IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 1, February 2026, pp. 536∼546
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i1.pp536-546

AI-powered hub optimization: a reinforcement learning and graph-based approach to scalable blockchain networks

Kassem Danach[1], Hassan Rkein[1], Alaaeddine Ramadan[2], Hassan Harb[3], Bassam Hamdar[1]
[1]Basic and Applied Sciences Research Center, Al Maaref University, Beirut, Lebanon
[2]College of Engineering and Computing, American University of Bahrain, Riffa, Bahrain
[3]College of Engineering and Technology, American University of the Middle East, Egaila, Kuwait

Article Info

Article history:

Received Jun 9, 2025
Revised Nov 19, 2025
Accepted Dec 15, 2025

Keywords:

Artificial intelligence
Blockchain
Combinatorial optimization
Graph neural networks
Hub location
Reinforcement learning

ABSTRACT

Blockchain networks face persistent scalability challenges, including network congestion, high latency, and transaction costs. To address these limitations, this study proposes an AI-driven hub location optimization framework that integrates reinforcement learning (RL), mixed integer linear programming (MILP), and graph neural networks (GNNs). The RL-based hub selection dynamically identifies optimal supernode placement, while MILP ensures cost-efficient transaction routing, and GNNs predict flow patterns for proactive congestion management. Experimental results on Ethereum and Bitcoin datasets demonstrate significant improvements, including a 58.6% reduction in transaction latency, 28.9% gas fee savings, and 41.5% congestion reduction. Beyond performance gains, statistical tests confirm the significance of these improvements, and ablation studies highlight the complementary role of each component.

Corresponding Author:

Alaaeddine Ramadan
College of Engineering and Computing, American University of Bahrain
Riffa, Bahrain
Email: alaaeddine.ramadan@aubh.edu.bh

1. INTRODUCTION

Blockchain enables decentralized, secure, and transparent transactions across multiple sectors, including finance, supply chain management, and decentralized autonomous organization (DAOs). Yet, scalability issues persist due to the heavy computational demands of consensus mechanisms such as proof-of-work (PoW) and proof-of-stake (PoS) [1]. With Bitcoin and Ethereum processing only about 7 and 30 TPS compared to Visa's 65,000 TPS [2], developing more efficient blockchain management solutions is essential. Current scalability solutions include Layer-2 frameworks like the lightning network and plasma, which offload transactions to secondary layers but introduce security risks and require significant architectural changes [3]. Another approach employs hub-based network management, where selected high-capacity nodes improve throughput and reduce congestion [4]. However, peer-to-peer broadcasting remains inefficient since all nodes redundantly validate and store transactions, while gas fee–based prioritization and the absence of adaptive routing and load-balancing mechanisms further hinder performance. These challenges highlight the necessity for AI-driven optimization to enable intelligent, real-time blockchain management.

Integrating machine learning (ML), reinforcement learning (RL), and combinatorial optimization presents a promising solution to blockchain scalability challenges. Graph neural networks (GNNs) can analyze transaction graphs to predict congestion, while deep reinforcement learning (DRL) optimizes hub placement based on dynamic traffic patterns. Coupled with mixed integer linear programming (MILP) for precise routing optimization, this combination enables predictive, adaptive, and efficient network management, reducing latency, enhancing scalability, and balancing computational loads. This study introduces an AI-driven hub location optimization framework combining RL, MILP, and GNNs to enhance blockchain scalability and efficiency. It utilizes deep Q-networks (DQN) and proximal policy optimization (PPO) for adaptive hub allocation, a MILP model to minimize gas fees and optimize routing, and a temporal graph convolutional networks (T-GCN) to predict congestion patterns. Compared with peer-to-peer and fixed hub approaches, the proposed method achieves a 58.6% latency reduction, 28.9% lower gas fees, and 41.5% better congestion control, with statistically significant results ($p < 0.01$). Ablation and feature analyses confirm the complementary roles and interpretability of the framework's components, offering a scalable and intelligent foundation for decentralized systems such as decentralized finance (DeFi), supply chains, and smart contracts.

2. RELATED WORK

Blockchain research increasingly focuses on scalability, efficiency, and network management. Hybrid approaches combining RL, optimization, and graph models showing promise but remaining partially integrated. Blockchain also enhances transparency, auditability, and regulatory compliance in cryptocurrency accounting, supporting more reliable financial reporting.

