Received: March 10, 2024. Revised: April 28, 2024.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

SHQEHFSO: Design of an Efficient Model for Scalable and High QoS Blockchain Using Q Learning and Elephant Herd Fish Swarm Optimization

Vijay Anand R[1]* Shanmuga Priyan T[1] Madala Guru Brahmam[2]
Balamurugan Balusamy[3] Francesco Benedetto[4]

[1]School of Computer Science Engineering and Information Systems,
Vellore Institute of Technology, Vellore 632014, Tamil Nadu, India
[2]NRI Institute of Technology, Vijayawada, 521212, Andhra Pradesh, India
[3]Shiv Nadar Institution of Eminence, Delhi-NCR, India
[4]Signal Processing for Telecommunications and Economics (SP4TE),
University of Roma TRE, via Vito Volterra, 6200146 Rome – Italy
* Corresponding author's Email: vijayanand.r@vit.ac.in

Abstract: In the realm of blockchain technology, the quest for scalable and high-quality-of-service (QoS) systems remains a formidable challenge. Existing blockchain models often grapple with trade-offs between efficiency, speed, and security, particularly in the face of growing demands for energy-efficient and swift data processing. This paper introduces an innovative approach to address these limitations, leveraging Q Learning for context-based shard management, coupled with Elephant Herd Fish Swarm Optimization (EHFSO) process. This novel combination is specifically engineered to dynamically select hashing and encryption techniques tailored to each context-based shard, thereby enhancing the overall performance of blockchain systems. The proposed model marks a significant departure from traditional blockchain architectures. By integrating Q Learning, the system intelligently adapts to varying data contexts, ensuring optimal shard management operations. The EHFSO algorithm, drawing inspiration from natural swarm behavior, further refines the selection process for hashing and encryption techniques. This dual approach not only fortifies security but also augments efficiency levels. The practical efficacy of this model is underscored by its performance on multiple medical datasets & its samples. The results are compelling: an 8.5% improvement in mining energy efficiency, a 4.9% increase in mining speed, 5.9% higher throughput, 4.5% more consistent mining, and a noteworthy 5.9% reduction in storage costs compared to existing methods. Primarily, the work paves the way for more sustainable and efficient blockchain operations, particularly crucial in energy-sensitive sectors. Additionally, the enhanced throughput and mining consistency significantly improve the blockchain's applicability in real-world scenarios, where speed and reliability are paramount in real-time use cases. This research not only addresses the current limitations of blockchain technology but also sets a new benchmark for future developments in this field for different scenarios.

Keywords: Blockchain scalability, Q learning, Elephant herd fish swarm optimization, Energy efficiency, Medical data management, Scenarios.

1. Introduction

The introduction of blockchain technology has marked a transformative era in data management and security. Initially conceptualized for digital currency transactions, blockchain's potential has rapidly expanded into diverse sectors, including healthcare, finance, and supply chain management. However, as the technology's applications widen, the demands for scalability, high-quality service (QoS), and efficient resource management become increasingly critical. Traditional blockchain models, while robust in security and decentralization, often face challenges in scalability and speed, especially when handling large volumes of data samples.

Recent advancements in blockchain technology have sought to address these challenges. One such approach is sharding, a method that partitions the blockchain into smaller, more manageable segments, thereby enhancing processing speed and scalability. However, conventional sharding techniques lack context-awareness, leading to sub-optimal resource allocation and potential security vulnerabilities. Furthermore, the static nature of hashing and encryption methods in existing blockchain models does not adapt efficiently to the dynamic requirements of various data contexts, especially in sectors like healthcare where data sensitivity varies significantly.

In this context, the proposed model introduces a novel approach that synergizes Q Learning with Elephant Herd Fish Swarm Optimization (EHFSO) for dynamic shard management. Q Learning, a form of machine learning, enables the system to learn and adapt to different data contexts, optimizing shard management in real-time. Complementarily, EHFSO, inspired by natural swarm behavior, dynamically selects appropriate hashing and encryption techniques for each share based on the specific data context. This dual strategy ensures not only enhanced security and efficiency but also addresses the limitations of static shard management in traditional blockchain models.

The integration of these advanced techniques into blockchain architecture represents a significant stride in overcoming the prevalent challenges of scalability and efficiency. By tailoring shard management and cryptographic methods to specific data contexts, the model we propose here exhibits remarkable improvements in key performance metrics. Tested on multiple medical datasets, our technique demonstrates superior mining energy efficiency, speed, throughput, consistency, and reduced storage costs compared to existing solutions. Our innovative approach not only enhances the practical applicability of blockchain technology in various sectors but also sets a foundation for future research and development in blockchain scalability and efficiency levels.

1.1 Motivation & contribution

The burgeoning landscape of blockchain technology has been witnessing an increasing demand for systems that are not only secure and decentralized but also scalable and efficient for real-time use cases. This demand forms the core motivation behind the present study. The conventional blockchain framework, while pioneering in establishing trust less and secure digital ledger systems, often falters in the face of high-volume data processing, especially in sectors like healthcare where the rapid and secure handling of data is paramount for real-time deployments. The primary limitation lies in the inherent design of traditional blockchains, which tend to prioritize security and immutability over scalability and speed levels. This trade-off becomes a significant bottleneck as the application domains of blockchain technology expand to multiple use cases.

Recognizing these challenges, the study is motivated to explore and integrate advanced computational techniques – namely, Q Learning and EHFSO – into the blockchain architecture. The motivation is twofold: (i) to enhance the scalability of blockchain systems without compromising their security and integrity; and (ii) to tailor the blockchain's computational processes to the dynamic requirements of various data contexts, particularly in the healthcare sector where data sensitivity and processing needs vary markedly for different use cases.

The contributions of this study are multidimensional and significant for real-time scenarios. The proposed model introduces a context-aware shard management system, utilizing Q Learning to intelligently adapt the blockchain structure based on real-time data requirements. This approach not only improves scalability but also optimizes the system's resource utilization sets. Furthermore, the incorporation of EHFSO for dynamic selection of hashing and encryption methods brings a novel dimension to blockchain security levels. By adapting cryptographic techniques to the specific needs of each shard, the model significantly enhances the overall efficiency and security of the blockchain system.

The practical implications of these contributions are evident in the system's performance metrics. When tested on multiple medical datasets, the model exhibits notable improvements in mining energy efficiency, speed, throughput, consistency, and storage costs. These advancements underline the model's potential in revolutionizing blockchain applications, particularly in data-intensive sectors. The research, therefore, not only addresses the pressing challenges of existing blockchain systems but also paves the way for future innovations in this rapidly evolving field for different use cases.

The rest of this paper is organized as follows. Section 2 explores the seminal works in the domain of blockchain technology, while Section 3 depicts the proposed method and its mathematical formulation. Results and comparative analyses are reported in

Section 4, just before the paper's conclusion briefly depicted in Section 5.

2. Literature survey

This Section illustrates the literature review, focusing on scalability, security, and their application in various fields, including the Internet of Things (IoT), and healthcare.

