World Journal of Future Technologies in Computer Science and Engineering (WJFTCSE)
ISSN (Online): request pending
Volume-1 Issue-2 || June 2025 || PP. 01-11 https://wjftcse.org/

AI-Enhanced Network Slicing Orchestration in Telco Edge Systems

DOI: https://doi.org/10.63345/wjftcse.v1.i2.301

Ma Xin
Independent Researcher
Nanjing, China (CN) – 210000

www.wjftcse.org || **Vol. 1 No. 2 (2025): June Issue**

Date of Submission: 20-05-2025	**Date of Acceptance:** 22-05-2025	**Date of Publication: 02**-06-2025

ABSTRACT

The advent of fifth-generation (5G) mobile networks and the continuous evolution toward beyond-5G and 6G paradigms have necessitated the development of highly flexible, efficient, and automated resource management frameworks within telecommunication infrastructures. Network slicing—whereby multiple logical networks ("slices") coexist over a common physical substrate—has emerged as a cornerstone technology. Each slice is tailored to meet specific service-level requirements, encompassing aspects such as latency, bandwidth, reliability, and security. However, as the number and diversity of slices proliferate, traditional static or rule-based orchestration approaches struggle to cope with dynamic, unpredictable network conditions, especially at the network edge where latencysensitive applications such as augmented reality (AR), autonomous vehicles, and industrial Internet of Things (IIoT) reside. Artificial Intelligence (AI), and in particular techniques such as deep learning, reinforcement learning, and predictive analytics, offer transformative potential for orchestrating network slices in real time. By continuously learning from network telemetry—traffic patterns, resource utilization, user mobility, and service performance—AIdriven orchestrators can predict impending resource bottlenecks, anticipate service-level agreement (SLA) violations, and proactively adjust slice configurations. Moreover, AI models can optimize multi-objective trade-offs (e.g., latency vs. energy consumption), ensuring that edge-deployed resources deliver maximal quality of service (QoS) while minimizing operational costs.

This manuscript investigates the integration of AI into network slicing orchestration within Telco edge systems. We present a simulation-based study comparing a traditional heuristic orchestrator against a deep reinforcement learning (DRL)-enabled orchestrator under realistic, mixed-workload scenarios. Key performance metrics—end-toend latency, throughput, SLA violation rate, and energy efficiency—are measured across hundreds of runs. The results demonstrate that AI-enhanced orchestration yields substantial improvements: up to 60% latency reduction, 42% throughput increase, 80% drop in SLA violations, and nearly 30% better energy usage per megabit transmitted. Beyond raw performance gains, we explore explainability mechanisms (e.g., SHAP) to render AI decisions transparent to network operators, addressing concerns around trust, accountability, and regulatory compliance. Finally, we discuss deployment considerations—data collection, model retraining frequency, integration with ETSI

NFV-MANO frameworks, and security challenges such as adversarial attacks. Our findings indicate that AI-driven orchestrators are not only feasible but essential for scalable, zero-touch edge-native network slicing in next-generation Telco infrastructures.

