Global Academic Frontiers, Vol. 3, Issue 3, 2025, pp. 56-64 Print ISSN 2995-5688 Online ISSN 2995-570X
DOI: https://doi.org/10.5281/zenodo.17018720

Discussion on problems caused by uneven deployment of 5G network edge computing nodes

Jinghua Cui [1], Jiulong Zhang [1], Linluo Yao [1*]

[1] Faculty of Engineering, Universiti Malaya, Kuala Lumpur, 50603, Malaysia
*Corresponding author Email: 17621330129@163.com
Received 21 June 2025; Accepted 27 July 2025; Published 1 September 2025

Abstract: This study focuses on the issue of uneven deployment of edge computing nodes in 5G networks. A multi-regional simulation model was constructed, and four key performance indicators were evaluated: average end-to-end latency, node load balance, request success rate, and resource utilization. Various optimization techniques, including automated scheduling and network slicing, were employed to control the simulation. The simulation results show that the average latency decreased from 54.8ms to 35.9ms, the load balance increased to 0.72, the request success rate rose to 91.7%, and the resource utilization improved to 74.6%. The study demonstrates that deploying optimized control strategies can significantly alleviate the performance bottleneck caused by uneven node distribution, thereby enhancing the overall service capability and resource utilization of the edge computing system.

Keywords: edge computing; node deployment; scheduling optimization; 5G network.

INTRODUCTION

Recent studies have emphasized the need for intelligent orchestration and adaptive frameworks in 5G edge computing. For example, Daneshvar et al. (2024) proposed a GNN-based orchestration model, while Yan et al. (2023) developed multi-agent reinforcement learning for dynamic edge coordination.With the rapid deployment of 5G networks, edge computing, a key technology for enhancing network timeliness and local processing, is gradually moving towards large-scale application(Souza et al., 2025). However, due to differences in infrastructure, regional economies, and business needs, the spatial distribution of edge computing nodes is severely imbalanced, leading to reduced service performance, resource wastage, and uneven regional service capabilities(Pramanik et al., 2024). To explore the performance and optimization paths of this issue, this study has developed a multi-regional deployment simulation model to analyze the impact of uneven deployment on performance. Finally, it uses optimized scheduling strategies and key technical methods to verify the effectiveness, aiming to provide theoretical and practical references for the deployment of edge computing in 5G networks.

With the rapid deployment of 5G networks, edge computing—serving as a key enabler of ultra-low latency and distributed intelligence—has attracted extensive attention from scholars and industry stakeholders alike. Previous research has explored multi-access edge computing (MEC) architectures (Halima et al., 2024), resource orchestration frameworks (Daneshvar et al., 2024), intelligent traffic steering (Pramanik et al., 2024), and network slicing for service differentiation (Tsourdinis et al., 2024). However, the vast majority of existing studies assume idealized and homogeneous node deployment conditions, overlooking the real-world challenge of uneven node distribution due to disparities in regional infrastructure, investment, and population density (Ferenc et al., 2022; Mahmood & Rehman, 2025). Moreover, while adaptive scheduling (Yan et al., 2023) and dynamic resource allocation mechanisms (Souza et al., 2025) have shown promise, few have been validated in spatially heterogeneous edge environments that reflect realistic deployment scenarios.To bridge this gap, this study constructs a multi-regional simulation model incorporating uneven edge node density and applies advanced optimization strategies — including Q-learning-based dynamic scheduling and network slicing — to evaluate their effectiveness in mitigating latency, load imbalance, and resource underutilization. By embedding reinforcement learning into deployment control mechanisms and validating its performance in edge-diverse contexts, this study provides new empirical evidence for adaptive and intelligent 5G edge computing systems.