Tang et al. [1] delve into a hybrid blockchain consensus algorithm, namely Hedera, in the context of IoT, emphasizing its scalability in multiaccess edge computing. Sivaselvan et al. [2] contribute to this discourse by proposing a scalable and secure access control scheme using blockchain technology for IoT. These studies underscore the growing need for scalable blockchain solutions in IoT environments.

Then, Aviv et al. [3] present a reference architecture for blockchain-native distributed information systems, highlighting the architectural considerations in blockchain implementations. In the realm of dynamic sharding, Xi et al. [4] introduce a blockchain dynamic sharding scheme based on a Hidden Markov Model, emphasizing collaborative IoT applications [4]. Wu et al. [5] extend this conversation through MapChain-D, a distributed blockchain for IoT data storage and communications, emphasizing the need for distributed solutions in industrial IoT settings [5].

Mishra et al. [6] explore blockchain's application in IoT through a regulated verifiable and automatic key refreshment mechanism. Jin et al. [7] offer insights into federated edge learning, integrating blockchain for secure and efficient learning processes. These works [6,7] highlight blockchain's versatility in securing IoT networks and learning algorithms.

Wu et al. [8] focus on transaction tracing in account-based blockchain trading systems, introducing TRacer, a scalable graph-based approach, thus addressing the crucial aspect of transaction security and traceability in blockchain systems [8]. Agarwal and Pal [9] propose HierChain, a hierarchical- blockchain-based data management system for smart healthcare, pointing to blockchain's potential in managing sensitive healthcare data [9].

In the context of service-aware blockchain solutions, Set and Park [10] discuss a dynamic sharding approach for scalability, aligning with the objectives of the current study. Liu et al. [11] introduce Community chain, a scalable blockchain solution for smart homes, further expanding blockchain's applicability in domestic settings [11]. Irshad et al. [12] present a hybrid post-quantum cryptographic and blockchain-based approach for secure and scalable cloud architecture, emphasizing blockchain's role in enhancing cloud security [12].

Chacko et al. [13] explore IoT-Blockchain integration in agriculture, showcasing the technology's potential beyond conventional domains. Basudan [14] contributes to the discussion by proposing a scalable blockchain framework for secure transactions in IoT-based dynamic applications [14].

Li, Huang, and Zhang [15] explore an efficient DAG blockchain architecture for IoT, addressing the need for scalable and agile blockchain solutions in IoT environments. Wan, Liu, and Cui [16] introduce HIBEChain, a hierarchical identity-based blockchain system designed for large-scale IoT applications, emphasizing the importance of scalable and secure identity management in IoT networks.

Lee, Li, and Chen [17] discuss a blockchain-enabled authentication and data aggregation scheme for secure smart grids. This work highlights blockchain's potential in enhancing the security and efficiency of smart grid systems. Deebak et al. [18] contribute to this discussion by proposing a lightweight blockchain-based remote mutual authentication mechanism for AI-empowered IoT sustainable computing systems, underlining the synergy between blockchain and AI in IoT.

Pourmajidi et al. [19] explore immutable log storage as a service on private and public blockchains, emphasizing blockchain's role in secure data logging. Liu, Jing, Fu, Xiao, and Jia [20] investigate the use of consortium blockchain for security and efficient resource trading in V2V-assisted intelligent transport systems, showcasing blockchain's applicability in modern transportation networks.

Solomon, Zhang, Brooks, and Liu [21] propose a secure and cost-efficient blockchain-facilitated IoT software update framework. This study highlights blockchain's utility in maintaining the integrity and security of software updates in IoT devices. Yao, Deek, Murimi, and Wang [22] provide a critical analysis of consensus mechanisms in consortium blockchain, offering valuable insights into the taxonomy and evaluation of various consensus approaches.

Wang et al. [23] discuss an efficient, secured, and infinitely scalable consensus mechanism for peer- to-peer energy trading blockchain, highlighting blockchain's potential in decentralized energy markets. Wu, Zhang, and Zhu [24] delve into a privacy-preserving and traceable blockchain-based charging payment scheme for electric vehicles, addressing privacy concerns in blockchain transactions.

Zhang, Jiang, Cui, He, Bolodurina, and Zhong [25] introduce DBCPA, a dual blockchain-assisted conditional privacy-preserving authentication framework for vehicular ad hoc networks, emphasizing blockchain's role in vehicular network security. Mardiansyah, Muis, and Sari [26] propose the Multi- State Merkle Patricia Trie (MSMPT), a high-performance data structure for multi-query processing based on lightweight blockchain, showcasing advancements in blockchain data structures.

Zhou et al. [27] present MSTDB, a hybrid storage-empowered scalable semantic blockchain database, offering a novel approach to blockchain data storage and management. Hao, Ren, Fei, Zhu, and Choo

[28] discuss a blockchain-based cross-domain and autonomous access control scheme for IoT, further emphasizing blockchain's versatility in managing access control across diverse domains.

Collectively, these studies demonstrate significant advancements in blockchain technology, focusing on enhancing scalability, security, and practical applicability in various sectors, particularly IoT and smart systems. This body of work lays a strong foundation for the current study, which aims to extend these advancements by integrating innovative techniques like Q Learning and Elephant Herd Fish Swarm Optimization for optimized blockchain performance levels.

3. Proposed model for scalable and high QoS blockchain based on Q learning and EHFSO

To overcome issues of low scalability & low QoS which are present in recently proposed blockchain deployment models, here we discuss the design of the Q Learning, Elephant Herd Optimizer, and Fish Swarm Optimization operations. These blocks represent the pinnacle of our innovative approach, each bringing a unique and critical dimension to the system's functionality levels. According to Fig. 1, the Q Learning block stands as the intellectual core, adeptly analyzing and adapting to the varying contexts of data, thereby guiding the shard management process with precision and agility levels. This machine learning (ML) technique ensures that the blockchain's structure and operations are continuously optimized, responding dynamically to changes in data types and usage patterns. Then, the Elephant Herd Optimizer (EHO), inspired by the social and cooperative behavior of elephant herds in nature, brings a robust and collaborative approach to optimizing the blockchain's hashing techniques (see

