Deep Reinforcement Learning-based Dynamic Sharding for Blockchain IoT

Pooja Khobragade and Ashok Kumar Turuk
Department of Computer science and engineering
National Institute of Technology, Rourkela
Rourkela, India
520cs1004@nitrkl.ac.in, akturuk@nitrkl.ac.in

***Abstract*—Internet of Things (IoT) and Blockchain integration are having a huge impact on the future of technological progress. IoT has progressed from an emerging notion to a widely used technology, shaping the future of digital connectivity. With billions of networked IoT devices generating large amounts of data, efficient data management becomes critical. Blockchain has been explored extensively to enhance security in IoT networks however, its scalability limitations become evident when handling large-scale deployments. Sharding is recognized as a promising approach to improve blockchain scalability by partitioning the network into multiple independent groups. These groups, called shards, process transactions in parallel, increasing throughput while reducing communication, computation, and storage overhead. Despite its advantages, many existing blockchain sharding models rely on static algorithm, which fail to accommodate the dynamic nature of blockchain networks. Factors such as variable node involvement and possible security concerns present problems that static sharding cannot solve. To address these restrictions, deep learning provides a strong solution for dynamic and multidimensional sharding in blockchain-based IoT systems. Deep learning, with its capacity to understand complex patterns and adapt to changing network circumstances, can improve the efficiency, security, and scalability of blockchain-powered IoT networks. This article proposes a deep reinforcement learning-based dynamic shards in blockchain IoT applications to overcome scalability difficulties.**

***Index Terms*—Blockchain, IoT, Scalability, Deep-learning, Dynamic Sharding**

I. Introduction

Integrating IoT and blockchain in smart buildings has significantly improved security and operational efficiency. However, many building IoT systems still depend on centralized cloud storage and control, making them vulnerable to single points of failure. Additionally, seamless data exchange between smart devices, such as monitoring gateways and access control systems, is crucial for efficient operation. As a decentralized ledger, blockchain improves data security, transparency, privacy, and efficiency while decreasing reliance on centralized authority [1]. It promotes confidence among distributed entities, allowing for safe and autonomous smart building ecosystems. Despite these advantages, blockchain suffers scalability issues that affect system throughput and transaction confirmation times, making it difficult to conduct large-scale IoT sensing activities efficiently. This work addresses these challenges by presenting a scalable, high-performance solution to optimize blockchain's role in managing extensive IoT sensing operations within smart buildings.

Sharding, a scalability technique derived from distributed database systems, involves partitioning data across multiple servers to enhance performance and reduce the workload on a single node [2]. This concept has been adapted in blockchain, leading to the development of various sharding solutions such as Elastico, OmniLedger, RapidChain, TEEShard, and Monoxide [3]. Ethereum 2.0 introduced a sharded blockchain integrated with a beacon chain to improve transaction throughput. All existing sharding systems typically assume that blockchain nodes are stable and identical within a peer-to-peer network. As a result, they tend to use random and static sharding methods, where nodes are randomly assigned to different shards with predetermined settings. However, these static sharding approaches are unsuitable for IoT blockchain systems, which involve dynamic and heterogeneous environments. Deep Reinforcement Learning (DRL) uses deep neural networks (DNNS) to solve dynamic control issues as function approximators, allowing efficient decision-making in situations with high-dimensional state and action spaces. This enables agents to adapt independently, analyze feedback, and improve their methods in response to changing circumstances. Blockchain integration with machine learning has recently gained attention in real-world applications like the Industrial Internet of Things (IIoT) [4]. DRL-based approaches are emerging as a promising solution to enhance blockchain scalability and performance. This research proposed a DRL-based dynamic sharding framework for blockchain IoT systems to solve the scalability and security challenges. The framework begins by securely registering and authenticating IoT devices on the blockchain, ensuring that only trusted devices are admitted into the network [5]. A reputation-based node selection mechanism is employed to evaluate and rank nodes based on historical behaviour, enabling reliable shard formation and leader selection. To adaptively manage the dynamic nature of IoT environments, the DRL model determines optimal shard partitioning, structure, and leader assignment by continuously observing the system state and receiving performance-based