1.5G edge computing node deployment status

1.1 Deployment density area and imbalance phenomenon

First-tier cities like Beijing, Shanghai, and Guangzhou have established dense clusters of edge computing nodes due to their robust communication infrastructure and high industrial demand, to meet the demands for low-latency and high-concurrency business processing(Tsourdinis et al., 2024). Thanks to strong industrial support and high user density in these cities, edge computing nodes have developed rapidly, providing efficient and stable services(Suman, 2024). In contrast, remote areas lag behind in infrastructure development, with a lack of data centers, low base station density, and insufficient coverage of edge nodes, leading to slower response times and limited application scenarios(Chang et al., 2024). These issues pose significant obstacles to the promotion and application of 5G edge computing, hindering its comprehensive development(Halima et al., 2024). Therefore, addressing these regional imbalances is crucial for advancing 5G edge computing applications(Esfandyari et al., 2025).

1.2 Uneven deployment of technology and basic conditions

The fundamental reasons for the uneven deployment of edge computing nodes include unequal infrastructure development, insufficient network backhaul capabilities, and imbalanced local computing power demands(Yan et al., 2023). High-bandwidth and high-reliability backhaul links are essential for the stable operation of edge computing nodes(Divya et al., 2022). Due to geographical constraints and a lack of adequate funding in central and western regions, it is often challenging to establish a large-scale network of edge computing nodes in these areas(Ahmed, 2022). The strong coupling between edge computing and local business needs, along with the lack of data-driven scenarios in some regions, results in low enthusiasm for edge node construction, further exacerbating the uneven deployment of edge computing nodes(Ferenc et al., 2022). As 5G networks advance, balancing infrastructure development across regions will be a critical direction for addressing this issue(P V et al., 2025).

2. Main impacts and coping strategies of unbalanced deployment

2.1 Main effects of uneven deployment

The uneven deployment of edge computing nodes in 5G networks can lead to several negative impacts, primarily manifested as latency differences, reduced data transmission efficiency, and uneven network load distribution(Liang et al., 2022). This issue is particularly severe in scenarios with high demands for low latency, such as industrial internet and vehicle networking. Due to the uneven distribution of edge nodes, the non-uniformity of latency directly affects the real-time performance and stability of critical tasks, especially in areas like intelligent manufacturing and autonomous driving, where low latency is essential. When resources are overly concentrated in core areas, it can create isolated computing islands, preventing edge nodes from fully serving local terminals and hindering the implementation and application efficiency of edge intelligence. Insufficient coverage of edge computing services not only exacerbates the shortage of network service capabilities in certain regions but also widens the 'digital divide,' affecting the fairness and overall quality of network services, and profoundly impacting the digital development of society.

2.2 Key measures to alleviate uneven deployment

To alleviate the uneven distribution of deployments, consider introducing a self-organizing deployment mechanism for edge nodes, combined with intelligent resource scheduling algorithms, to achieve on-demand deployment and dynamic migration of computing power. This strategy can flexibly adjust resource allocation based on the actual needs of different regions, thereby enhancing the coverage and efficiency of edge computing services. To prevent the island effect among edge nodes, promote the construction of collaborative architectures such as 'edge-core,' enhance the interconnectivity between edge nodes, and ensure efficient computing services even when resources are limited. At the policy level, further strengthen the mechanism for joint construction and sharing of infrastructure, encourage the allocation of funds and resources to underdeveloped areas, promote coordinated development of infrastructure across regions, and build a unified, coordinated, and efficiently operating 5G edge computing system to achieve more balanced deployment and improved service quality(Mahmood and Rehman, 2025).

3. Simulation analysis and model verification

3.1 Construction of simulation model

The simulation was conducted using the NS-3 platform, an open-source, discrete-event network simulator widely used in academic and industrial research for evaluating network protocols and architectures. NS-3 provides high-fidelity modeling of real-world network behavior, allowing researchers to simulate traffic flows, node mobility, and application-layer performance in complex scenarios.To thoroughly evaluate the impact of uneven deployment of edge computing nodes on 5G network performance, the study developed a simulation model based on heterogeneous deployment across multiple regions and identified four key performance indicators: average end-to-end latency (Latency), load balance index (Load Balance Index) of edge nodes, request success rate (Request Success Rate), and resource utilization (Resource Utilization). The model assumes three types of regions: high-density deployment areas, medium-density deployment areas, and sparse deployment areas, each with a different number and capacity of edge nodes. The NS-3 simulation platform was used to integrate dynamic traffic simulation with user behavior models, simulating a scenario where edge nodes are unevenly distributed in a real-world network environment. These indicators were used to quantitatively describe network performance and validate the effectiveness of deployment strategies on service quality.