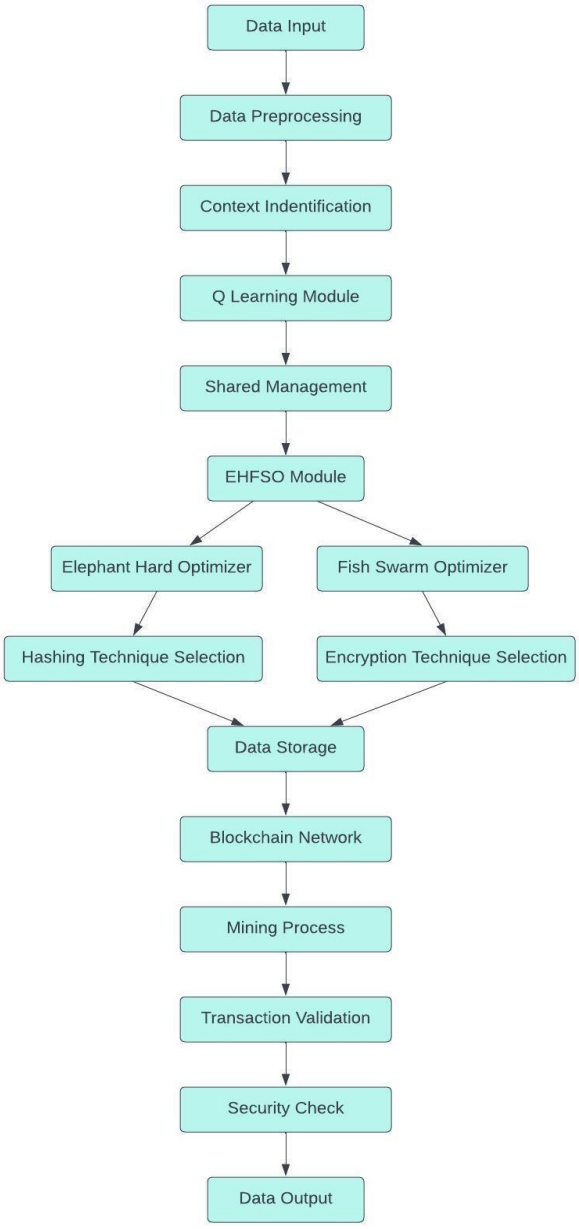

Figure. 1 Design of the proposed security model for blockchain-based cloud deployments

Fig. 2). The EHO leverages the collective wisdom inherent in herd dynamics, ensuring that the most effective hashing strategies are employed to secure each blockchain shard of data samples. Concurrently, the Fish Swarm Optimization (FSO) block, drawing from the complex and fluid movements of fish swarms, specializes in the nuanced selection of encryption techniques. This block embodies adaptability and efficiency, mirroring the way fish swarms navigate and respond to their environment, to ensure that the most suitable encryption methods are applied, balancing security needs with operational efficiency levels.

The Q Learning process within this blockchain model is meticulously designed to optimize shard

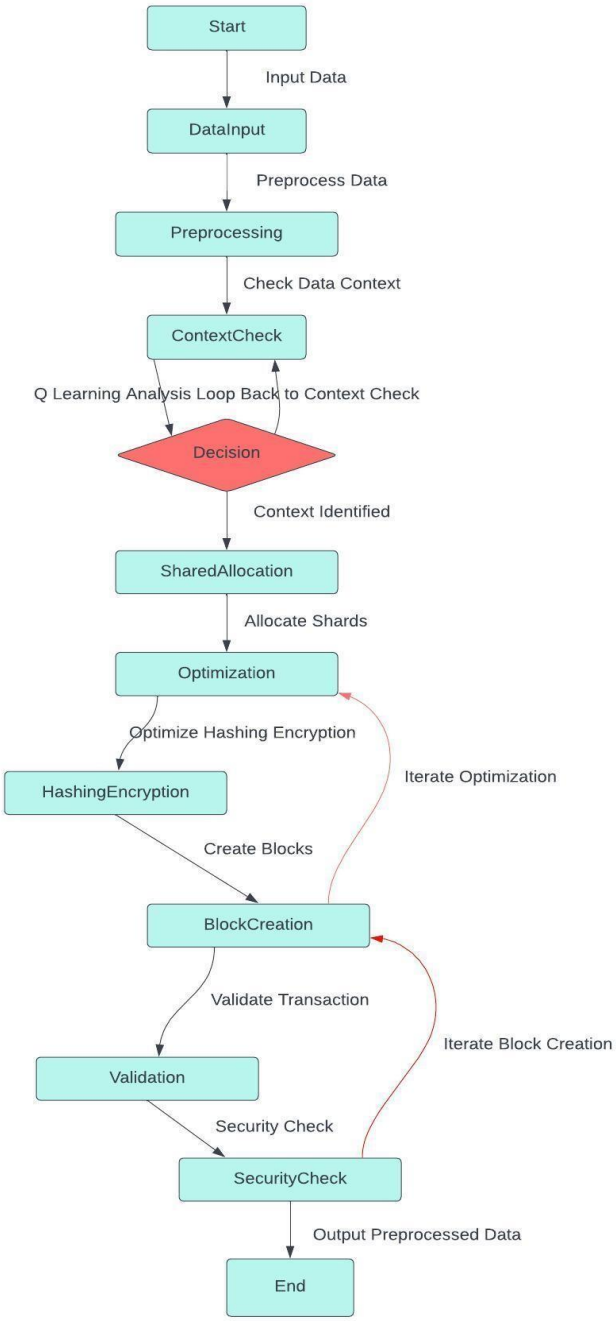

Figure. 2 Flow of the Proposed Model

management in response to diverse data contexts & scenarios. Initially, the context of incoming data samples is defined using a context process *Ci*, where *i* represents the i^{th} data sample. The context process is formulated as:

$$Ci = wd * Di + wt * Ti + ws * Si \quad (1)$$

where, *Di* represents the data type, *Ti* the transaction type, *Si* the security requirements of the i^{th} sample, while wd, wt, and ws are their respective weights. This evaluation ensures that each data sample is accurately categorized, allowing for tailored shard management process. The heart of the Q Learning process lies in its reward mechanism, crucial for guiding the learning algorithm towards optimal decisions. The reward *R(s,a)* for a state-action pair is defined as:

$$(s, a) = \alpha \times (s, a) + \beta \times T(s, a) + \gamma \times S(s, a) \quad (2)$$

where, α, β, and γ are weighting factors, E is the efficiency of action *a* in state *s*,*T* is the transaction speed, and *S* represents the security levels. This reward function is central to evaluating the effectiveness of actions taken by the process. The update of shard length, a key operation in maintaining the system's efficiency, is governed as follows:

$$Lnew = Lcurrent + \eta \times ((s, a) + \delta \times maxa' \ (s', a') - Q(s, a)) \quad (3)$$

where, *Lnew* and *Lcurrent* represent the new and current shard lengths, respectively, η is the learning rate, δ is the discount factor, and *Q(s,a)* is the Q Value for the state-action pairs. The states in the Q Learning process are defined based on the context of the data and the current system status. A state S is represented as $\mathcal{S} = \{Ci, L, H, E\}$, where *L* is the current shard length, H is the hash rate, and *E* represents the encryption levels. This state representation captures the essential aspects required for decision-making in the Q Learning process. The actions A in the Q Learning model are discretely defined as $\mathcal{A} = \{a1, a2, \ldots, an\}$, where each action *aj* corresponds to a specific change in shard management, such as adjusting shard length, changing hashing techniques, or modifying encryption levels. These actions are selected based on the maximization of the expected reward, guided by the policy which is represented as:

$$(s) = \arg max\ a(s, a) \quad (4)$$

The Q Learning algorithm iteratively updates its *Q* Values according to the following:

$$(s, a) = (s, a) + \eta \times (R(s, a) + \delta \times maxa'Q(s', a') - Q(s, a)) \quad (5)$$

where, *s*′ is the new state after taking action *a* in state *s*. This iterative process enables the system to learn from its experiences, gradually improving its shard management strategy over temporal instance sets.