feedback. The approach dynamically balances performance and security by frequently rearranging shard configurations in response to changing network conditions. An efficient consensus mechanism is executed within each shard by selecting the highest-reputation node as a leader, thereby ensuring lightweight and effective validation. Furthermore, a security layer continually checks consensus issues and detects rogue nodes using reputation ratings to ensure the system's integrity. The contributions of this paper are:

- A DRL-based adaptive sharding approach is proposed that dynamically manages partitions, structure, and leader selection to maintain a long-term balance between throughput and security across all shards.
- Reputation-based node selection to ensure trustworthy participation in shard formation and consensus.
- Adaptive shard reshuffling to optimize load distribution and resilience.

II. Related work

The rapid integration of IoT technologies across various industries has created a growing need for scalable, cross-domain platforms capable of aggregating heterogeneous data to support complex industrial ecosystems. Due to its inherent properties such as decentralization, transparency, and immutability blockchain has emerged as a promising solution to address the challenges of data security and trust in IoT environments. Consequently, academic and industrial researchers have proposed numerous blockchain-based frameworks to enhance the functionality and security of IoT systems.

These solutions have been applied across various domains, including Smart Homes, Smart Transportation, Smart Campuses, Smart Offices, Smart Factories, and Smart Healthcare, thereby illustrating the flexibility of blockchain in supporting diverse IoT applications. Current research efforts largely focus on the design of system architectures and consensus protocols tailored to IoT-specific requirements, with an emphasis on ensuring the integrity, confidentiality, and reliability of transmitted data.

For instance, in the context of IIoT, a consortium blockchain-based framework has been introduced to enable secure and transparent energy transactions. In healthcare, a blockchain-enabled data management system optimized through a lightweight consensus protocol that reduces communication and memory overhead is proposed by [6]. Similarly, other studies have explored the development of lightweight blockchain architectures to minimize computational and storage demands for resource-constrained IoT devices [7].

These systems' critical challenges are balancing performance and resource efficiency, particularly scalability and latency. Since different IoT use cases often entail varying quality-of-service (QoS) requirements, it is essential to adapt the underlying blockchain infrastructure during deployment to meet these specific operational demands effectively.

Sharding technology is commonly utilized in databases and cloud services. Sharding technology improves data storage efficiency by optimizing storage methods for each node as stored data grows. Sharding technology is now frequently employed in blockchain to address scalability issues. This strategy improves overall system efficiency and performance , as well as response time, by dividing data into smaller, more manageable chunks called shards. Recognizing the scalability benefits, academics have begun to investigate the inclusion of sharding into blockchain systems to alleviate fundamental performance limitations. In blockchain applications areas, transaction throughput has become an important statistic for assessing system efficacy. Sharding has emerged as a possible option in response, allowing blockchain networks to execute many more transactions per second, increasing scalability and responsiveness. Ensuring security in blockchain sharding is one of the key challenges, particularly in the presence of malicious actors. Shard-level attacks, such as takeovers of individual shards and double-spending across shards, can undermine the system's integrity [8]. Some prior proposed approaches, like random assignment of shards, periodic reconfiguration, and fraud detection mechanisms, help to mitigate these risks. Another significant challenge is efficiently managing transactions that span multiple shards. Handling transactions across multiple shards introduces coordination challenges [9]. Transaction consistency and complexity are overhead to sharding and must be maintained in protocols such as two-phase and asynchronous. Recent advances attempt to increase the efficiency of cross-shard transactions to enhance the security and scalability of blockchain. Despite developments, these systems still face considerable problems, especially in dealing with emerging threats like adaptive collusion assaults and dynamically adapting to IoT network needs. The proposed solution leverages the real-time shading using DRL to provide a strong method to prevent strategic collusion and ensure balanced node distribution.