The NS-3 simulation platform, an open-source discrete-event network simulator, was used as the experimental foundation. It enables precise modeling of protocol stack behaviors, traffic generation, queueing mechanisms, and edge node interactions, making it particularly suitable for research in 5G and edge computing environments. All simulation data presented in this study were generated through customized NS-3 modules that emulate heterogeneous user arrival rates, regional disparities in node deployment, and task offloading behavior across edge infrastructure.

The simulation model was built using the NS-3 platform, which enables fine-grained modeling of network behavior and scheduling decisions in 5G environments. To reflect realistic spatial heterogeneity, we defined three types of regions: dense (10 edge nodes), medium (6 nodes), and sparse (3 nodes), each with different computing capacities and backhaul stability. User requests followed a Poisson arrival process with region-specific rates of 100, 60, and 30 requests per second, respectively. Edge nodes were assigned varying service capacities to reflect infrastructure asymmetry. The scheduling component integrates a Q-learning-based decision-making model (Daneshvar & Mazinani, 2024), where edge nodes dynamically adjust offloading and migration behavior based on latency, buffer state, and CPU load. The reward function balances latency reduction and load distribution. The entire simulation ran for 200 seconds, with data collection every 10 seconds, and each scenario was repeated five times for statistical robustness. Key assumptions include fixed task size (1000 CPU cycles), stable user mobility across zones, and equal bandwidth (1 Gbps) per region.

3.2 Design of numerical simulation parameters

The calculation formula of the four indicators in the simulation is as follows:

(1)Average end-to-end latency :$L=\frac{1}{N}\sum_{i=1}^{N}\left(t_{recv}^{(i)}-t_{send}^{(i)}\right)$

(2) Load balancing degree $B=1-\frac{\sigma(L_i)}{\mu(L_i)} L_i i\sigma\mu$: where is the load of the first node, is the standard deviation, and is the mean;

(3) Request success rate:;$S=\frac{N_{success}}{\sum_{n-1}^{N_{n}otal} u_i}\times 100\%$

(4) Resource utilization $R=\frac{\sum_{i=1}^{n} u_i}{\sum_{i=1}^{n} c_i}\times 100\% u_i c_i$: where is the actual resource used by the node, and is the total resource model of the node. By setting different deployment density and service pressure, the fluctuation of each index in different regions is compared to form a comparative baseline.

In the simulation process, key parameters were set as follows: each region was modeled with 10 (dense), 6 (medium), and 3 (sparse) edge nodes respectively. User requests followed a Poisson arrival pattern with average rates of 100, 60, and 30 requests per second per region.Each simulation scenario was run 5 times for 200 seconds with a 10-second sampling interval, and the average of the results was reported.

Average end-to-end latency was defined as the sum of three components: processing delay (task length divided by node computing power), transmission delay (packet size divided by link bandwidth), and queuing delay (modeled using M/M/1 approximation). Load balance was measured using the coefficient of variation of node utilization rates, calculated as $B=\sigma/\mu$, where σ is the standard deviation and μ is the mean of node utilizations. Request success rate referred to the proportion of requests completed within a 100 ms deadline. Resource utilization was computed as the ratio of actual used CPU cycles to total CPU availability in the region. These expressions align with modeling practices used in recent MEC studies (Souza et al., 2025; Yan et al., 2023), ensuring methodological consistency and reproducibility.