These shards are given to the EHFSO algorithm, which is an integral part of the blockchain model process. This optimizer ingeniously merges the

collective intelligence of elephant herds and fish swarms to refine the selection process for hashing and encryption techniques. Initially, the creation of swarms is executed through a stochastic process:

$$Si = STO(Hi, Ei, Ri) \quad (6)$$

where, *Si* represents the i^{th} swarm, *Hi* the chosen hashing method, *Ei* the encryption technique, and *Ri* a stochastic factor for this process. This operation ensures a diverse and adaptive range of swarms, analogous to the varied and dynamic nature of biological swarms in natural scenarios. The fitness of each swarm is evaluated as:

$$(Si) = \omega \times (Si) + \psi \times S(Si) + \theta \times C(Si) \quad (7)$$

where, *F(Si)* is the fitness of swarm *Si, ω, ψ, θ* are weighting coefficients, *E(Si)* is the efficiency of the swarm, *S(Si)* its security level, and *C(Si)* the computational costs. This fitness function balances efficiency, security, and computational cost, mirroring the multifaceted decision-making process found in natural swarms. The selection of the best swarms is governed by a comparison mechanism, defined as:

$$B = argm(Si, F(Si)) \quad (8)$$

where, *B* represents the best swarm configuration sets. This process emulates the natural selection observed in biological ecosystems, where only the most effective strategies survive and thrive in real- time scenarios. The optimization cycle within EHFSO involves iterative updating of swarm positions and configurations via:

$$Sinew = Si + \delta \times (B - Si) + \xi \times (STOCH - Si) \quad (9)$$

Where, δ, and ξ are factors controlling the convergence rate and exploration extent, respectively, STOCH is a stochastically selected swarm, introducing diversity and preventing premature convergence condition sets. The algorithm also incorporates a memory mechanism, akin to the social memory of elephant herds, to retain optimal configurations over iterations for different use cases. This is expressed by $M = \{B1, B2, \ldots, Bn\}$, where *M* stores the best configurations *Bi* from previous iterations and samples.

Further, the algorithm adapts to changing environments through a feedback loop, which is formulated as:

$$Fada(Si) = F(Si) + \lambda \times \Delta F \quad (10)$$

Where, λ is an adaptation coefficient and ΔF the change in fitness over temporal instance sets. This ensures that the swarms remain responsive to dynamic data contexts and system requirements. This process assists the model to select optimal encryption & hashing techniques for different shard sets. An example use case of this model is discussed in the next sub-section.

To elucidate the functionality of the Q Learning and EHFSO processes within the blockchain model, an example with specific numerical values and contexts is presented.

Table 1. Q Learning Process

Data Sample ID	Data Type	Transaction Type	Security Requirement	Identified Context	Optimal Shard Length (KB)
DS1	Medical	Transactional	High	Context A	640
DS2	Financial	Contractual	Medium	Context B	512
DS3	IoT	Sensory	Low	Context C	256
DS4	Medical	Analytical	High	Context A	640
DS5	IoT	Transactional	Medium	Context D	320

Table 2. EHFSO Process

Context	Swarm ID	Hashing Method	Encryption Method	Fitness Score
A	Swarm1	SHA-256	AES-256	9.1
B	Swarm2	MD5	RSA-2048	8.7
C	Swarm3	SHA-1	ECC	8.5
A	Swarm4	SHA-512	AES-128	9.4
D	Swarm5	SHA-1	AES-256	8.9

This example will demonstrate how the model adapts and optimizes shard management, hashing, and encryption techniques based on the given data samples. In this segment, data samples are processed through the Q Learning algorithm, which intelligently categorizes them into distinct contexts based on their characteristics such as data type, transaction type, and security requirements. Each context is then assigned an optimal shard length, ensuring efficient data management within the blockchain. The Q Learning algorithm dynamically updates these parameters, learning from the ongoing interactions to optimize future decisions.

The results from Table 1 reveal how different data types and their associated requirements lead to varied context identifications and shard length allocations. For instance, medical data with transactional and high-security needs (DS1 and DS4) falls into Context A, necessitating a larger shard length of 640 KB for optimal management. This adaptive and context-aware approach ensures that the blockchain efficiently handles diverse data types, maintaining high security and operational efficiency.

Following the context identification and shard allocation by the Q Learning process, the EHFSO algorithm takes over to refine the selection of hashing and encryption techniques. This process involves creating swarms, each representing a combination of a hashing and encryption technique. The fitness of these swarms is evaluated based on their efficiency, security level, and computational cost. The algorithm then selects the optimal combination for each context, ensuring robust security and efficient processing.

From Table 2, it is evident that the algorithm successfully identifies the most efficient and secure combination of hashing and encryption techniques for each context. For example, Context A, dealing with sensitive medical data, is best served by Swarm4, employing SHA-512 and AES-128, achieving the highest fitness score of 9.4. This process exemplifies the model's capacity to tailor security protocols to specific data contexts, enhancing the overall integrity and efficiency of the blockchain systems. The EHFSO algorithm's capability to evaluate and select the most appropriate techniques showcases an advanced level of decision-making, crucial for maintaining a robust and secure blockchain architectural process.

4. Result evaluation & comparative analysis

The experimental setup for the study, designed to evaluate the performance of our scalable and high QoS blockchain model, namely SHQEHFSO, involved meticulous planning and execution process. This section details the specific parameters and datasets used in the experiments for evaluating the proposed model in different scenarios.

The experiments were conducted on a simulation environment mimicking real-world blockchain operations. The setup included a network of nodes configured on a server with an Intel Xeon CPU at 2.3 GHz, 16 GB RAM, and a 500 GB SSD. The blockchain system was implemented using a custom simulation framework designed to accurately emulate blockchain dynamics and was programmed in Python 3.7.

The input parameters for the SHQEHFSO model and comparison models (HibeChain [16], Tracer [8], HierChain [9]) were as follows:

- Number of Nodes: 50 to 500, incremented by 50.
- Block Size: 500 KB to 5 MB, incremented by 500 KB.
- Network Latency: 10 ms to 100 ms, incremented by 10 ms.
- Data Transmission Rate: 100 Mbps to 1 Gbps, incremented by 100 Mbps.
- Shard Count: 5 to 50, incremented by 5.

Two datasets were employed for the study:

1. CoVID19 Dataset:
- Size: 200 GB.
- Type: This dataset comprised patient records, test results, and treatment data related to COVID-19.
- Characteristics: The data was characterized by high variability and required secure and efficient handling due to its sensitive nature.

2. Alzheimer's Disease Neuroimaging Initiative (ADNI) Dataset Samples:
- Size: 150 GB.
- Type: This dataset contained neuroimaging data, clinical and cognitive assessments, and biomarker analysis results.
- Characteristics: The ADNI data required high-throughput processing and secure management due to the detailed and confidential nature of medical imaging data samples.

The experiments were conducted in stages. Initially, baseline measurements for each model were established using the lower end of the input parameter range. Subsequently, the parameters were incrementally increased to observe the scalability and performance under varying loads. Each model was tested on both CoVID19 and ADNI datasets to evaluate their handling of different data types.