III. Proposed DRL-based Dynamic Sharding model for Blockchain IoT

The proposed system model depicts a blockchain-based architecture shown in Figure 1 that uses dynamic sharding to improve IoT network speed, security, and scalability. The architecture has different layers of functionality of each layer is explained below:

- **IoT-device layer:** Real-world IoT devices that produce transaction data for the blockchain are shown in the bottom-most area. This covers mobile networks, industrial sensors, smart home appliances, driverless cars, and smart grid elements. Secure authentication methods like DTLS (Datagram Transport Layer Security) allow devices to communicate with the blockchain while maintaining data integrity and guarding against spoofing attacks.
- **Network layer:** It is intermediate layer that provides communication infrastructure to connect IoT devices

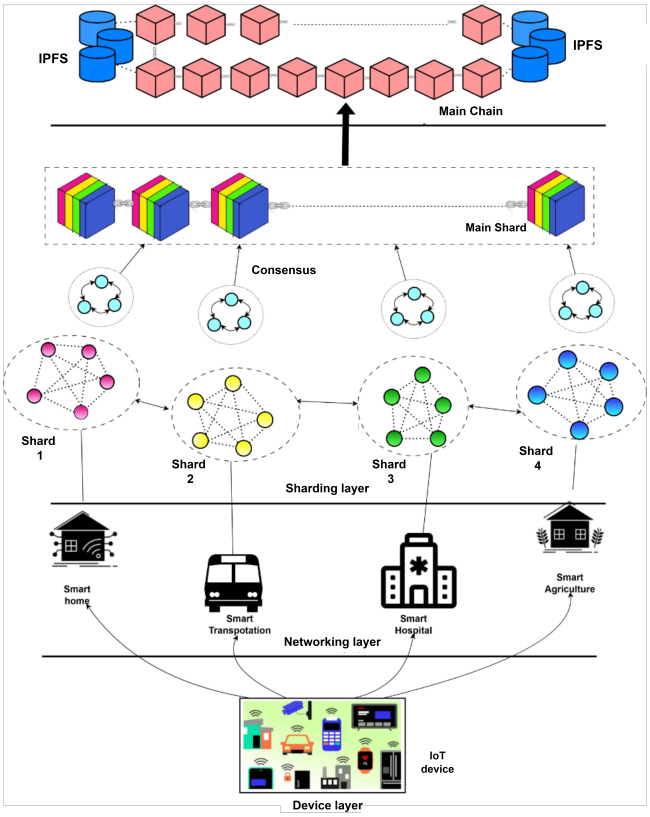

Fig. 1. Layer architecture for sharding solution

with the blockchain system. The network layer routes data from devices to the appropriate shards.

- **Sharding layer:**The blockchain network uses dynamic sharding, where nodes are split up into many shards according to demand and network conditions, to manage heavy transaction loads and improve efficiency. The Main Shard, a global ledger, records important state updates from various shards. Each shard processes a subset of transactions and functions semi-independently. Throughput is improved, and this division reduces computational overhead and improves throughput.
- **Blockchain layer:** IoT transactions are stored, verified, and secured using blockchain technology. Block represents individual shard transactions shared in a conventional blockchain structure. InterPlanetary File System (IPFS) stores off-chain data and blockchain metadata to lower the overhead of on-chain storage.