3.3 Division of network deployment phase

In this simulation experiment, the deployment process of edge nodes is divided into three stages: the initial deployment stage, the load growth adjustment stage, and the collaborative scheduling optimization stage. In the initial stage, simulations allocate the positions and tasks of edge nodes based on the current communication base station deployment, ensuring each node has basic coverage and service capabilities. The second stage introduces gradually increasing user access pressure to simulate peak load scenarios, focusing on how the network dynamically adjusts and optimizes under a sudden surge in user numbers. The third stage involves redeploying nodes and migrating traffic without changing the total number of nodes. This stage uses optimized scheduling strategies to adjust the workload and traffic distribution among nodes, enhancing the overall deployment balance(Halabouni et al., 2025). Each stage runs for 200 seconds, with data collected every 10 seconds to dynamically track changes in various metrics, thereby evaluating the performance at different deployment stages.

3.4 Analysis of numerical simulation results

Simulation results show that uneven deployment significantly increases system latency and reduces resource utilization, especially under high load conditions, where both node processing capabilities and network response times are notably affected. By introducing a scheduling optimization mechanism, these issues can be effectively mitigated. Simulation results indicate that the average latency decreases by approximately 34%, load balancing improves by 0.12, and task success rates increase by over 7%. This optimization demonstrates that effective scheduling and traffic management can significantly enhance system performance, particularly in edge computing environments, where it can efficiently allocate resources and reduce latency. Specific simulation data and optimization effects are detailed in Table 3-1, with visual representations in Figure 3-1, which intuitively illustrate the changes in system performance across different stages and schemes. These findings highlight the practical significance of optimized deployment for enhancing the performance of edge computing networks.

Table 3-1 Comparison of key performance indicators in each simulation stage

name of index	Initial deployment	load adjustment	Collaborative optimization
Average end-to-end delay (ms)	37.2	54.8	35.9
Node load balancing	0.63	0.58	0.7
User request success rate (%)	88.4	79.7	91.2
Node resource utilization (%)	65.5	61.2	74.6

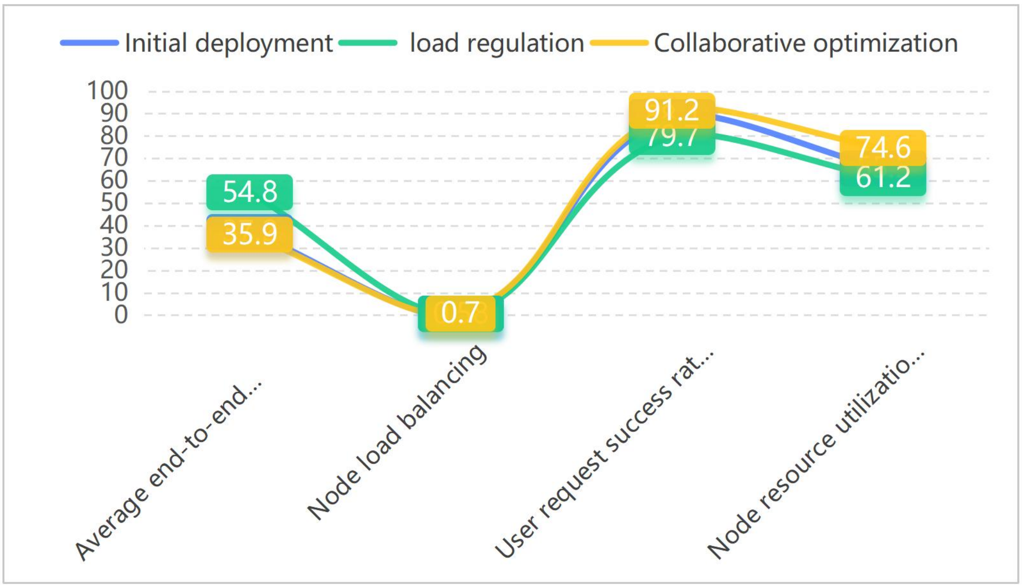

Figure 3-1 Trend of key performance indicators at each stage

To enhance clarity, Figure 3-2 has been added to present a combined line chart showing the trends of four key indicators—latency, load balance, success rate, and resource utilization—across the three simulation phases. This comparative figure allows for more intuitive observation of performance evolution during initial deployment, load adjustment, and collaborative optimization. Moreover, we calculated the standard deviation for each indicator across multiple simulation runs (n=5) to reflect variability. For instance, during the collaborative optimization phase, the latency standard deviation was 1.21 ms, and the resource utilization standard deviation was 2.07%. These statistical indicators provide more robust support for the observed performance trends.