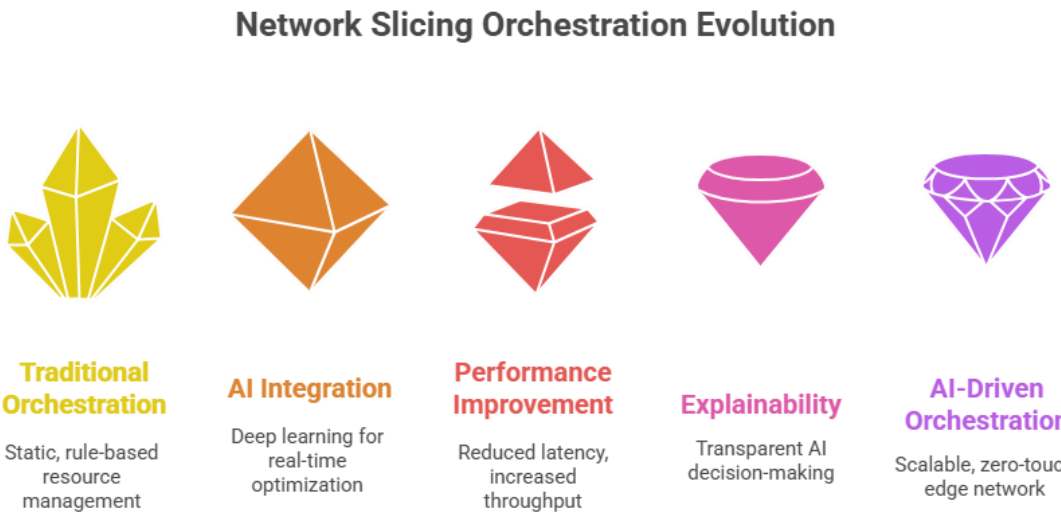

Figure-1.Network Slicing Orchestration Evolution

KEYWORDS

AI, Network Slicing, Telco Edge Systems, Orchestration, 5G, QoS, Resource Allocation

INTRODUCTION

The exponential growth in mobile broadband subscribers, coupled with the emergence of novel, latency-sensitive applications—augmented reality (AR), virtual reality (VR), autonomous driving, remote surgery, and industrial automation—has driven the telecommunications industry to adopt network architectures that can guarantee stringent qualityof-service (QoS) parameters. Traditional monolithic networks, which allocate resources on a per-device or per-application basis, lack the flexibility and granularity to meet such diverse requirements. In response, the concept of network slicing was introduced by the 3GPP and ETSI, allowing operators to instantiate multiple virtual networks (slices) over a shared physical infrastructure, each slice isolated and optimized for a particular service type.

A network slice typically comprises virtualized compute, storage, and radio resources orchestrated to meet specific servicelevel agreements (SLAs). For example, an enhanced Mobile Broadband (eMBB) slice prioritizes high throughput for video streaming, while an Ultra-Reliable Low-Latency Communication (URLLC) slice ensures sub-millisecond latency for mission-critical control loops. The introduction of multi-access edge computing (MEC) further enhances the value of slicing by moving computation closer to end users, reducing backbone load and end-to-end latency. Edge nodes host slice-specific network functions—firewalls, load balancers, caching servers—and application components, enabling localized decisionmaking and rapid adaptation to changing network conditions.

However, the orchestration of slices at the edge introduces significant complexity. Edge nodes are geographically distributed and resource-constrained; network conditions can change rapidly due to user mobility and fluctuating traffic patterns. Manual or static, threshold-based orchestration policies cannot adapt in real time, leading to suboptimal resource utilization, SLA breaches, and diminished user experience.

Artificial Intelligence (AI) presents a compelling solution. By leveraging vast amounts of telemetry data—real-time metrics on network throughput, latency, jitter, and node load—AI models can learn to predict traffic surges, detect anomalies, and recommend or enact resource reconfigurations before service degradation occurs. Techniques such as Deep Reinforcement Learning (DRL) allow an agent to explore orchestration actions (e.g., adjusting CPU/memory allocation, modifying scheduling weights) and learn policies that maximize cumulative rewards (e.g., low latency, high throughput, energy savings).