Algorithm 1 explains the working of the proposed architecture. The following are notations used in Algorithm: B: Blockchain Network, D: IoT Devices, S: Shard Set, A_{drl}: DRL Agent, $\phi(t)$: Current network traffic load, $\vartheta(t)$: Resource availability, $Rep(t)$: Reputation of participating nodes (to mitigate malicious attacks), R^*: Historical reward information (performance benchmark), $T_g(t)$: Target throughput, $B_s(t)$: Block size, $K(t)$: Number of shards, $Y(t)$: Node

Algorithm 1 DRL based dynamic sharding for blockchain IoT

Require: B, D, S, A_{drl}
Ensure: Optimized Sharding Configuration
begin
Shard Set Size Initialization
Determine $|S|$ using $|S| = \lceil N/\mu \rceil$
Step 1: IoT Device Registration, Authentication, and Initial Assignment
for each device $d_i \in D$ **do**
Verify device identity and register d_i in blockchain
if d_i fails authentication **then**
Reject registration request
else
Assign d_i to shard S_k using:

$$k = H(ID_{d_i}) \bmod |S|$$

end if
end for
Step 2: Reputation-Based Node Selection for Sharding
for each node $n_j \in B$ **do**
Compute reputation score:

$$Rep(n_j) = \alpha_1 C_s + \alpha_2 U_t + \alpha_3 A_c - \alpha_4 M_f$$

Sort nodes based on $Rep(n_j)$.
end for
Step 3: Dynamic Shard Configuration using DRL
for each epoch t **do**
Observe system state $S(t) = \{\phi(t), \vartheta(t), Rep(t), R^*\}$
Select action $A(t) = \{T_g(t), B_s(t), K(t), Y(t)\}$ using DRL policy
Execute shard reshuffling based on $A(t)$
Compute reward $R(t)$ based on throughput and security constraints
Update DRL model using $R(t)$
end for
Step 4: Consensus Execution and Leader Selection
for each shard S_k **do**
Select leader node with highest $Rep(n_j)$ in S_k
Execute consensus process within the shard
Update blockchain with validated transactions
end for
Step 5: Malicious Node Detection and Security Enhancement
Compute adaptive threshold:

$$Threshold = \mu_{Rep} - \sigma_{Rep}$$

for each node $n_j \in B$ **do**
Monitor consensus inconsistencies to estimate malicious ratio R^*
if $Rep(n_j) < Threshold$ **then**
Flag n_j as malicious and isolate from consensus
end if
end for
end

assignment strategy, C_s: consensus success rate, U_t: uptime ratio, A_c: accuracy of transaction validation, M_f: Misbehavior frequency, $H(\cdot)$: Cryptographic hash, ID_{d_i}: Device's blockchain address, μ_{Rep} and σ_{Rep} are mean and standard deviation of recent reputation score, $S_1, S_2, \ldots, S_{|S|}$: Shard identifiers.

The core contribution of this work is the integration of a DRL agent into the sharding process for blockchain-enabled IoT networks. Traditional static sharding techniques predefine shard boundaries and node assignments, which limits their ability to respond to dynamic conditions such as varying transaction rates, node failures, or security threats. In contrast, the proposed approach allows the network to self-adapt through continuous learning. Also balances the performance and security without manual tuning and scales seamlessly with network size and transaction volume.

IV. Methodology and Experimental setup

The proposed algorithm introduces a DRL framework to dynamically configure sharding in a blockchain-enabled IoT environment. Initially, the network determines the shard count based on node population and communication constraints, followed by a deterministic hash-based mapping of devices to shards to ensure balanced distribution and resistance to targeted attacks. Node reputation is calculated from operational metrics, including consensus success rate, uptime, validation accuracy, and misbehavior frequency, allowing for trust-based leader selection within each shard. The DRL agent continuously monitors system states such as load, latency, security risk, and reputation distribution and determines optimal actions for shard reshuffling, leader reallocation, or resource adjustment to maximize throughput while maintaining security. Unlike static sharding models, this adaptive mechanism learns from feedback over time, enabling the system to respond to changing node availability, traffic fluctuations, and malicious activity. This integration of DRL into the sharding process represents the core novelty of the work, as it couples scalability enhancement with real-time security adaptation in blockchain-based IoT applications.