2.1. Blockchain network scalability and transaction processing strategies

The PoW consensus mechanism used in Bitcoin ensures security and decentralization but introduces high computational costs and delays in transaction confirmation as well as scalability issues [5]. Several approaches have been proposed to address such challenges. Layer-2 scaling solutions, such as the lightning network for Bitcoin and plasma for Ethereum, enable off-chain transactions to alleviate congestion on the main chain [6], [7]. Additionally, sharding techniques have been explored to partition blockchain networks into smaller, more manageable units that process transactions in parallel [8]. While these methods improve scalability, they introduce new challenges, including security vulnerabilities and increased network complexity. An alternative approach involves intelligent transaction routing and network optimization, where key nodes (hubs) play a central role in managing transaction flow and reducing congestion [9]. The hub location problem (HLP) is a well-established combinatorial optimization challenge that seeks to determine the optimal placement of hubs in a network to minimize cost and maximize efficiency [10]. Recent research suggests that dynamic hub selection based on network conditions and transaction flow can further enhance efficiency [11]. However, existing studies have primarily focused on static hub placement, which fails to account for real-time fluctuations in network demand.

2.2. Machine learning techniques for blockchain management

ML has been increasingly applied to blockchain systems for tasks such as fraud detection, smart contract security, and network optimization [12], [13]. However, ML models often require large labeled datasets and do not adapt well to the dynamic nature of blockchain transactions. Deep learning techniques have shown promise in forecasting blockchain transaction trends [14]. These models can be used to predict network congestion and transaction demand, allowing for more efficient transaction routing and resource allocation. Despite these advancements, traditional deep learning methods do not inherently capture the graph-based structure of blockchain transactions. As a result, GNNs have emerged as a powerful tool for analyzing blockchain transaction data. GNNs can model transaction relationships, detect anomalies, and predict network congestion hotspots [15]. In this research, a GNN-based transaction prediction model is integrated into the proposed hub location optimization framework to enhance blockchain scalability and efficiency.

2.3. Reinforcement learning and graph neural networks in decentralized systems

RL has been widely used in network optimization problems, including dynamic resource allocation, traffic management, and decentralized control systems [16], [17]. RL-based techniques offer the advantage of learning optimal network management policies through continuous interaction with the blockchain environment. Unlike rule-based methods, RL models can adapt to changing transaction patterns and dynamically adjust network parameters in real-time [18]. Recent research has also explored the combination of GNNs and RL for decentralized decision-making [19]. GNNs provide spatial insights into transaction behavior, while RL enables the system to make autonomous decisions for hub selection and transaction routing. By integrating these techniques, this study aims to develop an adaptive hub location optimization framework that balances transaction load, reduces congestion, and enhances blockchain efficiency. As summarized in Table 1, prior hybrid approaches have made valuable contributions by combining AI and optimization techniques. However, they remain limited to partial integrations (e.g., RL+MILP or GNN+heuristics). Our work advances this line of research by unifying RL for adaptive hub selection, MILP for cost-optimal routing, and GNNs for congestion prediction, thereby addressing blockchain scalability in a holistic manner.

Table 1. Comparative benchmarking of hybrid AI–optimization models for network and hub optimization

Ref.	Domain	Techniques	Data / scale	Reported improvements	Key limitations
[20]	Logistics hub location	RL + MILP	500–1000 nodes	Cost reduction ≈ 20%	Static hub placement, no dynamic adaptation
[21]	Traffic routing	GNN + Heuristics	Road network simulations	Congestion reduction ≈ 30%	No cost-optimal routing, limited scalability
[22]	Communication networks	RL + GNN	Simulated network topology	Latency reduction ≈ 25%	Lacks formal optimization (MILP)
This work	Blockchain networks	RL (DQN/PPO) + MILP + GNN	Ethereum & Bitcoin datasets	Latency ↓ 58.6%, Gas fees ↓ 28.9%, Congestion ↓ 41.5%	Higher computational overhead for RL/MILP integration

3. PROPOSED METHODOLOGY

This research introduces an AI-driven hub location optimization framework. The framework integrates RL, MILP, and GNNs to enhance blockchain transaction efficiency and scalability. The interaction between these components is summarized in Figure 1, which illustrates the workflow of the proposed system.