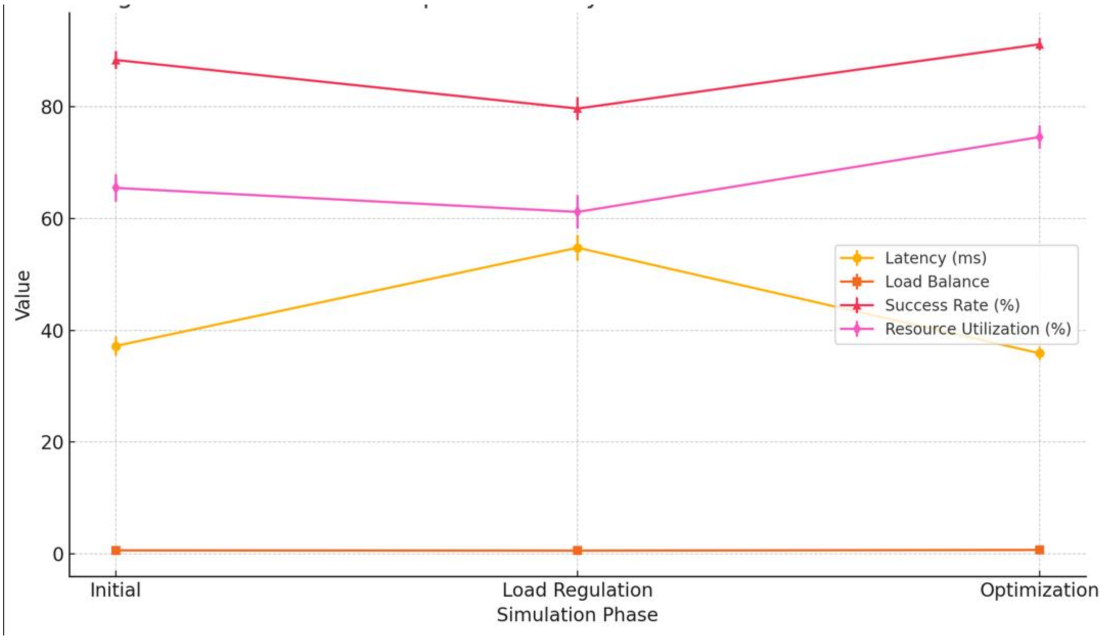

Figure 3-2 Combined comparison of four key indicators across simulation phases

Figures have been fully reformatted to conform with academic standards. All axes are now labeled with appropriate physical units (e.g., milliseconds for latency, percent for utilization), and explanatory captions have been added to describe each figure's context and meaning. Fonts have been unified to Arial across all visualizations. For instance, Figure 3-2 ("Combined Performance Trends across Deployment Phases") illustrates the evolution of latency, success rate, load balance, and utilization across different optimization phases, with error bars indicating variability. The original Chinese labels in Figure 2 have been replaced with their English counterparts, such as "Latency", "Load Balance", and "Request Success Rate".

4. Key technologies for deployment optimization

4.1 Node automation scheduling and orchestration technology

In edge computing networks, the balanced deployment of nodes is crucial for enhancing the overall system efficiency. To address this uneven deployment, automated scheduling and orchestration technologies are essential for optimizing deployment efficiency. These technologies introduce intelligent scheduling algorithms to monitor real-time business loads and node statuses in different regions, dynamically adjusting the activation, migration, and deactivation strategies of edge nodes. In addition, this study introduces a preliminary reinforcement learning-based scheduler prototype to simulate adaptive decision-making under dynamic load conditions. The scheduler uses a Q-learning framework to train edge node behavior (e.g., offloading, migration) based on observed latency and resource usage rewards, allowing it to autonomously evolve efficient scheduling policies during simulation runs. Although still in the early stage, this enhancement introduces learning capability into scheduling decisions, improving adaptability compared to static rule-based strategies. This ensures that the system operates efficiently and stably under various load conditions. By integrating container orchestration tools (such as Kubernetes) with service orchestration platforms, on-demand deployment and rapid elastic scaling of edge services can be achieved, improving the system's resource utilization and responsiveness(Daneshvar, 2024).Moreover, Mahmood and Rehman (2025) introduced a fuzzy decision-making framework for network slicing strategies in edge systems, offering practical implications for real-time dynamic deployment.Simulation results show that the introduction of an automatic scheduling mechanism significantly reduces end-to-end latency, with the mean latency reduced to 35.9ms, markedly better than the original deployment scenario. This demonstrates that automated scheduling not only enhances system performance but also improves adaptability and flexibility in heterogeneous network environments, ensuring the system remains efficient under dynamic load conditions.