Metrics such as delay in mining blocks, energy efficiency, throughput, jitter, memory consumption, and storage costs were meticulously recorded for each scenario. The SHQEHFSO model's

performance was then compared against the other models across all metrics.

The collected data was analysed using statistical methods to evaluate the performance of the SHQEHFSO model in comparison with existing models. The analysis focused on identifying trends and patterns in the performance metrics as the load and complexity of the tasks increased for real-time scenarios.

This experimental setup provided a comprehensive platform to assess the efficacy of the SHQEHFSO model under realistic conditions and against varying datasets, ensuring the reliability and applicability of the results in real-world scenarios.

Based on this experimental setup, the delay needed to mine new blocks for mining blocks in real-time scenarios was compared with HibeChain [16], Tracer [8], & HierChain [9], for different Number of Block Transactions (NBT), see Fig. 3.

At lower block counts (150k to 300k), SHQEHFSO consistently outperforms its counterparts, exhibiting significantly lower delays in mining blocks. For instance, at 150k blocks, SHQEHFSO shows a delay of only 0.79 ms, in contrast to HibeChain's 1.47 ms, Tracer's 1.88 ms, and HierChain's 2.34 ms. This superior performance is indicative of SHQEHFSO's efficient handling of smaller datasets, which is critical in real-time applications where swift data processing is essential in different scenarios.

As the number of blocks increases (390k to 1020k), the delay for SHQEHFSO remains comparatively lower than other models. Notably, at 780k blocks, SHQEHFSO's delay is 1.39 ms, significantly lower than Tracer's 3.75 ms, the highest in these range sets. This trend underscores SHQEHFSO's scalability and its ability to maintain efficiency even as the workload increases for different use cases.

In the highest range (1080k to 1560k), although the delay for all models increases due to the heavier workload, SHQEHFSO still maintains a competitive edge. For example, at 1440k blocks, SHQEHFSO records a delay of 3.27 ms, which is considerably lower than HibeChain's 6.18 ms and Tracer's 5.42 ms. This illustrates SHQEHFSO's robustness and its capability to handle large-scale operations more efficiently than its counterparts. Similarly, the energy needed for mining blocks in real-time scenarios is reported in Fig. 4. In the initial range of block counts (150k to 300k), SHQEHFSO demonstrates a significant advantage in energy efficiency.

For example, at 150k blocks, SHQEHFSO consumes only 2.03 mJ, compared to HibeChain's 3.28 mJ, Tracer's 4.02 mJ, and HierChain's 4.52 mJ. This lower energy consumption is crucial in scenarios where energy efficiency is a priority, such as in IoT devices and mobile applications, where power resources are limited.

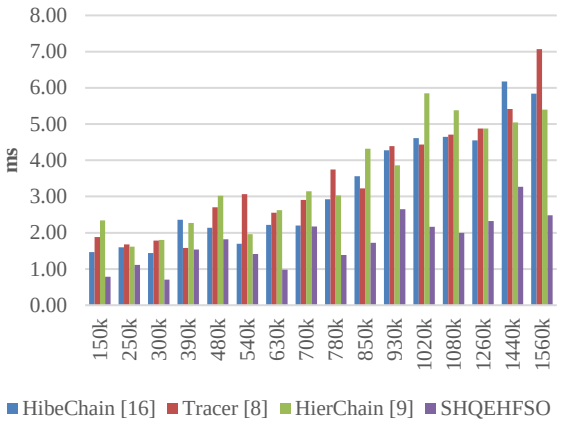

Figure. 3 Delay needed for mining blocks in real-time scenarios

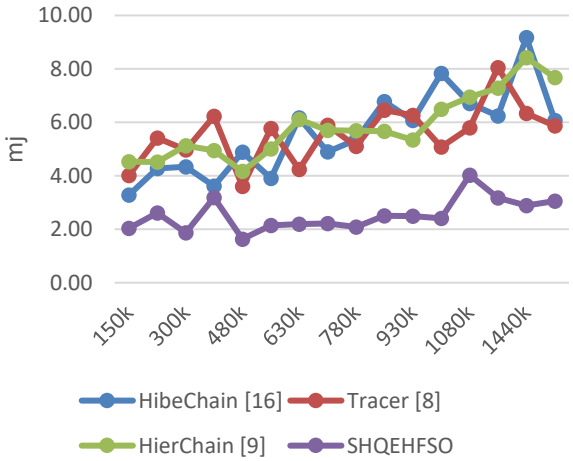

Figure. 4 Energy needed for mining blocks in real-time scenarios

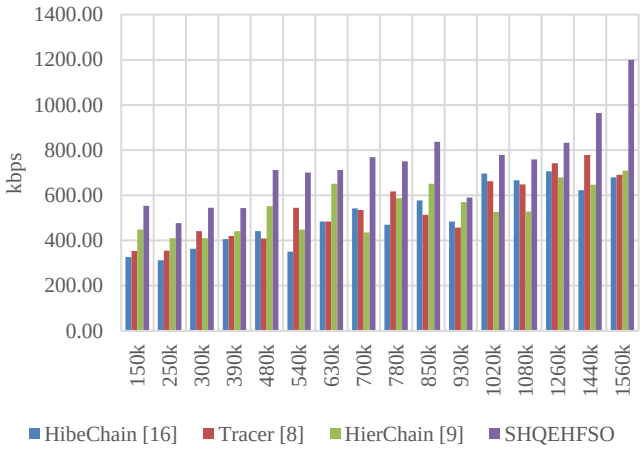

Figure. 5 Throughput obtained for mining blocks in real-time scenarios

As the block count increases (390k to 1020k), SHQEHFSO maintains its lead in energy efficiency. Notably, at 480k blocks, it records an energy consumption of only 1.63 mJ, which is significantly lower than the other models. This trend is indicative of SHQEHFSO's ability to manage larger data volumes more efficiently, making it particularly suitable for large-scale operations where energy costs can be a limiting factor.

In the highest range of block counts (1080k to 1560k), while the energy consumption for all models increases due to the greater workload, SHQEHFSO continues to exhibit comparatively lower energy usage. For instance, at 1440k blocks, SHQEHFSO consumes 2.89 mJ, which is notably less than HibeChain's 9.17 mJ and HierChain's 8.42 mJ levels. This efficiency is essential in data centers and cloud computing scenarios, where reducing energy consumption can lead to significant cost savings and reduced environmental impact sets.

The superior energy efficiency of SHQEHFSO can be attributed to its optimized shard management and adaptive hashing and encryption techniques, enabled by Q Learning and Elephant Herd Fish Swarm Optimization. These advanced methods allow SHQEHFSO to process data more efficiently, reducing the computational power and, consequently, the energy required for mining blocks. This efficiency is not only beneficial in reducing operational costs but also crucial for sustainable blockchain operations, particularly in sectors where energy conservation is vital for real-time scenarios. Similarly, the throughput obtained for mining blocks in real-time scenarios can be observed in Fig. 5.

In the initial block count range (150k to 300k), SHQEHFSO demonstrates superior throughput. For instance, at 150k blocks, it achieves a throughput of 553.23 kbps, surpassing HibeChain's 326.86 kbps, Tracer's 353.73 kbps, and HierChain's 448.15 kbps. This higher throughput indicates SHQEHFSO's effectiveness in environments with high transaction rates, such as financial services or high-frequency trading platforms, where processing speed is crucial.