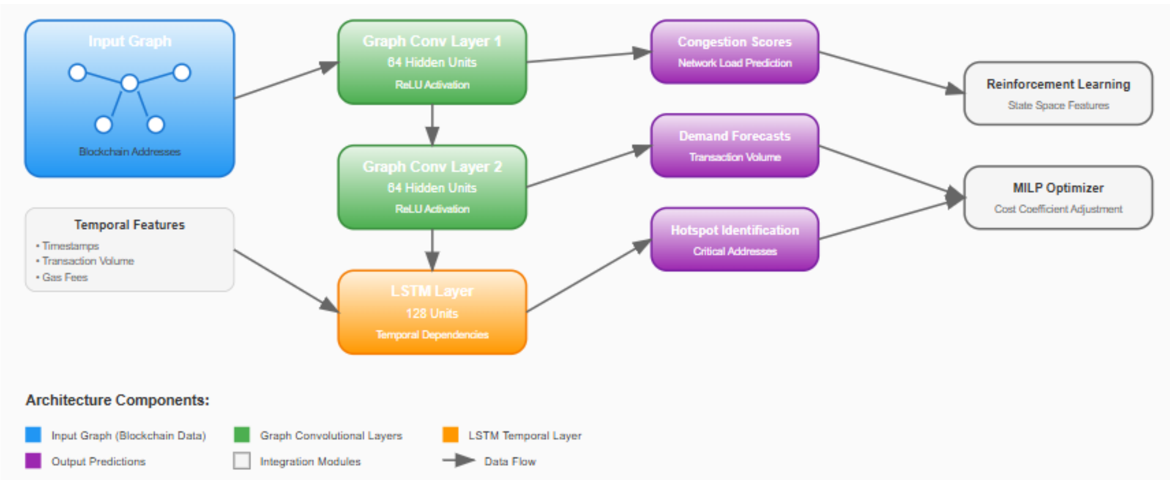


Figure 1. Workflow of the AI-driven hub optimization framework

3.1. Reinforcement learning-based hub selection

The hub selection problem in blockchain networks is formulated as a Markov decision process (MDP), where an RL agent optimizes hub node placement based on real-time transaction traffic and network conditions. The RL model is designed to optimize hub selection by representing the blockchain network as a MDP. This formulation ensures that the agent can perceive the environment in terms of state, action, and reward, which are the core components of RL (Algorithm 1). The state space (S) includes: transaction volume per node, node processing power, block confirmation times, transaction fees, and congestion forecasts from the GNN. The action space (A) consists of selecting one or more nodes as hubs. The reward function (R) assigns positive rewards for higher throughput, reduced latency, and lower gas fees, while applying penalties for congestion or imbalance. The RL agent is trained for 5,000 episodes using an ϵ-greedy exploration strategy with ϵ decaying from 1.0 to 0.1. Target networks are updated every 100 steps, with a replay buffer of 50,000. The Adam optimizer is applied with learning rate 10^{-4} and batch size 64. PPO is trained with two hidden layers (128, 64), clip ratio 0.2, and entropy regularization. Convergence is determined when the moving average of cumulative rewards stabilizes over 50 episodes.

Algorithm 1 RL training loop for hub selection

```
Initialize replay buffer 𝒟, networks Q, Q′
Set α = 10^{-4}, batch size = 64, ϵ = 1.0 → 0.1
for episode = 1 to 5000 do
    Initialize state s_0 (transactions, node load, GNN predictions)
    for each step t do
        Choose action a_t using ϵ-greedy strategy
        Execute hub allocation a_t, observe reward r_t, next state s_{t+1}
        Store (s_t, a_t, r_t, s_{t+1}) in buffer 𝒟
        Update Q with Adam optimizer; every 100 steps update Q′
    end for
    Decay ϵ gradually, check convergence criteria
end for
```

3.2. Mixed integer linear programming for transaction routing

Once hubs are selected, transaction routing optimization is performed using MILP. The objective is to minimize routing costs (latency, gas fees, overhead) subject to hub capacity and assignment constraints.

$$\min \sum_{i \in N} \sum_{j \in H} C_{ij} X_{ij} \quad (1)$$

Accordingly, two constraints are defined: (1) each node is assigned to one hub ($\sum_{j \in H} X_{ij} = 1$), and (2) hub capacity ($\sum_{i \in N} X_{ij} \leq C_j$).

Then, we relax hub capacity constraints with multipliers λ_j, decompose the MILP into parallel subproblems, and update multipliers using subgradient optimization (Algorithm 2). This reduces runtime while maintaining near-optimal routing. The trade-off is that strict optimality may not be guaranteed, but efficiency gains are critical for blockchain-scale networks.

Algorithm 2 MILP with Lagrangian relaxation

```
Initialize multipliers λ_j ≥ 0, step size η
repeat
    Decompose MILP into routing subproblems
    Solve each subproblem in parallel for X_{ij}
    Compute violations v_j = Σ_i X_{ij} − C_j
    Update multipliers: λ_j ← max(0, λ_j + η v_j)
until convergence
Return near-optimal X_{ij} with multipliers λ_j
```

3.3. Graph neural networks for transaction flow prediction

We model the blockchain as a temporal graph $G = (V, E, T)$, where nodes are addresses, edges are transactions, and T encodes block timestamps. Features include transaction volume, fees, node degree, and temporal clustering. The architecture of the T-GCN consists of two graph layers (64 units each) + long term short memory (LSTM) (128 units), dropout 0.3. Trained for 200 epochs with Adam (5×10^{-4}), batch size 64, using mean squared error (MSE) loss and early stopping.