The proposed DRL-based dynamic sharding mechanism is evaluated in a simulated blockchain-IoT environment built on Ethereum with Python-based IoT device emulation. Application layer sharding controlled by a Proximal Policy Optimization (PPO)-based DRL agent. Experiments is conducted with 100 – 500 IoT nodes, 20–100 validator nodes, 256 B transactions at 1 – 10 tx/s, and a shard size target of 20 nodes. The DRL agents' learning rate is 0.0003, $\gamma = 0.99$, $\beta_1 = 0.5$, $\beta_2 = 0.3$, $\beta_3 = 0.2$, and 500 training episodes. DRL agent optimized shard configuration based on throughput, latency, and security. The reputation scores are derived from consensus participation metrics.

A. Results and Discussion

The proposed DRL-based dynamic sharding framework outperformed both static and adaptive techniques in throughput, latency, and security metrics.

Figure 2 illustrates the relationship between the number of shards, nodes, and the shard complexity factor in dynamic sharding. As the number of nodes and shards increases, the shard complexity factor also increases, indicating a higher computational overhead. The graph shown in Figure 3 compares the transaction throughput of three sharding techniques: (i) Static, (ii) Adaptive, and (iii) DRL-based dynamic sharding. The DRL-based method consistently achieves the highest with 42% transactions per second (TPS) improvement compared to static sharding as the number of nodes increases. The graph shown in Figure 4 illustrates the latency comparison among the three sharding methods. The DRL-based approach consistently achieves the lowest latency, with a 31% reduction across different network sizes, by learning and adapting to dynamic network conditions in real time. At the same time, static sharding relies on pre-defined rule-based adjustments, resulting has the highest latency, with fixed shard structures and load imbalance. The graph shown in Figure 5 represents shard integrity as the percentage of malicious nodes increases. The DRL-based dynamic sharding maintains higher shard integrity than Adaptive and Static Sharding, making it more resilient to security threats.

The DRL has the ability to continuously learn from network state changes such as node participation, malicious activity, and transaction load and adapt shard configurations accordingly. Unlike static methods that remain fixed and adaptive methods that rely on pre-defined rules, the DRL agent optimizes a multi-objective reward function balancing scalability, efficiency, and security. This enables proactive shard reallocation, balanced leader selection, and load-aware shard count adjustment, resulting in better performance under both normal and adversarial conditions.

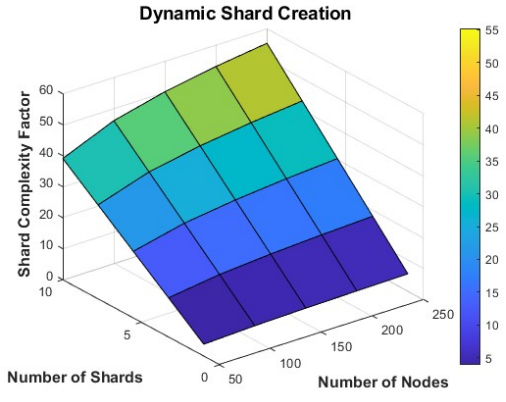

Fig. 2. Shard creation using DRL approach

V. Conclusion

This paper proposes a DRL-based dynamic shard creation mechanism to solve the scalability issue in blockchain IoT architecture. Current sharding blockchain solutions cannot