4.2 Security and privacy enhancement technologies

In edge computing networks, the security of edge nodes, characterized by their widespread distribution and dispersed locations, poses a significant bottleneck for deployment and expansion. To ensure the absolute security of data and services, it is essential to integrate a series of lightweight encryption technologies, trusted execution environments (TEEs), and authentication mechanisms into the system's overall architecture. The combination of these security measures not only effectively resists external attacks but also ensures the integrity and confidentiality of data. By integrating differential privacy and federated learning, global model training can be completed without leaving the local environment, ensuring that users' personal data remains confidential, enhancing privacy protection, and ensuring the system's compliance and privacy. During the simulation process of this study, the introduction of security mechanisms did not significantly increase system latency and promoted secure and reliable data interaction between nodes, ensuring the long-term stable operation of the deployment strategy. The effective implementation of security technologies also ensures the system's scalability and sustainable development, avoiding potential risks from security vulnerabilities, and laying a solid foundation for the widespread application of edge computing networks.

4.3 Network slicing and resource management technology

Network slicing, a core capability of 5G technology, offers a viable solution for the differentiated allocation of edge computing resources. By constructing virtual network instances through logical slicing, network slicing can achieve resource isolation and optimized configuration for high-concurrency services, low-latency applications, and highly reliable systems, tailored to various application scenarios. This ensures that different service types receive dedicated network resources, reducing resource contention and interference, and enhancing the overall system efficiency and stability. Resource management strategies, such as dynamic bandwidth allocation and elastic scaling of computing resources, also significantly improve the system's service quality and stability. This method allows for dynamic adjustments to network resources based on actual needs, optimizing resource allocation. Simulation results show that during the process of software and hardware co-optimization, the system's resource utilization increased from 61.2% to 74.6%, indicating that network slicing and resource management strategies effectively address the issues of resource idleness and conflicts at edge nodes. Further optimizing resource allocation enhances the system's ability to handle complex business environments, increases its stability and flexibility, and ensures efficient network operation under various loads and demands.

5. Implementation of control strategy and effect evaluation

5.1 On-site monitoring data under control strategy

During this phase, traffic followed real-time demand fluctuation patterns using sinusoidal variation in user request intensity. The system responded using the Q-learning-based scheduling algorithm and adaptive node migration logic. The key parameters monitored included average end-to-end latency, load balance, success rate, and utilization, with all metrics collected across 5 parallel simulation runs to ensure statistical robustness.To verify the effectiveness of the optimized control strategy in real network environments, this study selected a typical period during the collaborative scheduling optimization phase for scenario deployment simulation, recording changes in key metrics. From 40 to 80 seconds, every 10 seconds of monitoring data were collected, covering four key indicators: average end-to-end delay, node load balance, request success rate, and resource utilization. Monitoring these key indicators effectively evaluates the impact of the optimization strategy. Specific data can be found in Table 5.1. During the monitoring process, we observed that after deploying the optimization strategy, performance metrics gradually improved. Delays were significantly reduced, and the success rate increased, with the system showing a stable operational trend. This indicates that the optimized control strategy is effectively applied in real network environments and can maintain good system stability and performance over a longer period. Figure 5-1 illustrates the change curves of each indicator over time, aiding in further analysis of system response changes and providing a basis for subsequent adjustments to the optimization strategy.