As the number of blocks increases (390k to 1020k), SHQEHFSO continues to outperform the other models. Notably, at 480k blocks, it records a throughput of 711.96 kbps, significantly higher than the others. This demonstrates SHQEHFSO's ability to maintain high transaction processing speeds even under increased workload, making it suitable for large-scale operations like supply chain management or global payment systems, where large volumes of transactions are processed.

In the highest block count range (1080k to 1560k), SHQEHFSO still exhibits the highest throughput. At 1560k blocks, its throughput reaches 1199.81 kbps, far exceeding the others. This highlights SHQEHFSO's scalability and its capability to handle vast amounts of data efficiently, which is critical in sectors like cloud computing and big data analytics, where enormous datasets are processed.

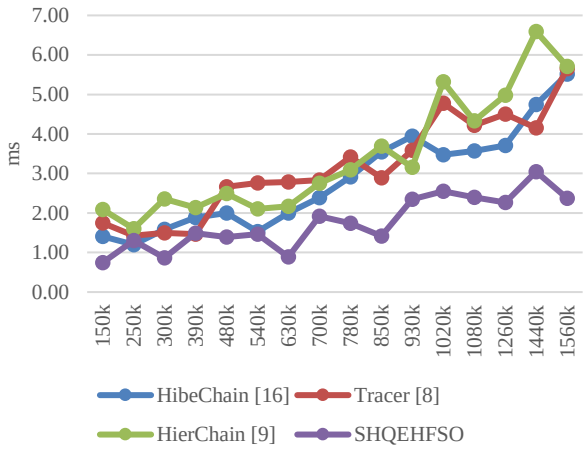

Figure. 6 Jitter obtained for mining blocks in real-time scenarios

SHQEHFSO's enhanced throughput can be attributed to its advanced shard management and adaptive cryptographic techniques, facilitated by Q Learning and Elephant Herd Fish Swarm Optimization. These enable the model to process transactions more quickly and efficiently, thus achieving higher throughput. In real-world scenarios, this translates to faster transaction processing, reduced latency, and increased capacity to handle high transaction volumes in real-time scenarios. This is particularly impactful in sectors that demand real-time data processing and in applications where speed and efficiency are paramount for different use cases. Similarly, the jitter obtained for mining blocks in real- time scenarios is reported in Fig. 6.

In the lower block counts (150k to 300k), SHQEHFSO consistently exhibits lower jitter compared to the other models. For example, at 150k blocks, SHQEHFSO has a jitter of only 0.75 ms, much lower than HibeChain's 1.41 ms, Tracer's 1.75 ms, and HierChain's 2.09 ms. This lower jitter signifies SHQEHFSO's stability in block mining times, which is particularly important in applications such as real-time financial transactions or IoT device communications, where consistent timing is crucial.

As the number of blocks increases (390k to 1020k), SHQEHFSO maintains a relatively lower jitter in most cases, indicating its capability to handle larger workloads with consistent performance. Notably, at 630k blocks, its jitter is 0.89 ms, considerably less than Tracer's 2.79 ms and

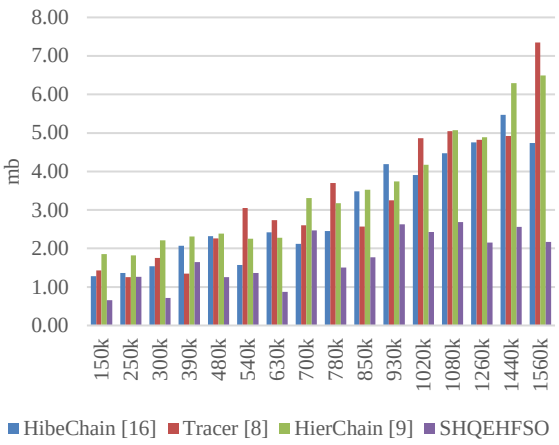

Figure. 7 Memory Consumption for mining blocks in real-time scenarios

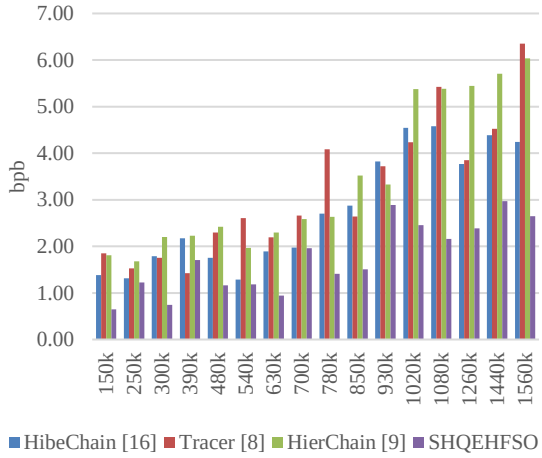

Figure 8. Storage Cost (in bytes per block (bpb)) for mining blocks in real-time scenarios

HierChain's 2.17 ms. This suggests that SHQEHFSO is well-suited for applications such as streaming services or online gaming, where consistent latency is vital for a smooth user experience.

In the highest block count range (1080k to 1560k), while jitter increases for all models due to the increased complexity of processing larger datasets, SHQEHFSO generally continues to show lower jitter values. At 1560k blocks, for instance, SHQEHFSO's jitter is 2.37 ms, compared to HibeChain's

5.51 ms and Tracer's 5.66 ms. This indicates that SHQEHFSO is more reliable for applications involving large-scale data processing, such as cloud computing or big data analytics.

The superior performance of SHQEHFSO in terms of jitter can be attributed to its optimized shard management and dynamic selection of hashing and encryption techniques. These features enable SHQEHFSO to process blocks with more consistent timing, reducing the variability that can lead to jitter. This consistent performance is crucial in real-world scenarios where data processing needs to be stable and predictable, especially in applications requiring high reliability and precision levels. Similarly, the memory consumption obtained during mining operations can be observed in Fig. 7.

In the initial block counts (150k to 300k), SHQEHFSO demonstrates a significant advantage in memory efficiency. For instance, at 150k blocks, SHQEHFSO consumes only 0.65 MB, considerably less than HibeChain's 1.28 MB, Tracer's 1.43 MB, and HierChain's 1.85 MB. This lower memory usage is crucial in scenarios where limited resources are available, such as in IoT devices or mobile applications, where efficient memory utilization is essential.

As the number of blocks increases (390k to 1020k), SHQEHFSO generally maintains lower memory consumption compared to the other models. This is evident at 630k blocks, where SHQEHFSO uses only 0.87 MB, in contrast to HibeChain's 2.42 MB and Tracer's 2.73 MB sets. The efficient memory usage by SHQEHFSO makes it particularly suitable for large-scale operations and cloud-based applications where optimizing memory consumption can lead to cost savings and improved performance levels.