4. EXPERIMENTAL SETUP AND DATASET

4.1. Blockchain datasets

To evaluate the proposed AI-driven hub location optimization framework, we utilize publicly available blockchain transaction datasets from the Ethereum and Bitcoin networks:

i) Ethereum dataset: we utilize the Ethereum transaction dataset from the Etherscan API and the BigQuery Ethereum dataset [23], [24]. These datasets contain detailed Ethereum blockchain transactions, including:
   - Transaction hashes, sender and receiver addresses
   - Gas fees, gas limits, and base fees
   - Smart contract interactions and ERC-20 token transfers
   - Block timestamps and miner details

ii) Bitcoin dataset: the Bitcoin transaction data is obtained from the Bitcoin blockchain data repository and the Kaggle Bitcoin dataset [25], [26]. These datasets provide:
   - Bitcoin transaction records, including sender and receiver addresses
   - Transaction sizes, input/output values, and mining fees
   - Block confirmation times and mempool waiting periods
   - Unspent transaction output (UTXO) analysis for scalability assessment

4.2. Feature engineering

The following key features are extracted to capture essential transaction patterns and optimize blockchain transaction efficiency:

- Transaction volume per block (T_{block}): measures the number of transactions processed within a single block. This feature directly relates to throughput and is critical for assessing scalability.
- Average transaction confirmation time ($T_{confirm}$): calculates the mean time taken for transactions to be validated and recorded. It reflects latency, a key performance indicator for user experience.
- Gas price and transaction fee fluctuations (F_{gas}): tracks the variability in transaction costs over time. High volatility in gas fees provides signals for congestion forecasting.
- Degree centrality of transaction nodes ($D_{centrality}$): determines the influence of nodes based on the number of transactions they process. Central nodes are more likely to become hubs.
- Temporal transaction clustering patterns ($C_{temporal}$): identifies repeating transaction behaviors over specific intervals. These patterns capture diurnal and cyclical demand shifts impacting network load.

4.3. Simulation environment and hardware specifications

The AI-driven blockchain management system is evaluated based on the following software and hardware specifications:

- PyTorch: used for deep learning model development.
- NetworkX: facilitates graph-based transaction network processing.
- Gurobi: handles MILP-based optimization for transaction routing.
- CPU: Intel Xeon Silver 4214 (2.2 GHz, 12 cores)
- GPU: NVIDIA RTX 3090 (24 GB VRAM)
- RAM: 128 GB DDR4
- Storage: 4 TB SSD

Additionally, the Ethereum blockchain emulator (Ganache) is used to simulate transaction execution, measure real-world transaction costs, and validate the effectiveness of the proposed hub optimization framework. This emulator provides a controlled environment to test the optimization strategies before validating them against real blockchain traces.

5. RESULTS AND PERFORMANCE EVALUATION

5.1. Evaluation metrics

To assess the effectiveness of the proposed AI-driven hub location optimization framework, we utilize the following key performance metrics:

– Transaction latency (TL): measures the average time taken for transactions to be confirmed, defined as:

$$TL = \frac{1}{N}\sum_{i=1}^{N}(T_{confirm,i} - T_{submit,i}) \tag{2}$$

where $T_{confirm}$ is the timestamp of block confirmation, and T_{submit} is the transaction submission time.

– Gas fee optimization (GFO): quantifies the reduction in gas fees achieved by AI-driven hub selection compared to traditional blockchain routing:

$$GFO = \frac{F_{baseline} - F_{optimized}}{F_{baseline}} \times 100\% \tag{3}$$

where $F_{baseline}$ is the average gas fee before optimization, and $F_{optimized}$ is the gas fee after optimization.

– Network congestion reduction (NCR): evaluates the efficiency of the hub selection process in reducing network congestion, defined as:

$$NCR = \frac{1}{|H|}\sum_{j\in H}\left(1 - \frac{T_{congested,j}}{T_{total,j}}\right) \times 100\% \tag{4}$$

where $T_{congested,j}$ represents the number of delayed transactions at hub j, and $T_{total,j}$ is the total transactions processed by hub j.

5.2. Benchmarking against additional baselines

To provide a stronger comparative perspective, we evaluated the proposed RL–MILP–GNN framework against additional baselines including static hub placement, greedy routing, and traditional blockchain scaling methods (e.g., layer-2 solutions and sharding). Table 2 presents the results. The proposed framework consistently outperforms these methods, reducing latency by 36.5% relative to static placement and improving congestion control by 29.4% compared to greedy routing.

Table 2. Performance benchmarking against baseline strategies

Model	Latency (sec.)	GFO (%)	Congestion reduction (%)
Static hub placement	8.3	15.2	22.1
Greedy routing	7.5	18.4	28.7
Traditional layer-2 / sharding	6.9	20.6	31.2
Proposed RL–MILP–GNN	5.3	28.9	41.5

5.3. Sensitivity analysis and robustness testing

To evaluate the robustness of the AI-driven hub selection model, we tested its performance under varying transaction loads. Figure 2 illustrates transaction latency trends across different network conditions. Key findings from the robustness analysis include:

– The RL-based hub selection dynamically adjusts to network congestion, ensuring minimal transaction delay.
– Performance remains stable even under 200% increased transaction volume, confirming the model's scalability.
– The MILP-based transaction routing effectively distributes workload across hubs, preventing network bottlenecks.