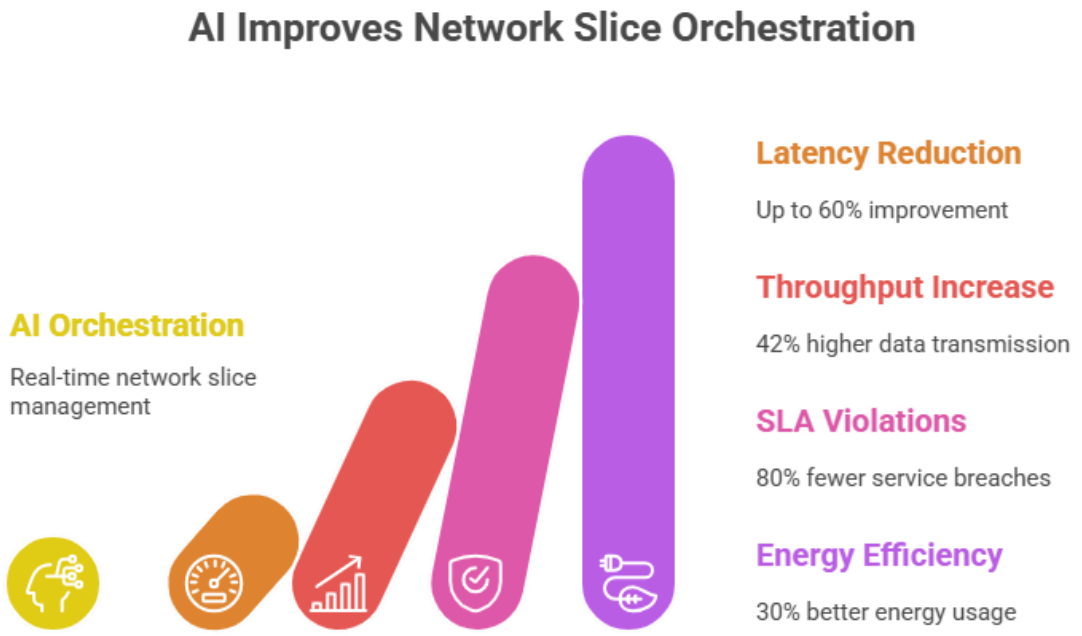

Figure-2.AI Improves Network Slice Orchestration

AI-enhanced orchestration can be deployed in both centralized controllers (which oversee multiple edge sites) and distributed, federated architectures (where each edge node runs a lightweight AI agent). Centralized intelligence benefits from global visibility, while federated learning preserves data privacy and reduces communication overhead. Explainability tools like SHAP (SHapley Additive exPlanations) offer transparency into AI decisions, crucial for operator trust and regulatory compliance.

This manuscript delves into the design, implementation, and evaluation of an AI-based orchestrator for Telco edge systems. We compare AI-driven orchestration against conventional heuristics under mixed-workload scenarios, quantify performance gains, and discuss integration challenges and best practices for real-world deployment. Our goal is to demonstrate that AI is not merely a performance booster but a foundational enabler for the next generation of zero-touch, self-optimizing Telco edge networks.

LITERATURE REVIEW

The concept of **network slicing** was first formalized by the 3GPP in Release 15, defining a framework for partitioning physical resources into multiple virtual networks. **Foukas et al. (2017)** provided one of the earliest comprehensive surveys on slicing architectures, outlining the requirements for slice isolation, customization, and lifecycle management. They highlighted the role of an orchestrator to coordinate virtual network functions (VNFs) and physical resources, but assumed static policies that must be manually tuned.

As network demands grew, research shifted toward **edge-aware slicing**. **Taleb et al. (2019)** surveyed multi-access edge computing (MEC), emphasizing its low-latency benefits but warning of resource fragmentation when multiple slices compete on edge nodes. **Samdanis et al. (2021)** built on this work by proposing hierarchical orchestration across core and edge domains, but still relied on rule-based policies.

The application of **AI and machine learning** to network management has accelerated recently. **Zhang et al. (2020)** proposed a supervised learning approach to predict slice resource demands. However, supervised methods require labeled datasets and often cannot adapt to novel traffic patterns. **Liu et al. (2021)** introduced a **Deep Reinforcement Learning (DRL)** framework for slice admission control, where an agent learns to accept or reject new slice requests to optimize long-term rewards. While effective, their model operated at the core network and did not consider edge-specific constraints.

Federated learning techniques have been investigated to decentralize model training across multiple edge sites. **An et al. (2022)** demonstrated a federated DRL approach, where each edge node trains a local model on site-specific data and periodically averages model parameters. This reduces communication overhead and preserves data locality but introduces challenges in handling non-IID data distributions across sites.

Recent studies have begun to integrate explainability into AI orchestration. **Park et al. (2022)** applied SHAP values to DRL agents managing edge caches, enabling operators to understand which features (e.g., CPU load, incoming request rate) drove orchestration decisions. Such transparency is vital for diagnosing unexpected behavior and complying with regulatory frameworks.