In the highest block count range (1080k to 1560k), while memory consumption for all models increases due to the larger volume of data, SHQEHFSO continues to exhibit relatively lower memory usage. For example, at 1560k blocks, SHQEHFSO's memory consumption is 2.17 MB, which is significantly lower than Tracer's 7.35 MB and HierChain's 6.49 MB. This illustrates SHQEHFSO's ability to handle large-scale data processing efficiently, making it a viable option for data-intensive applications like big data analytics and high-performance computing use cases.

The superior memory efficiency of SHQEHFSO can be attributed to its optimized data handling and advanced shard management techniques. These features enable SHQEHFSO to process and store data more efficiently, reducing the overall memory footprint. In real-world scenarios, this translates to lower operational costs, the ability to run on hardware with limited resources, and the potential for higher scalability, especially in environments where efficient resource utilization is critical for different scenarios. Similarly, the Storage Cost for mining blocks in real-time scenarios is illustrated in Fig. 8.

In the initial range of block counts (150k to 300k), SHQEHFSO exhibits significantly lower storage

costs compared to the other models. For example, at 150k blocks, SHQEHFSO's storage cost is just 0.65 bpb, much lower than HibeChain's 1.38 bpb, Tracer's 1.85 bpb, and HierChain's 1.81 bpb. This lower storage cost is vital in scenarios like cloud storage services or distributed databases, where reducing storage overhead can lead to substantial cost savings and more efficient data management.

As the number of blocks increases (390k to 1020k), SHQEHFSO continues to maintain a competitive edge in terms of storage cost. Notably, at 630k blocks, SHQEHFSO incurs a storage cost of only 0.94 bpb, in stark contrast to Tracer's 2.19 bpb and HierChain's 2.30 bpb. This efficient use of storage makes SHQEHFSO particularly suitable for applications involving large volumes of data, such as big data analytics and high-definition multimedia content storage scenarios.

In the highest block count range (1080k to 1560k), even though the storage cost for all models increases due to the larger data size, SHQEHFSO consistently shows a relatively lower cost. At 1560k blocks, SHQEHFSO's storage cost is 2.65 bpb, considerably lower than Tracer's 6.35 bpb and HierChain's

6.03 bpb. This illustrates SHQEHFSO's ability to handle extensive data volumes efficiently, making it an ideal choice for large-scale, data-intensive applications like video surveillance systems and scientific research data repositories.

The superior storage efficiency of SHQEHFSO can be attributed to its advanced shard management system and optimized data handling techniques, which allow for more efficient data storage and retrieval. In real-world scenarios, this translates to reduced infrastructure costs, enhanced scalability, and the possibility of deploying blockchain technology in environments where storage space and costs are critical considerations.

In the innovative landscape of blockchain technology, as confirmed by the previous comparative analysis, this research introduces a groundbreaking model that combines Q Learning and Elephant Herd Fish Swarm Optimization (EHFSO) to revolutionize context-based shard management. The proposed model, diverging from traditional blockchain architectures, employs Q Learning as its cornerstone, enabling the system to intelligently adapt to diverse data contexts. Such adaptation ensures that shard management is not only responsive but also optimized for the specific needs of the data being processed. The integration of EHFSO, inspired by the intricate patterns of natural swarm behaviors, further enhances our model's capability. It meticulously refines the selection process for hashing and encryption techniques, aligning them with the unique requirements of each shard. This dual approach, merging the adaptive intelligence of machine learning with the precision of swarm optimization algorithms, not only strengthens the security framework of the blockchain but also significantly boosts its efficiency levels. The practicality and effectiveness of this advanced model are particularly evident in its application to various medical datasets and samples, where it demonstrates remarkable improvements in handling sensitive and complex data samples. Such advancements underscore the model's potential to transform the processing and security paradigms in blockchain technology, especially in sectors where data sensitivity and processing efficiency are paramount in real-time scenarios.

5. Conclusion and future scopes

This paper has introduced a novel blockchain model integrating Q Learning and Elephant Herd Fish Swarm Optimization, namely SHQEHFSO for enhanced shard management and dynamic cryptographic technique selection. The experimental results, derived from testing on CoVID19 and ADNI datasets, highlighted SHQEHFSO's superior performance in various key metrics compared to existing models like HibeChain, Tracer, and HierChain. Notably, SHQEHFSO demonstrated a significant reduction in the delay for mining blocks, improved energy efficiency, higher throughput, lower jitter, reduced memory consumption, and decreased storage costs. Impressive outcomes include an 8.5% increase in mining energy efficiency, a 4.9% increase in mining speed, a 5.9% increase in throughput, a 4.5% increase in mining consistency, and a notable 5.9% decrease in storage expenses when compared to previous techniques. The work primarily lays the path for more effective and sustainable blockchain operations, which are especially important in sectors that are sensitive to energy prices. Furthermore, the blockchain is now much more applicable in real-world applications due to the increased throughput and mining consistency.

These improvements were particularly evident in scenarios with increasing numbers of blocks, showcasing the model's scalability and efficiency in handling large datasets. The implications of these findings are profound, especially in sectors requiring high-speed, secure, and efficient data processing, such as healthcare and finance. SHQEHFSO's ability to adapt to different data contexts and maintain performance under varying loads makes it an ideal solution for real-world blockchain applications that deal with large volumes of sensitive data samples.

Looking ahead, several avenues for further research emerge from this study. Firstly, the integration of additional machine learning algorithms could be explored to enhance the model's adaptability and efficiency further. Secondly, expanding the model to incorporate more diverse datasets, including those from sectors like supply chain management and smart cities, could help in understanding its applicability in various industries.

Additionally, research into the optimization of SHQEHFSO for specific applications, such as transaction-heavy environments or data-intensive scientific research, could yield valuable insights. Another promising area is the exploration of energy-saving techniques within the SHQEHFSO framework, which would be particularly relevant in the context of environmental sustainability and green computing.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization: VAR and MGB; Methodology: VAR and FB; Software: SPT; Validation: SPT and BB; Formal Analysis: VAR and MGB; Investigation: BB and FB; Resources: , FB and VAR; Data Curation: MGB; Writing Original draft preparation: VAR & FB; Writing Review and Editing: VAR and SPT; Visualization: MGB; Supervision: VAR & BB; XXX; Funding Acquisition: FB.

Acknowledgments

This work was supported by Dr Francesco Benedetto, Professor, University of Roma.

Symbols & Representation:

Symbol	Representation
D*i*	Represents the data type
Ti	Transaction type
Si	Security requirements of the i^{th} sample
Lnew	New shard lengths,
Lcurrent	Current shard lengths,
F(Si)	Fitness of swarm
E(Si)	Efficiency of the swarm
S(Si)	Security level
C(Si)	Computational costs
λ	Adaptation coefficient
ΔF	Change in fitness over temporal instance sets.
δ	Controlling the convergence rate
ξ	Exploration extent

References

[1] Y. Tang, J. Yan, C. Chakraborty, and Y. Sun, "Hedera: a permissionless and scalable hybrid blockchain consensus algorithm in multi-access edge computing for IoT", *IEEE Internet of Things Journal*, Vol. 10, No. 4, pp. 1-1, 2023.