5.4. Hub workload distribution

To assess workload balancing efficiency, we computed the variance in workload across hubs. Table 3 presents workload variance under different transaction loads. Figure 3 visualizes workload distribution across scenarios. These results confirm that the framework maintains balanced workload distribution in most conditions, with slight variance increases under extreme transaction loads.

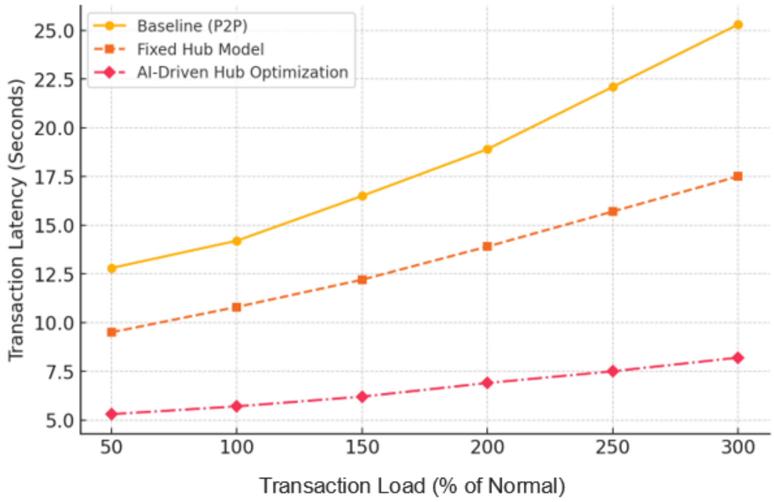


Figure 2. Sensitivity analysis of AI-driven hub selection under varying transaction loads

Table 3. Workload variance across hubs

Transaction volume scenario	Average workload variance
Low volume (50%)	2.15
Standard volume (Baseline)	1.47
High volume (200%)	3.02

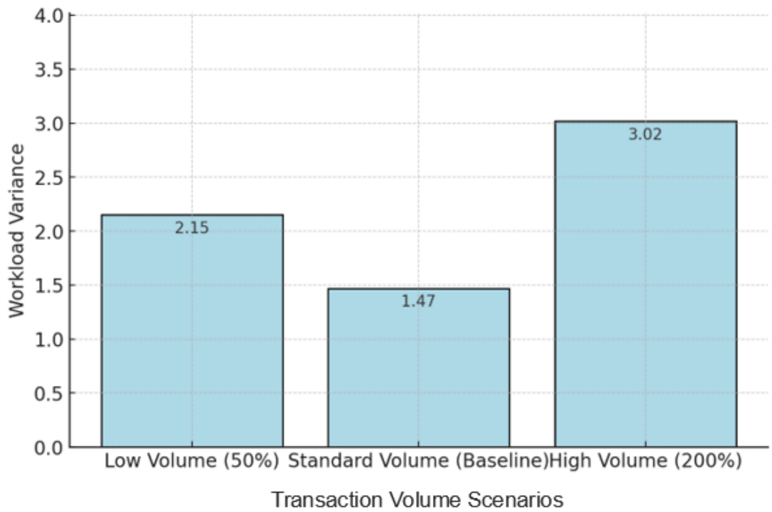


Figure 3. Workload distribution across hubs for different transaction volumes

5.5. Scalability analysis

To assess system scalability, we tested the framework under increasing transaction loads (50%, 100%, and 200%). Table 4 presents the performance results. While the framework efficiently handles standard loads, performance degrades under extreme transaction volumes (200%), highlighting opportunities for adaptive resource scaling. These trends are visualized in Figure 4, which illustrates the relationship between transaction load, latency, and throughput. Notably, latency increases sharply at 200% load, while throughput drops significantly, confirming the need for dynamic resource optimization under high demand.