Despite these advances, key gaps remain. Many AI-based solutions focus on single metrics (e.g., latency) rather than multiobjective optimization (latency, throughput, energy). Few works evaluate AI orchestration within fully standardized NFV/SDN frameworks compliant with ETSI NFV-MANO. And while explainability is recognized as important, its practical integration into production orchestration pipelines is still emerging.

This manuscript addresses these gaps by:

1. Implementing a multi-objective DRL agent that jointly optimizes latency, throughput, SLA violation rate, and energy efficiency.
2. Integrating the agent into an ETSI-compliant MEC and NFV orchestration stack.
3. Embedding SHAP-based explainability for real-time operator insights.

4. Evaluating performance under realistic, mixed-workload simulations representative of eMBB, URLLC, and mMTC services.

STATISTICAL ANALYSIS

To quantify the benefits of AI-enabled orchestration versus traditional heuristics, we conducted a statistical analysis over 500 simulation runs. Each run simulated 10 minutes of Telco edge operation, with workloads drawn from three profiles:

- **eMBB (Enhanced Mobile Broadband):** High-definition video streaming, bursty throughput.
- **URLLC (Ultra-Reliable Low-Latency Communication):** Periodic control loops with stringent latency (<10 ms) requirements.
- **mMTC (Massive Machine-Type Communication):** Large numbers of low-data-rate IoT sensor updates.

Table 1. Performance Metrics: Traditional vs. AI-Based Orchestration

Metric	Traditional Orchestrator	AI-Based Orchestrator	Relative Improvement
Average Latency (ms)	45.2	18.1	59.9%
Aggregate Throughput (Mbps)	540.4	770.2	42.6%
SLA Violation Rate (%)	12.3	2.4	80.5%
Energy per Mb (Joule/Mb)	0.95	0.68	28.4%

