efficient data access control with fine-grained data protection in cloud-assisted IIoT, IEEE Internet Things J. 8 (4) (2020) 2886–2899, http://dx.doi.org/10.1109/JIOT.2020.3020979.

[128] K. Sha, T.A. Yang, W. Wei, S. Davari, A survey of edge computing-based designs for IoT security, Digit. Commun. Netw. 6 (2) (2020) 195–202, http://dx.doi.org/10.1016/j.dcan.2019.08.006.

[129] Y. Ding, H. Han, W. Lu, Y. Wang, N. Zhao, X. Wang, X. Yang, DDQN-based trajectory and resource optimization for UAV-aided MEC secure communications, IEEE Trans. Veh. Technol. (2023) 1–6, http://dx.doi.org/10.1109/TVT.2023.3335210.

[130] Y. Xu, B. Li, N. Zhao, Y. Chen, G. Wang, Z. Ding, X. Wang, Coordinated direct and relay transmission with NOMA and network coding in nakagami-m fading channels, IEEE Trans. Commun. 69 (1) (2020) 207–222, http://dx.doi.org/10.1109/TCOMM.2020.3025555.

[131] Z. Zou, Y. Jin, P. Nevalainen, Y. Huan, J. Heikkonen, T. Westerlund, Edge and fog computing enabled AI for IoT-an overview, in: 2019 IEEE International Conference on Artificial Intelligence Circuits and Systems, AICAS, IEEE, 2019, pp. 51–56, http://dx.doi.org/10.1109/AICAS.2019.8771621.

[132] Q. Liang, P. Shenoy, D. Irwin, Ai on the edge: Characterizing ai-based iot applications using specialized edge architectures, in: 2020 IEEE International Symposium on Workload Characterization, IISWC, IEEE, 2020, pp. 145–156, http://dx.doi.org/10.1109/IISWC50251.2020.00023.

[133] A.L. Diedrichs, F. Bromberg, D. Dujovne, K. Brun-Laguna, T. Watteyne, Prediction of frost events using machine learning and IoT sensing devices, IEEE Internet Things J. 5 (6) (2018) 4589–4597, http://dx.doi.org/10.1109/JIOT.2018.2867333.

[134] H. Jafari, O. Omotere, D. Adesina, H.-H. Wu, L. Qian, IoT devices fingerprinting using deep learning, in: MILCOM 2018-2018 IEEE Military Communications Conference, MILCOM, IEEE, 2018, pp. 1–9, http://dx.doi.org/10.1109/MILCOM.2018.8599826.

[135] S. Aneja, N. Aneja, M.S. Islam, IoT device fingerprint using deep learning, in: 2018 IEEE International Conference on Internet of Things and Intelligence System, IOTAIS, IEEE, 2018, pp. 174–179, http://dx.doi.org/10.1109/IOTAIS.2018.8600824.

[136] S. Pokhrel, R. Abbas, B. Aryal, IoT security: botnet detection in IoT using machine learning, 2021, http://dx.doi.org/10.48550/arXiv.2104.02231, arXiv preprint arXiv:2104.02231.

[137] Y. Meidan, M. Bohadana, A. Shabtai, M. Ochoa, N.O. Tippenhauer, J.D. Guarnizo, Y. Elovici, Detection of unauthorized IoT devices using machine learning techniques, 2017, http://dx.doi.org/10.48550/arXiv.1709.04647, arXiv preprint arXiv:1709.04647.

[138] I. Alrashdi, A. Alqazzaz, E. Aloufi, R. Alharthi, M. Zohdy, H. Ming, Ad-iot: Anomaly detection of iot cyberattacks in smart city using machine learning, in: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC, IEEE, 2019, pp. 0305–0310, http://dx.doi.org/10.1109/CCWC.2019.8666450.

[139] P. Zhang, M. Zhou, C. Li, A. Abusorrah, Dynamic evolutionary game-based modeling, analysis and performance enhancement of blockchain channels, IEEE/CAA J. Autom. Sin. 10 (1) (2022) 188–202, http://dx.doi.org/10.1109/JAS.2022.105911.

[140] K. Qian, Y. Liu, X. He, M. Du, S. Zhang, K. Wang, HPCchain: A consortium blockchain system based on CPU-FPGA hybrid PUF for industrial internet of things, IEEE Trans. Ind. Inform. 19 (11) (2023) 11205–11215, http://dx.doi.org/10.1109/TII.2023.3244339.

[141] Z. Yang, G. Li, J. Wu, W. Yang, Propagable backdoors over blockchain-based federated learning via sample-specific eclipse, in: GLOBECOM 2022-2022 IEEE Global Communications Conference, IEEE, 2022, pp. 2579–2584, http://dx.doi.org/10.1109/GLOBECOM48099.2022.10001370.