Digital Twin-Based Network Slice Optimization Using Deep Reinforcement Learning for Next-Generation Mobile Networks

Alessandro Gregory
Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY, USA.
alessandro.gregory@buffalo.edu

Feizhen Yang
School of Information Technology, University of Cincinnati, Cincinnati, OH, USA.
feizhenyang@uc.edu

Yichenfan Cai
Department of Computer Science, University of Houston, Houston, TX, USA.
yichenfanwork@uh.edu

Abstract

Next-generation mobile networks require dynamic and fine-grained resource management to accommodate highly diverse service requirements under stringent performance constraints. Network slicing, underpinned by digital twin technology and deep reinforcement learning, presents a transformative paradigm for autonomous slice optimization at scale. This paper presents a comprehensive system-level examination of digital twin-based network slice optimization using deep reinforcement learning, focusing on architectural design, deployment challenges, governance, sustainability, robustness, and fairness. We propose a closed-loop control architecture in which a high-fidelity digital twin continuously mirrors the physical radio access and core network infrastructure, enabling proactive, conflict-free resource allocation. The twin's predictive capabilities are combined with a centralized policy learning agent that uses proximal policy optimization to resolve the complex combinatorial resource allocation problem while respecting strict service-level agreements. Through an analysis of system integration, edge-cloud orchestration, and twin synchronization latency, the paper delineates the structural trade-offs between real-time responsiveness and modeling fidelity. Further, we discuss multi-tenant governance, regulatory implications, and the role of digital twins in ensuring transparent, auditable slice management. The study addresses robustness against twin model drift and adversarial perturbations, as well as fairness criteria that prevent resource starvation of lower-priority slices. An extended discussion covers energy footprint accounting, lifecycle carbon costs, and the alignment of slice optimization with sustainable networking initiatives. By synthesizing cross-domain insights from communications, artificial intelligence, distributed systems, and public policy, this work offers a forward-looking roadmap for deploying responsible, resilient, and equitable digital twin-driven network slicing platforms in 5G-Advanced and 6G ecosystems.

Keywords

digital twin, network slicing, deep reinforcement learning, 6G, mobile networks, resource allocation, QoS, sustainability, robustness.

1. Introduction

The evolution of mobile communication systems toward 5G-Advanced and sixth-generation architectures has introduced an unprecedented degree of heterogeneity in service requirements, encompassing enhanced mobile broadband, ultra-reliable low-latency communications, and massive machine-type communications. A foundational enabler for supporting such diverse traffic profiles on a shared physical infrastructure is network slicing, which permits the creation of multiple logical, isolated network instances tailored to specific service-level agreements and functional demands. Early standardization efforts and numerous survey contributions have thoroughly characterized the potential of network slicing to revolutionize resource utilization and tenant isolation [1]. The third Generation Partnership Project has progressively defined the system architecture incorporating end-to-end slicing within the 5G core and radio access network, specifying management and orchestration procedures that emphasize life-cycle automation [2]. Despite these advances, the dynamic and stochastic nature of real-world traffic, user mobility, and interference patterns renders static or threshold-based slice configuration insufficient for meeting ultra-low tail latencies and high reliability targets. This challenge motivates the integration of predictive, learning-based control mechanisms that can continuously adapt slice resource allocations while respecting the contractual boundaries of multiple tenants.

In parallel, the concept of the digital twin has emerged as a potent analytical enabler that creates a synchronized virtual representation of a physical network, capable of simulating, predicting, and optimizing its behavior in real time [3]. The digital twin paradigm, originally conceived in product lifecycle management, has been extended to complex networked systems, where it offers a sandboxed environment for testing reconfiguration actions before enacting them on the live infrastructure, thereby mitigating the risk of service degradation [4]. The confluence of network slicing and digital twin technology provides a natural architectural substrate for closed-loop automation: the twin maintains a continuously updated state of the network, runs what-if scenarios for candidate slice resource adjustments, and provides reward signals to a learning agent that settles on near-optimal policies. Recent literature has seen a surge of interest in applying deep reinforcement learning to network management tasks, driven by its capacity to handle high-dimensional state spaces and delayed rewards without requiring an explicit system model [5]. The combination of these three elements—slicing, digital twins, and deep reinforcement learning—enables a new class of self-optimizing networks that can proactively anticipate demand shifts and reconfigure slices accordingly.