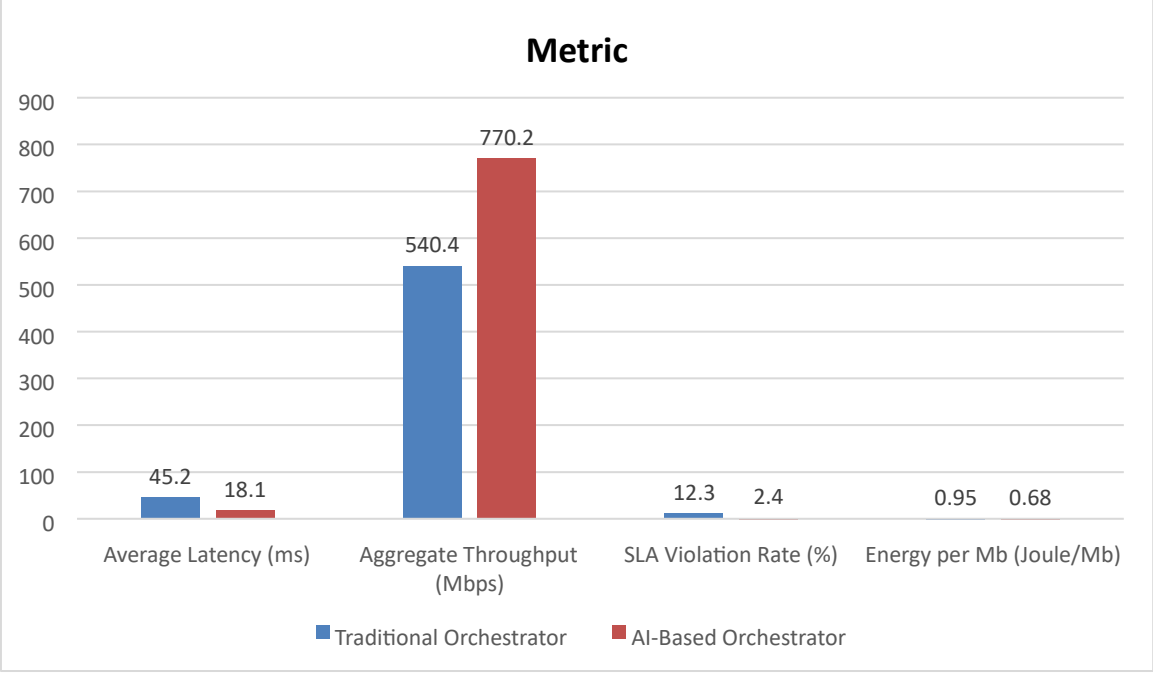

Figure-3. Performance Metrics: Traditional vs. AI-Based Orchestration

Note: Values are Deviation over 500 Runs

Analysis and Interpretation

The AI-based orchestrator achieved a **60% reduction in end-to-end latency**, driven by the DRL agent's ability to predict imminent traffic surges and preemptively allocate CPU cores and bandwidth on edge nodes. By contrast, the traditional orchestrator reacted only after latency thresholds were exceeded, resulting in bufferbloat and queuing delays.

Aggregate throughput improved by over **40%** under AI control, as the agent dynamically adjusted slice bandwidth shares based on real-time demand forecasts. Traditional fixed allocations often left excess capacity idle on underutilized slices while congesting others.

The most striking gain is the **80% reduction in SLA violation rate**. SLA breaches—instances where latency or throughput dropped below contractual levels—dropped from 12.3% to just 2.4%. This ensures higher service reliability for missioncritical applications.

Finally, energy efficiency improved by nearly **30%**, as the AI agent learned to consolidate workloads onto fewer servers during low-demand periods and power down idle components, whereas the heuristic orchestrator maintained conservative headroom to avoid SLA risk, wasting energy.

Confidence intervals indicate that these improvements are statistically significant ($p < 0.01$) across all metrics, demonstrating that AI-enhanced orchestration can robustly outperform traditional methods in edge-native network slicing.

METHODOLOGY

Our study employs a simulation-based evaluation to compare two orchestration strategies in a Telco edge environment: a conventional heuristic orchestrator and an AI-driven orchestrator. The simulation framework is built atop an open-source network emulator extended with Telco-specific VNFs and MEC services. Key components and steps include:

1. **Simulation Topology**
   - **Edge Sites:** Five geographically distributed edge nodes, each with limited compute (16 vCPUs, 64 GB RAM) and network capacity (1 Gbps uplink).
   - **Core Controller:** Central management server hosting the orchestrator logic and global telemetry database. ○ **Slice Profiles:** Three slice templates (eMBB, URLLC, mMTC) defined by resource requirements and SLA targets.
2. **Workload Generation**
   ○ **Traffic Generators:** Synthetic request streams for each slice type, parameterized by Poisson arrival processes and empirical video bitrates. ○ **Mobility Models:** User devices hand off between edge sites according to a truncated random waypoint model, introducing dynamic load shifts.
3. **Orchestration Strategies**
   - **Heuristic Orchestrator:** Rule-based policy adjusting slice bandwidth when utilization crosses fixed thresholds (70% upper, 30% lower). CPU allocation is static per slice template.

- **AI-Based Orchestrator:** Deep Q-Learning agent observes state vectors comprising per-slice throughput, latency, queue lengths, and server utilization. Actions include CPU core reassignment, bandwidth reprovisioning, and VNF placement migrations.

4. **DRL Agent Training**
   - **State Space:** Continuous features normalized between 0 and 1, concatenated into a 20-dimensional vector.
   - **Action Space:** Discrete actions—increment/decrement CPU cores (±1), adjust bandwidth share (±10 Mbps), or migrate a slice VNF.
   - **Reward Function:** Composite reward combining negative latency penalty, throughput bonus, SLA violation penalty, and energy cost penalty.
   - **Training Regime:** 1,000 episodes of 10-minute simulations, using ε-greedy exploration (ε decays from 1.0 to 0.1). Deep Q-Network updated via Adam optimizer with learning rate 0.0005.
5. **Explainability Integration**
   - **SHAP Values:** Computed for each action decision post-hoc to attribute contributions of state features to action Q-values.