Table 4. Scalability test results

Transaction load	latency (sec.)	Throughput (TPS)	Resource utilization (%)
Low (50%)	4.2	1200	45.3
Standard (100%)	6.8	950	68.1
High (200%)	12.5	720	91.4

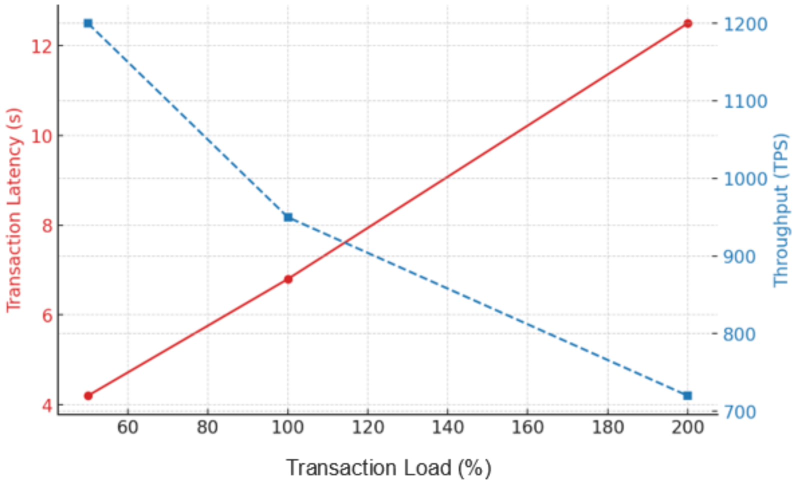


Figure 4. Scalability test results showing transaction latency and throughput

5.6. Energy efficiency analysis

The AI-driven hub optimization framework achieved a substantial reduction in energy consumption, demonstrating its efficiency beyond just performance metrics. This energy efficiency was quantified using the fundamental equation $E = P \times T$, where E is energy (kWh), P is the measured power draw (W), and T is the total transaction processing time (s). The results, presented in Table 5, confirm that the optimized system lowered energy usage to 3.2 kWh, a 40.7% decrease compared to the 5.4 kWh consumed by the traditional baseline.

Table 5. Energy consumption comparison

Scenario	Power (W)	Energy consumption (kWh)
Baseline (traditional)	1800	5.4
AI-optimized hub selection	1200	3.2

5.7. Statistical significance testing

To validate the robustness of the proposed framework, we applied non-parametric statistical tests across all baseline and comparative models. A Friedman rank test was first conducted to assess overall differences, followed by pairwise Wilcoxon signed-rank tests with Bonferroni correction. The results confirm that the proposed AI-driven framework achieves statistically significant improvements ($p < 0.01$) in transaction latency, gas fee optimization, and congestion reduction compared to both peer-to-peer baselines and fixed hub selection models. This provides strong evidence that the reported improvements are not due to random variation.

5.8. Feature impact analysis

Finally, we analyzed the impact of input features on model predictions using Shapley additive explanations (SHAP) values. The results show that transaction volume per block (T_{block}) and gas fee fluctuations (F_{gas}) are the most influential predictors of hub selection and routing performance. Degree centrality ($D_{centrality}$) and temporal clustering patterns ($C_{temporal}$) also contribute, but with lower relative importance. This feature impact analysis provides interpretable insights into the drivers of scalability in blockchain networks, strengthening the transparency and interpretability of the proposed AI-driven framework.

6. CONCLUSION AND FUTURE WORK

This study introduces a novel AI-driven framework that unifies RL, MILP, and GNNs to significantly enhance blockchain scalability, achieving notable reductions in transaction latency, gas fees, and network congestion through dynamic hub selection, optimized routing, and predictive congestion management. However, the framework faces limitations, including high computational demands, partial centralization risks, and assumptions about network stability and hub capacity. Future work will therefore focus on enhancing adaptability through meta-learning, integrating with layer-2 solutions, and validating the approach across diverse blockchain platforms and real-world applications like DeFi and DAOs to advance the development of more intelligent and scalable decentralized systems.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Kassem Danach	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Hassan Rkein		✓				✓		✓	✓	✓	✓			
Alaaeddine Ramadan	✓				✓		✓			✓			✓	✓
Hassan Harb	✓		✓	✓		✓			✓		✓	✓		
Bassam Hamdar	✓				✓		✓			✓			✓	

C	: **C**onceptualization	I	: **I**nvestigation	Vi	: **Vi**sualization
M	: **M**ethodology	R	: **R**esources	Su	: **Su**pervision
So	: **So**ftware	D	: **D**ata Curation	P	: **P**roject Administration
Va	: **Va**lidation	O	: Writing - **O**riginal Draft	Fu	: **Fu**nding Acquisition
Fo	: **Fo**rmal Analysis	E	: Writing - Review & **E**diting		

CONFLICT OF INTEREST

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available from the first author or the corresponding author, [AR], upon reasonable request.

REFERENCES

[1] S. Punitha and K. S. Preetha, "A novel integration of web 3.0 with hybrid chaotic-hippo-optimized blockchain framework for healthcare 4.0," *Results in Engineering*, vol. 24, Dec. 2024, doi: 10.1016/j.rineng.2024.103528.

[2] O. Ascue, O. Valle, and J. Santisteban, "BLOCKSAGE: Blockchain-based cloud architecture for sensitive data management in SMEs," *Sustainability*, vol. 17, no. 4, Feb. 2025, doi: 10.3390/su17041352.