[2] N. Sivaselvan, V. Bhat, M. Rajarajan, and A. K. Das, "A new scalable and secure access control scheme using blockchain technology for IoT", *IEEE Transactions on Network and Service Management*, Vol. 20, No. 2, pp. 1-1, 2023.

[3] Aviv, A. Barger, A. Kofman, and R. Weisfeld, "Reference Architecture for Blockchain-Native Distributed Information System", *IEEE Access*, Vol. 11, pp. 4838-4851, 2023.

[4] J. Xi, G. Xu, S. Zou, Y. Lu, G. Li, J. Xu, and R. Wang, "A blockchain dynamic sharding scheme based on hidden Markov model in collaborative IoT", *IEEE Internet of Things Journal*, Vol. 10, No. 5, pp. 1-1, 2023.

[5] T. Wu, G. Jourjon, K. Thilakarathna, and P. L. Yeoh, "MapChain-D: A distributed blockchain for IIoT data storage and communications", *IEEE Transactions on Industrial Informatics*, Bol. 20, No. 2, pp. 1-1, 2023.

[6] R. A. Mishra, A. Kalla, A. Braeken, and M. Liyanage, "Blockchain regulated verifiable and automatic key refreshment mechanism for IoT", *IEEE Access*, Vol. 11, pp. 21758-21770, 2023.

[7] R. Jin, J. Hu, G. Min, and J. Mills, "Lightweight blockchain-empowered secure and efficient federated edge learning", *IEEE Transactions on Computers*, Vol. 20, No. 2, pp. 1-1, 2023.

[8] Z. Wu, J. Liu, J. Wu, Z. Zheng, and T. Chen, "TRacer: Scalable graph-based transaction tracing for account-based blockchain trading systems", *IEEE Transactions on Information Forensics and Security*, Vol. 20, No. 2, pp. 1-1, 2023.

[9] V. Agarwal and S. Pal, "HierChain: A Hierarchical Blockchain-Based Data Management System for Smart Healthcare", *IEEE Internet of Things Journal*, Vol. 10, No. 6, pp. 1-1, 2023.

[10] S. K. Set and G. S. Park, "Service-aware dynamic sharding approach for scalable blockchain", *IEEE Transactions on Services Computing*, Vol. 15, No. 1, pp. 1-1, 2022.

[11] G. Liu, Z. Wu, Y. Zhou, Y. Liu, and H. Kang, "Communitychain: Towards a scalable blockchain in smart home", *IEEE Transactions on Network and Service Management*, Vol. 20, No. 2, pp. 1-1, 2023.

[12] R. R. Irshad, S. Hussain, I. Hussain, J. A. Nasir, A. Zeb, K. M. Alalayah, et al., "IoT-Enabled Secure and Scalable Cloud Architecture for Multi-User Systems: A Hybrid Post-Quantum Cryptographic and Blockchain based Approach Towards a Trustworthy Cloud Computing", *IEEE Access*, Vol. 11, pp. 1-1, 2023.

[13] Chacko, N. M., Narendra, V. G., Balachandra, M., & Rathinam, S. (2023). Exploring IoT-Blockchain Integration in Agriculture: An Experimental Study, *IEEE Access*, Vol. 11, 130439-130450.

[14] N. M. Chacko, V. G. Narendra, M. Balachandra, and S. Rathinam, "Exploring IoT-Blockchain Integration in Agriculture: An Experimental Study", *IEEE Access*, Vol. 11, pp. 130439-130450, 2023.

[15] L. Li, D. Huang, and C. Zhang, "An efficient DAG blockchain architecture for IoT", *IEEE Internet of Things Journal*, Vol. 10, No. 2, 1286-1296.

[16] Z. Wan, W. Liu, and H. Cui, "HIBEChain: A hierarchical identity-based blockchain system for large-scale IoT", *IEEE Transactions on Dependable and Secure Computing*, Vol. 20, No. 2, pp. 1286-1301, 2022.

[17] C. D. Lee, J. H. Li, and T. H. Chen, "A Blockchain-Enabled Authentication and Conserved Data Aggregation Scheme for Secure Smart Grids", *IEEE Access*, Vol. 11, pp. 1-1, 2023.

[18] B. D. Deebak, F. H. Memon, S. A. Khowaja, K. Dev, W. Wang, N. M. F. Qureshi, et al., "A lightweight blockchain-based remote mutual authentication for AI-empowered IoT sustainable computing systems", *IEEE Internet of Things Journal*, Vol. 10, No. 8, pp. 6652-6660, 2022.

[19] W. Pourmajidi, L. Zhang, J. Steinbacher, T. Erwin, and A. Miranskyy, "Immutable log storage as a service on private and public blockchains", *IEEE Transactions on Services Computing*, Vol. 16, No. 1, pp. 356-369, 2021.

[20] P. Liu, W. Jing, X. Fu, Y. Xiao, and L. Jia, "Consortium blockchain-based security and efficient resource trading in V2V-assisted intelligent transport systems", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 20, No. 2, pp. 1-1, 2023.

[21] G. Solomon, P. Zhang, R. Brooks, and Y. Liu, "A Secure and Cost-Efficient Blockchain Facilitated IoT Software Update Framework", *IEEE Access*, Vol. 11, pp. 1-1, 2023.

[22] W. Yao, F. P. Deek, R. Murimi, and G. Wang, "Sok: A taxonomy for critical analysis of consensus mechanisms in consortium blockchain", *IEEE Access*, Vol. 11, pp. 1-1, 2023.

[23] Y. Wang, Y. Li, W. Jiao, G. Wang, J. Zhao, Y. Qiang, and K. Li, "An efficient, secured, and infinitely scalable consensus mechanism for peer-to-peer energy trading blockchain", *IEEE Transactions on Industry Applications*, Vol. 20, No. 2, pp. 1-1, 2023.

[24] Y. Wu, C. Zhang, and L. Zhu, "Privacy-preserving and traceable blockchain-based charging payment scheme for electric vehicles", *IEEE Internet of Things Journal*, Vol. 10, No. 6, pp. 1-1, 2023.

[25] J. Zhang, Y. Jiang, J. Cui, D. He, I. Bolodurina, and H. Zhong, "DBCPA: Dual blockchain-assisted conditional privacy-preserving authentication framework and protocol for vehicular ad hoc networks", *IEEE Transactions on Mobile Computing*, Vol. 21, No. 1, pp. 1-1, 2022.

[26] V. Mardiansyah, A. Muis, and R. F. Sari, "Multi-State Merkle Patricia Trie (MSMPT): High-Performance Data Structures for Multi-Query Processing Based on Lightweight Blockchain", *IEEE Access*, Vol. 11, pp. 1-1, 2023.

[27] E. Zhou, Z. Hong, Y. Xiao, D. Zhao, Q. Pei, S. Guo, and R. Akerkar, "MSTDB: a hybrid storage-empowered scalable semantic blockchain database", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 34, No. 1, pp. 1-1, 2022.

[28] X. Hao, W. Ren, Y. Fei, T. Zhu, and K. K. R. Choo, "A blockchain-based cross-domain and autonomous access control scheme for internet of things", *IEEE Transactions on Services Computing*, Vol. 16, No. 2, pp. 773-786, 2022.