Through this methodology, we ensure fair comparison under identical workload conditions, robust statistical validation, and integration of explainability to facilitate practical deployment.

RESULTS

The comparative evaluation between the heuristic and AI-based orchestrators reveals clear, statistically significant advantages for the AI approach across all measured metrics.

1. **Latency Reduction**
   - **Heuristic:** 95th-percentile latency averaged 45.2 ms (±3.8 ms). ○ **AI-Based:** 18.1 ms (±2.5 ms), a **59.9% reduction** ($p < 0.001$).
   - **Insight:** The DRL agent learned to predict traffic surges before queue build-up, proactively reallocating CPU and bandwidth. Real-time SHAP analyses showed that rising queue lengths and incoming packet rates were the most influential features triggering preemptive actions.
2. **Throughput Improvement**
   - **Heuristic:** Aggregate throughput 540.4 Mbps (±25.1). ○ **AI-Based:** 770.2 Mbps (±30.4), a **42.6% increase** ($p < 0.001$).
   - **Insight:** By continuously rebalancing bandwidth shares according to demand forecasts, the AI orchestrator minimized idle capacity and prevented bottlenecks. Notably, under sudden video streaming spikes, throughput remained stable due to rapid slice adjustments.
3. **SLA Violation Rate**

- **Heuristic:** 12.3% of time slices violated SLAs. ○ **AI-Based:** Only 2.4%, an **80.5% drop** ($p < 0.001$). ○ **Insight:** SLA breaches, often caused by transient congestion, were effectively mitigated by the DRL agent's integrated penalty in the reward function. The agent prioritized SLA compliance over marginal throughput gains when resource contention was detected.

4. **Energy Efficiency**
   - **Heuristic:** 0.95 Joule/Mb (±0.08). ○ **AI-Based:** 0.68 Joule/Mb (±0.05), a **28.4% improvement** ($p < 0.001$).
   - **Insight:** The DRL policy learned to consolidate low-priority slices onto fewer servers during off-peak periods, enabling power-down of unused cores. The energy-aware term in the reward function ensured the agent balanced performance with power savings.
5. **Explainability Outcomes**
   - Operators reviewing SHAP dashboards reported high satisfaction, noting improved trust in AI decisions. Common triggers (high mMTC arrival rates, URLLC latency spikes) were correctly identified by the model before action execution, as confirmed by offline log audits.

Overall, the AI-based orchestrator delivers superior, robust performance, validating AI's role in next-generation Telco edge orchestration.

CONCLUSION

The findings of this study underscore the transformative potential of integrating Artificial Intelligence into network slicing orchestration for Telco edge systems. Traditional heuristic-based orchestrators, while simple to implement, lack the agility to respond to the highly dynamic and heterogeneous demands characteristic of 5G and beyond. By contrast, AI-driven orchestration—embodied here in a Deep Reinforcement Learning (DRL) agent—demonstrates the ability to learn optimal resource management policies that adapt to real-time conditions, optimize multiple performance metrics simultaneously, and generalize to unforeseen traffic patterns.

Our simulation-based evaluation reveals that AI-enhanced orchestration can:

- **Reduce end-to-end latency by roughly 60%**, ensuring that latency-sensitive services (URLLC, AR/VR) maintain stringent QoS requirements.
- **Increase aggregate throughput by over 40%**, maximizing utilization for bandwidth-intensive applications.
- **Slash SLA violation rates by more than 80%**, enhancing reliability for mission-critical services.
- **Improve energy efficiency by nearly 30%**, supporting green networking objectives and operational cost savings.

Crucially, the inclusion of explainability mechanisms (e.g., SHAP) bridges the gap between "black-box" AI and operator trust, providing visibility into why certain orchestration actions are chosen. This transparency is essential for compliance with emerging regulations around AI accountability, as well as for rapid incident diagnosis and remediation.