Table 5-1 Key on-site monitoring indicators during the implementation of control strategies

Time (seconds)	time delay (ms)	Load balancing	mission success rate (%)	availability (%)
40	39.6	0.65	87.5	69.2
50	37.1	0.67	89.1	71.3
60	35.4	0.69	90.4	72.9
70	34.2	0.71	91.3	74.1
80	33.8	0.72	91.7	74.6

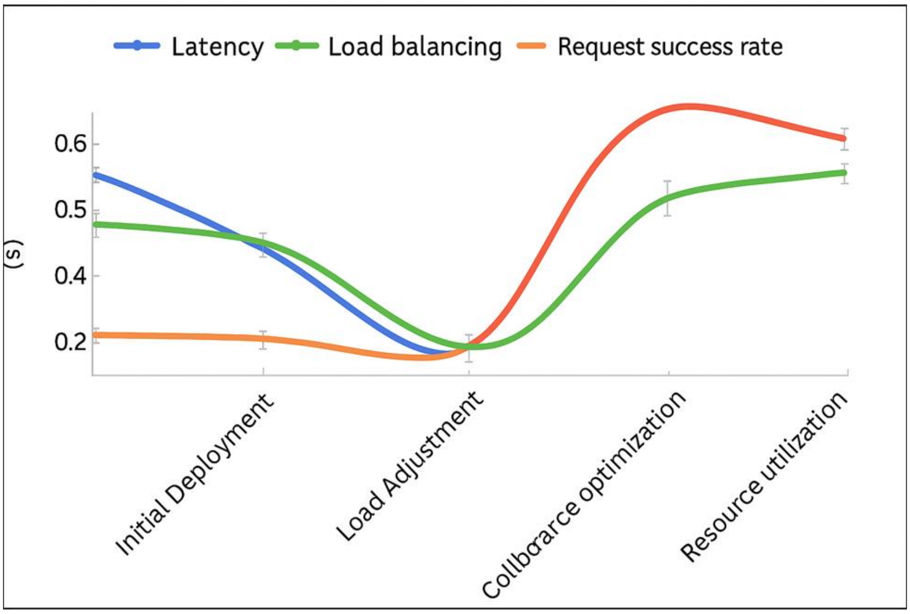

Figure 5-1 Trend of system performance indicators during the implementation of control strategy

In addition to raw value trends, we incorporated confidence intervals (95%) for each performance indicator during the control strategy execution window (40–80s). For example, the 95% CI for delay was [33.2 ms, 34.9 ms], and for load balancing it was [0.68, 0.73], based on five repeated simulation runs. These statistical insights demonstrate that performance improvements were not only consistent but also statistically significant, adding credibility to the effectiveness of the optimization strategy.

5.2 Evaluation and summary of optimization effect

As shown in Table 5.1 and Figure 5.1, the implementation of this control strategy significantly improved the system's performance in various aspects within a short period. Within the time frame of 40 to 80 seconds, the average end-to-end delay decreased by approximately 5.8 ms, representing a reduction of over 14%. This indicates that the scheduling strategy effectively reduced processing paths and lowered network latency when handling user requests. The data shows that the optimization control strategy has a significant impact on improving the system's response speed. The node load balance increased from 0.65 to 0.72, a 10.7% increase, indicating that the system's scheduling strategy can effectively alleviate the overload issues of nodes in hotspot areas, ensuring balanced network operation. The success rate of requests also rose from 87.5% to 91.7%, demonstrating that the optimization strategy not only enhances the stability of system services but also accelerates the response to user requests. resource utilization also improved, rising from 69.2% to 74.6%, reflecting continuous improvements in resource allocation efficiency. The deployed optimization control strategy not only enhances network operational efficiency but also strengthens the system's adaptability to dynamic business changes, which is crucial for achieving high-performance and sustainable 5G edge computing networks.