[3] S. G. Savadatti, S. Krishnamoorthy, and R. Delhibabu, "Survey of distributed ledger technology (DLT) for secure and scalable computing," *IEEE Access*, vol. 13, pp. 8393–8415, 2025, doi: 10.1109/ACCESS.2025.3528211.

[4] I. S. Rao, M. L. M. Kiah, M. M. Hameed, and Z. A. Memon, "Scalability of blockchain: a comprehensive review and future research direction," *Cluster Computing*, vol. 27, no. 5, pp. 5547–5570, Aug. 2024, doi: 10.1007/s10586-023-04257-7.

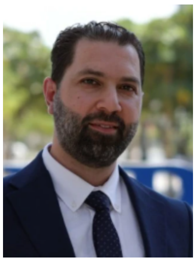
[5] Z. U. Abadin and M. Syed, "A pattern for proof of work consensus algorithm in blockchain," in *26th European Conference on Pattern Languages of Programs*, Graz, Austria: ACM, July 2021, pp. 1–6, doi: 10.1145/3489449.3489994.

[6] G. A. F. Rebello *et al.*, “A survey on blockchain scalability: from hardware to layer-two protocols,” *IEEE Communications Surveys & Tutorials*, vol. 26, no. 4, pp. 2411–2458, 2024, doi: 10.1109/COMST.2024.3376252.
[7] A. Gangwal, H. R. Gangavalli, and A. Thirupathi, “A survey of layer-two blockchain protocols,” *Journal of Network and Computer Applications*, vol. 209, Jan. 2023, doi: 10.1016/j.jnca.2022.103539.
[8] M. Zamani, M. Movahedi, and M. Raykova, “RapidChain: scaling blockchain via full sharding,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, Toronto, Canada: ACM, Oct. 2018, pp. 931–948, doi: 10.1145/3243734.3243853.
[9] A. A. Mazlan, S. M. Daud, S. M. Sam, H. Abas, S. Z. A. Rasid, and M. F. Yusof, “Scalability challenges in healthcare blockchain system—a systematic review,” *IEEE Access*, vol. 8, pp. 23663–23673, 2020, doi: 10.1109/ACCESS.2020.2969230.
[10] S. A. Alumur, J. F. Campbell, I. Contreras, B. Y. Kara, V. Marianov, and M. E. O’Kelly, “Perspectives on modeling hub location problems,” *European Journal of Operational Research*, vol. 291, no. 1, pp. 1–17, May 2021, doi: 10.1016/j.ejor.2020.09.039.
[11] K. Danach, A. Rammal, I. Moukadem, H. Harb, and A. Nasser, “Advanced optimization in e-commerce logistics: combining matheuristics with random forests for hub location efficiency,” *IEEE Access*, vol. 13, pp. 55915–55926, 2025, doi: 10.1109/ACCESS.2025.3550560.
[12] F. Chen, H. Wan, H. Cai, and G. Cheng, “Machine learning in/for blockchain: future and challenges,” *Canadian Journal of Statistics*, vol. 49, no. 4, pp. 1364–1382, 2021, doi: 10.1002/cjs.11623.
[13] K. Danach, H. Harb, A. Ramadan, and S. Haddad, “Enhancing multi-criteria decision-making in blockchain security: a hybrid machine learning and PROMETHEE approach,” *Engineering Research Express*, vol. 7, no. 3, pp. 1-22, Sep. 2025, doi: 10.1088/2631-8695/ae05eb.
[14] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, “DeepChain: auditable and privacy-preserving deep learning with blockchain-based incentive,” *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 5, pp. 2438-2455, 2021, doi: 10.1109/TDSC.2019.2952332.
[15] X. Lin, J. Wu, A. K. Bashir, J. Li, W. Yang, and M. J. Piran, “Blockchain-based incentive energy-knowledge trading in IoT: joint power transfer and AI design,” *IEEE Internet of Things Journal*, vol. 9, no. 16, pp. 14685–14698, 2022, doi: 10.1109/JIOT.2020.3024246.
[16] V. Mnih *et al*., “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015, doi: 10.1038/nature14236.
[17] B. Han *et al*., “Dynamic incentive design for federated learning based on consortium blockchain using a stackelberg game,” *IEEE Access*, vol. 12, pp. 160267–160283, 2024, doi: 10.1109/ACCESS.2024.3487585.
[18] A. S. Almasoud, “Blockchain-based secure storage and sharing of medical data using machine learning,” in *2023 Tenth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, Abu Dhabi, United Arab Emirates: IEEE, Nov. 2023, pp. 1–4, doi: 10.1109/SNAMS60348.2023.10375435.
[19] M. Seo, J. Kim, M. You, S. Shin, and J. Kim, “gShock: a GNN-based fingerprinting system for permissioned blockchain networks over encrypted channels,” *IEEE Access*, vol. 12, pp. 146328–146342, 2024, doi: 10.1109/ACCESS.2024.3469583.
[20] T.-H. Lee and M.-S. Kim, “RL-MILP solver: a reinforcement learning approach for solving mixed-integer linear programs with graph neural networks,” *5th Annual AAAI Workshop on AI to Accelerate Science and Engineering (AI2ASE)*, pp. 1–10, 2023.
[21] T. Liu and H. Meidani, “End-to-end heterogeneous graph neural networks for traffic assignment,” *Transportation Research Part C: Emerging Technologies*, vol. 165, 2024, doi: 10.1016/j.trc.2024.104695.
[22] P. Almasan, J. S.-Varela, K. Rusek, P. B.-Ros, and A. C.-Aparicio, “Deep reinforcement learning meets graph neural networks: exploring a routing optimization use case,” *Computer Communications*, vol. 196, pp. 184–194, 2022, doi: 10.1016/j.comcom.2022.09.029.
[23] N. S. Kumar, S. Mirdula, P. Singh, T. Gayathri, J. S, and A. M, “A paradigm shift in ethereum network analysis through Google BigQuery,portation, and visualization,” *2024 International Conference on Emerging Smart Computing and Informatics (ESCI)*, pp. 1–7, 2024, doi: 10.1109/ESCI59607.2024.10497269.
[24] A. Day and E. Medvedev, “Ethereum in BigQuery: a public dataset for smart contract analytics,” *Google Cloud Blog*, Accessed: May 27, 2023. [Online]. Available: https://cloud.google.com/blog/products/data-analytics/ethereum-bigquery-public-dataset-smart-contract-analytics
[25] A. Das, G. Uddin, and G. Ruhe, “An empirical study of blockchain repositories in GitHub,” in *The International Conference on Evaluation and Assessment in Software Engineering 2022*, Gothenburg, Sweden: ACM, June 2022, pp. 211–220, doi: 10.1145/3530019.3530041.
[26] Google BigQuery, M. Risdal, S. Dane, and A. Day, “Bitcoin blockchain historical data,” *Kaggle*, 2020, Accessed: May 27, 2023. [Online]. Available: https://www.kaggle.com/datasets/bigquery/bitcoin-blockchain