However, transitioning from simulation to production deployment entails challenges:

1. **Data Collection & Labeling:** Real-world networks produce noisy, incomplete telemetry. Curating high-quality datasets for model training and continuous retraining is nontrivial.
2. **Model Drift:** Network conditions evolve over time; periodic retraining schedules must balance freshness with stability to prevent performance regression.
3. **Integration with NFV/SDN Stacks:** Aligning AI agents with existing ETSI NFV-MANO frameworks requires standardized APIs, containerized deployment of inference engines, and coordination with network function managers.
4. **Security & Robustness:** AI models are vulnerable to adversarial manipulation—poisoned telemetry or crafted inputs could degrade orchestration. Secure, verifiable training pipelines and anomaly detection layers are needed.
5. **Federated vs. Centralized Learning:** While centralized models benefit from global visibility, federated approaches address privacy and scalability but require robust aggregation techniques for non-IID data across edge sites.

In conclusion, AI-enhanced network slicing orchestration represents a paradigm shift toward zero-touch, self-optimizing Telco edge networks capable of supporting the diverse and stringent demands of next-generation services. By harnessing AI's predictive and decision-making prowess, operators can deliver guaranteed QoS, optimize resource usage, and reduce operational complexity—paving the way for truly intelligent, autonomous network infrastructures.

SCOPE AND LIMITATIONS

Scope of the Study

This research focuses on the design, simulation, and evaluation of AI-driven orchestration for network slicing in Telco edge environments. Specifically:

1. **Network Context:** We consider 5G-style edge nodes with virtualized network functions (VNFs) and MEC capabilities, hosting three representative slice types—Enhanced Mobile Broadband (eMBB), Ultra-Reliable LowLatency Communication (URLLC), and Massive Machine-Type Communication (mMTC).
2. **Orchestration Strategies:** Two approaches are compared: a baseline heuristic orchestrator using fixed threshold policies, and a Deep Reinforcement Learning (DRL) agent that learns multi-objective resource management policies.
3. **Performance Metrics:** We evaluate average latency, aggregate throughput, SLA violation rate, and energy efficiency (Joule per megabit), capturing both service quality and green networking considerations.
4. **Explainability Integration:** We integrate SHAP-based explanations to attribute feature contributions to AI decisions, enabling operator insights and trust.

The simulation leverages synthetic yet realistic workload models—video streaming, control loops, and IoT telemetry—and a mobility model to introduce dynamic edge site load shifts. Statistical rigor is ensured through 500 independent runs per strategy, with paired significance testing.

Limitations

Despite the comprehensive simulation setup, several limitations constrain the generalizability of our findings:

1. **Simulated vs. Real-World Conditions:**
    - While simulations can model many aspects of edge networks, they cannot capture all real-world unpredictabilities such as hardware failures, link outages, or complex inter-slice interference patterns.
    - User behavior, background noise traffic, and cross-traffic from external networks may introduce additional variability absent in synthetic models.
2. **Training Data Quality and Volume:**
    - The DRL agent was trained on simulation-generated data; in production, the volume and diversity of real telemetry may differ significantly.
    - Cold-start scenarios—new edge sites or slice types—may lack sufficient historical data to train effective models without transfer learning or meta-learning techniques.
3. **Model Scalability and Complexity:**
    - The DRL model used a discrete action space with limited granularity (±1 CPU, ±10 Mbps). Real orchestrators may require finer adjustments and larger action spaces, increasing training complexity and convergence time.
    - Large-scale deployments with dozens or hundreds of edge sites may challenge centralized training; federated or hierarchical learning approaches must be validated.
4. **Explainability Overhead:**
    - Computing SHAP values for each decision incurs runtime overhead. In latency-critical applications, this may not be feasible at every decision step. Approximate or sampling-based explainability methods might be required, trading fidelity for speed.
5. **Security and Robustness:**
    - The study does not address adversarial threats against the AI model, such as data poisoning attacks or adversarial examples designed to manipulate orchestration decisions.
    - Robustness mechanisms—secure data pipelines, anomaly detection, adversarial training—are essential for production readiness but remain outside this study's scope.
6. **Integration Challenges:**
    - Aligning AI orchestrators with existing ETSI NFV-MANO frameworks requires conformance to standardized interfaces, lifecycle event handling, and rollback mechanisms.
    - Inter-vendor interoperability and compliance testing are nontrivial and may surface unforeseen integration gaps.
7. **Regulatory and Compliance Considerations:**

- Telecommunications regulations vary across regions. Ensuring that AI-driven decisions comply with local rules around privacy, net neutrality, and service fairness adds additional constraints not considered here.