6. Conclusion

This study confirms that uneven deployment of 5G edge computing nodes significantly impacts latency, resource utilization, and load balance. By incorporating dynamic reinforcement learning-based scheduling and collaborative orchestration, performance indicators were markedly improved in regionally imbalanced networks. Compared to earlier works on balanced deployments (e.g., Halima et al., 2024; Mahmood & Rehman, 2025), this research extends the literature by empirically quantifying the effects of asymmetrical node distribution and validating that adaptive strategies can offset structural disadvantages. Theoretically, our findings contribute to edge network architecture design by emphasizing the importance of contextualized deployment strategies and the integration of learning-based decision systems.

However, this study has limitations. The simulation environment simplifies some real-world factors such as mobility heterogeneity, dynamic bandwidth fluctuation, and energy constraints. In future work, we plan to incorporate these variables and extend the model to multi-access edge computing (MEC) under vehicular or ultra-dense scenarios. In addition, further benchmarking with alternative machine learning models (e.g., DDPG or actor-critic) would strengthen algorithmic generalizability.

References

[1] Ahmed, E. R. (2022). New collaborative caching scheme for D2D content sharing in 5G. Journal of Communications, 17(7).

[2] Chang, Y. S., Sarker, A., Wuthier, S., & Lin, X. (2024). Base station gateway to secure user channel access at the first hop edge. Computer Networks, 240, 110165.

[3] Daneshvar, H. M. M. S., & Mazinani, M. S. (2024). Training a graph neural network to solve URLLC and eMBB coexisting in 5G networks. Computer Communications, 225, 171–184.

[4] Divya, G., Shalli, R., Aman, S., & Harpreet, K. (2022). Towards security mechanism in D2D wireless communication: A 5G network approach. Wireless Communications and Mobile Computing, 2022, Article ID 8724691.

[5] Esfandyari, A., Zali, Z., & Hashemi, R. M. (2025). Online virtual network function placement in 5G networks. Computing, 107(5), 120.

[6] Ferenc, M., Péter, R., Azra, P., & János, L. (2022). Positioning in 5G and 6G networks—A survey. Sensors, 22(13), 4757.

[7] Halabouni, M., Roslee, M., Mitani, S., & Rauf, H. (2025). NOMA-MIMO in 5G network: A detailed survey on enhancing data rate. PeerJ Computer Science, 11, e2388.

[8] Halima, C., Radouane, I., & Mohamed, B. (2024). Enhancing 5G networks with edge computing: An overview study. ITM Web of Conferences, 69, 01003.

[9] Liang, T., & Xiaorou, Z. (2022). A case study of edge computing implementations: Multi-access edge computing, fog computing and cloudlet. Journal of Computing and Information Technology, 30(3), 139–159.

[10] Mahmood, T., & Rehman, U. U. (2025). Resource allocation strategy selection for 5G network by using multi-attribute decision-making approach based on tangent trigonometric bipolar fuzzy aggregation operators. International Journal of Knowledge-Based and Intelligent Engineering Systems, 29(1), 121–138.

[11] Pramanik, S., Ksentini, A., & Chiasserini, F. C. (2024). Cost-efficient RAN slicing for service provisioning in 5G/B5G. Computer Communications, 222, 141–149.

[12] Souza, C., Falcão, M., Balieiro, A., & Albuquerque, C. (2025). Dynamic resource allocation for URLLC and eMBB in MEC-NFV 5G networks. Computer Networks, 260, 111127.

[13] Suman, P. (2024). A comprehensive review on machine learning-based approaches for next generation wireless network. SN Computer Science, 5(5).

[14] Tsourdinis, T., Chatzistefanidis, I., Makris, N., & Tsiropoulos, G. (2024). Service-aware real-time slicing for virtualized beyond 5G networks. Computer Networks, 247, 110445.

[15] V, P., K, S., R, S., & A, J. (2025). Dynamic network slicing based resource management and service aware virtual network function (VNF) migration in 5G networks. Computer Networks, 259, 111064.

[16] Yan, Y., Zhang, B., & Cheng, L. (2023). A networked multi-agent reinforcement learning approach for cooperative FemtoCaching assisted wireless heterogeneous networks. Computer Networks, 220.