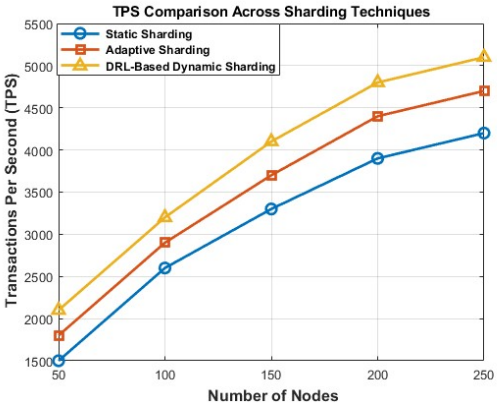

Fig. 3. Comparison of TPS vs Number of nodes for different sharding approaches

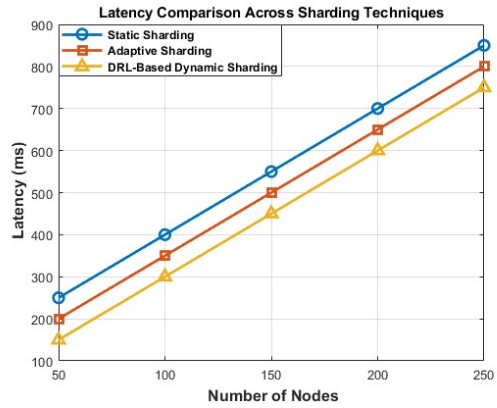

Fig. 4. Comparison of latency vs Number of nodes for different sharding approaches

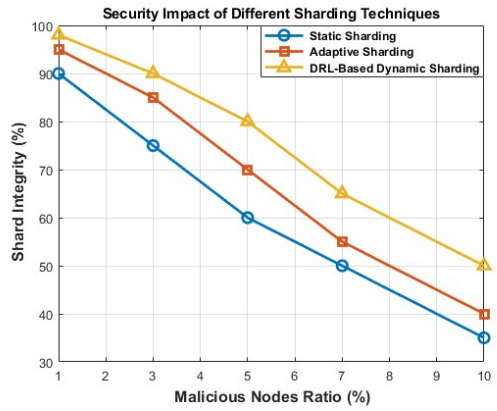

Fig. 5. Comparison of shard integrity vs Malicious nodes for different sharding approaches

satisfy the significant real-time data-sharing requirements of IoT applications because of their limited throughput and security vulnerabilities. We integrate sharding with the DRL mechanism to improve the sharding process and handle the constantly shifting network circumstances and demands. This technique maintains strong security while significantly increasing the sharding blockchain's scalability in IoT applications.

In future research, we aim to explore the creation of a lightweight consensus mechanism for sharding and investigate an identity management strategy for IoT devices. These efforts will further improve the security and performance of sharded blockchain systems in resource-constrained IoT devices.

REFERENCES

[1] 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*, 2023.

[2] Z. Hong, S. Guo, E. Zhou, W. Chen, H. Huang, and A. Zomaya, "Gridb: Scaling blockchain database via sharding and off-chain cross-shard mechanism," *arXiv preprint arXiv:2407.03750*, 2024.

[3] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," in *Proceedings of the 2019 international conference on management of data*, 2019, pp. 123–140.

[4] Z. Zhen, X. Wang, H. Lin, S. Garg, P. Kumar, and M. S. Hossain, "A dynamic state sharding blockchain architecture for scalable and secure crowdsourcing systems," *Journal of Network and Computer Applications*, vol. 222, p. 103785, 2024.

[5] C. Wei, H. Lin, Y. Que, and X. Wang, "Fcdsb: A fog computing network architecture based on dynamic sharding blockchain for consumer electronics in aiot," *IEEE Transactions on Consumer Electronics*, 2024.

[6] B. Zaabar, O. Cheikhrouhou, F. Jamil, M. Ammi, and M. Abid, "Healthblock: A secure blockchain-based healthcare data management system," *Computer Networks*, vol. 200, p. 108500, 2021.

[7] E. Bandara, D. Tosh, P. Foytik, S. Shetty, N. Ranasinghe, and K. De Zoysa, "Tikiri—towards a lightweight blockchain for iot," *Future Generation Computer Systems*, vol. 119, pp. 154–165, 2021.

[8] T. Cai, W. Chen, J. Zhang, and Z. Zheng, "Smartchain: A dynamic and self-adaptive sharding framework for iot blockchain," *IEEE Transactions on Services Computing*, 2024.

[9] C. Qin, B. Guo, Y. Shen, T. Li, Y. Zhang, and Z. Zhang, "A secure and effective construction scheme for blockchain networks," *Security and Communication Networks*, vol. 2020, no. 1, p. 8881881, 2020.