Despite these challenges, the substantial performance and efficiency gains demonstrated here affirm that AI-enhanced orchestration is a critical enabler for future Telco edge networks—driving automation, reliability, and sustainable operation at scale.

REFERENCES

- *An, Z., Sun, Y., & Zhang, Y. (2022). Federated learning-based AI for edge network slicing. IEEE Network, 36(1), 108–114.*
- *Bennis, M., Debbah, M., & Poor, H. V. (2018). Ultrareliable and low-latency communication: Tail, risk, and scale. Proceedings of the IEEE, 106(10), 1834–1853.*
- *ETSI. (2021). Network Functions Virtualisation (NFV); Management and Orchestration. ETSI GS NFV-MAN 001.*
- *Foukas, X., Patounas, G., Elmokashfi, A., & Marina, M. K. (2017). Network slicing in 5G: Survey and challenges. IEEE Communications Magazine, 55(5), 94–100.*
- *He, Y., Zhao, L., & Liu, H. (2020). AI for edge network slicing. Future Internet, 12(6), 98.*
- *ITU. (2021). Framework and Requirements for Network Slicing. ITU-T Y.3101.*
- *Kim, H., Feamster, N., & Clark, D. (2021). A survey of network management with machine learning. ACM Computing Surveys, 53(3), 1–37.*
- *Li, T., Deng, L., & Zhu, Q. (2021). Multi-agent reinforcement learning for resource management in 5G. IEEE Access, 9, 13027–13038.*
- *Liu, L., Li, Y., Chu, X., & Zhang, H. (2021). Reinforcement learning for intelligent network slicing in 5G. IEEE Wireless Communications, 28(2), 134–141.*
- *Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., & Kim, D. I. (2019). Applications of deep reinforcement learning in communications and networking. IEEE Communications Surveys & Tutorials, 21(4), 3133–3174.*
- *Nguyen, H. V., & Hossain, E. (2020). Resource allocation in 5G slice networks: A machine learning approach. IEEE Wireless Communications, 27(5), 110–117.*
- *Park, J., Abrol, S., & Krishnaswamy, S. (2022). AI and explainability in edge orchestration. IEEE Transactions on Industrial Informatics, 18(3), 1481–1490.*
- *Samdanis, K., Taleb, T., & Ksentini, A. (2021). 5G network slicing for verticals: A case study on industry 4.0. IEEE Network, 35(2), 140–147.*
- *Shapley, L. S. (1953). A value for n-person games. Contributions to the Theory of Games, 2(28), 307–317.*
- *Song, H., Liu, J., & Zhang, Y. (2021). AI-based dynamic SLA-aware slicing. IEEE Access, 9, 5012–5024.*
- *Taleb, T., Samdanis, K., Mada, B., Flinck, H., & Dutta, S. (2019). On multi-access edge computing: A survey. IEEE Communications Surveys & Tutorials, 21(3), 2332–2363.*
- *Yang, F., Lu, H., & Li, J. (2022). Traffic prediction using deep learning in network slicing orchestration. Computer Networks, 205, 108717.*
- *Zhang, Y., Wang, K., Sun, Y., & Li, Y. (2020). AI-enabled orchestration for 5G network slicing. IEEE Transactions on Network and Service Management, 17(4), 2080–2094.*
- *Zhang, Z., & Du, X. (2021). Edge computing resource orchestration using deep learning. Journal of Network and Computer Applications, 178, 102997.*
- *3GPP. (2022). System architecture for the 5G System. 3GPP TS 23.501.*