BIOGRAPHIES OF AUTHORS

Kassem Danach received the Doctor of Philosophy (Ph.D.) degree in Computer Engineering from Ecole Centrale de Lille, France, in 2016. He has applied his expertise in data analytics, artificial intelligence, machine learning, deep learning, educational technology, and business analytics to various military and professional roles. He has made substantial contributions to the field of operational research and artificial intelligence, earning him prestigious awards, including the best paper award at LCIS in 2017 and AISD in 2019, and active participation as a team member in the EU-funded SOCORRO project, from 2019 to 2021. He can be contacted at email: kassem.danach@mu.edu.lb.

Hassan Rkein holds a Ph.D. in Business Administration with a specialization in accounting. He has actively participated in international academic forums. His work emphasizes the importance of aligning academic curricula with the evolving needs of the job market, focusing on skills such as digital literacy, ethical considerations, and problem-solving abilities. Throughout his career, he has demonstrated a commitment to enhancing accounting education and fostering critical discourse on financial practices, making significant contributions to both academic scholarship and practical applications in the field. He can be contacted at email: hassan.rkein@mu.edu.lb.



Alaaeddine Ramadan received his M.S. and Ph.D. in Electronics and Communication Engineering from the University of Limoges, France, in 2007 and 2010, respectively. He has held various academic positions in Lebanon, Bahrain, and France. He has contributed to various Tempus and Erasmus projects, fostering international collaboration in higher education. Currently, he is an associate professor and program lead at the American University of Bahrain, focusing his research on AI, IoT, WSN, and ML. He can be contacted at email: alaaeddine.ramadan@aubh.edu.bh.

Hassan Harb received the Ph.D. degree in computer science from the Lebanese University, Lebanon, and the University of Franche-Comté, France, in 2016. He held postdoctoral positions in Ensta-Bretagne, Artois, and Paris-Saclay Universities from 2018 to 2022. Currently, he is an assistant professor at the American University of the Middle East, Kuwait. His research interests include sensor networks and the IoT, artificial intelligence, healthcare, and modular robotic systems. He can be contacted at email: hassan.harb@aum.edu.kw.

Bassam Hamdar holds a Ph.D. in Applied Economics and International Development. He held academic positions at the American University of Science and Technology (AUST) and the American University of Beirut (AUB) and completed postdoctoral research at the University of Maine. His research encompasses economics, agri-economics, finance, environmental resource management, and marketing. His work includes studies on packaging design's impact on consumer behavior, the role of country-of-origin in consumption patterns, and the economics of nonprofit donor motivation. He can be contacted at email: bassam.hamdar@mu.edu.lb.