This paper undertakes a system-level investigation of digital twin-based network slice optimization using deep reinforcement learning, moving beyond algorithm-centric contributions to examine the structural, operational, and governance challenges that arise when deploying such a system at scale. While existing works frequently emphasize proof-of-concept simulations with simplified channel models and static user distributions, the translation of these concepts into production-grade mobile networks demands careful consideration of twin fidelity, synchronization overhead, policy safety, multi-stakeholder governance, and overall sustainability. We therefore structure the discussion around a reference architecture, analyzing the interplay between twin engines, learning agents, and physical orchestration layers, and we highlight the architectural trade-offs that determine overall system viability. Subsequent sections address the robustness of the optimization loop under model drift and adversarial conditions, fairness in multi-slice resource allocation, and the energy and carbon implications of maintaining a high-fidelity twin and training deep reinforcement learning models continuously. Throughout the paper, we draw on insights from real-world mobile network operation, distributed systems design, and technology policy, aiming to provide a holistic perspective that can guide researchers, network operators, and regulators alike.

2. Digital Twin-Enabled Slice Optimization Architecture

A realistic deployment of digital twin-based slice optimization must reconcile the high update rate of radio conditions, the computational cost of twin synchronization, and the latency requirements of control decisions. Our reference architecture separates concerns into three interconnected planes: the physical infrastructure plane encompassing radio access points, transport network elements, and core functions; the digital twin mirror plane that ingests telemetry and maintains a time-sensitive simulation model; and the intelligence plane housing the deep reinforcement learning agent along with policy enforcement points. The twin is not a monolithic replica but a composition of sub-models capturing radio propagation, user equipment mobility, traffic generation, compute node load, and energy consumption, each updated at different frequencies determined by their dynamics. This composite modeling approach reduces computational burden while preserving sufficient fidelity for slice-level decision making. The twin periodically generates a state snapshot that is consumed by the learning agent, which outputs slice resource configuration vectors specifying radio resource block allocations, compute resource shares, and forwarding graph adjustments. The proposed actions are first validated in the twin's sandbox against safety constraints—such as minimum throughput guarantees and maximum latency bounds—before being committed to the physical orchestration layer via northbound interfaces of the software-defined networking controller and the management and orchestration framework.

A central challenge in this architecture is managing the staleness of the twin's state and the consequent impact on decision quality. Radio channels can decorrelate on the order of milliseconds, while end-to-end telemetry collection, aggregate processing, and twin synchronization may incur tens to hundreds of milliseconds of delay depending on the geographic distribution of the data centers hosting the twin. This temporal offset can cause the reinforcement learning agent to act on an outdated view, potentially leading to oscillatory resource reallocations or even momentary service level agreement violations. Mitigating this requires architectural choices such as edge-resident twin instances that proxy the global twin for time-critical radio resource management decisions, with a hierarchical model where local twins handle intra-cell scheduling while a centralized twin resolves inter-slice conflicts over transport and core resources. Such a layered architecture introduces a governance question about the delegation of authority: local twins must operate within bounds set by the global policy, yet the global policy must remain adaptable based on aggregated local experience. The interaction between these levels can be framed as a multi-agent reinforcement learning problem with coordination constraints, although a single global agent with twin-synthesized state aggregation offers simpler operational management at the cost of higher latency.

The digital twin must also interface with the network data analytics function standardized in 5G core networks to ingest long-term statistics and with the service orchestration layer to receive slice intent descriptors that express performance objectives in declarative forms. This integration allows the twin to not only mirror the current network state but also forecast future demand patterns by extrapolating from historical trends and scheduled events, such as large public gatherings or planned maintenance. The deep reinforcement learning agent can then adopt a model-based planning approach in which the twin acts as a learned environment model, enabling the agent to compute value estimates over simulated trajectories before interacting with the real network. The combination of model-free policy improvement with model-based lookahead presents a fertile design space for balancing sample efficiency and computational overhead. Architecting this loop demands careful attention to data provenance, versioning of twin models, and reproducibility of decision logs, which directly bear on operational accountability and regulatory compliance.

3. Deep Reinforcement Learning Policy Design and Integration

The resource allocation problem across multiple slices with heterogeneous quality-of-service targets can be cast as a sequential decision process under uncertainty, where the agent observes the composite state of the network's slices—including buffer occupancies, channel quality indicators, and active user counts—and selects per-slice resource budgets. Deep reinforcement learning is well-suited to this domain because the value function can capture long-term trade-offs between immediate throughput gains and future capacity headroom. A family of policy gradient methods, among which proximal policy optimization has emerged as a robust and sample-efficient algorithm for continuous control tasks, provides a reliable foundation for this learning task [8]. Proximal policy optimization constrains policy updates to a trust region, reducing the risk of catastrophic performance collapse during online training, which is critical when operating on production infrastructure where failed exploration can degrade live services. The algorithm's clipped surrogate objective ensures monotonic improvement properties without the heavy computational burden of second-order methods.

Integration of the learning agent with the digital twin amplifies these advantages because the twin can generate a diverse range of counterfactual traffic and failure scenarios, enabling the agent to learn recovery behaviors without exposing real users to degraded performance. During offline pre-training and ongoing refinement, the agent interacts with the twin replica, receiving reward signals that combine slice-specific utility functions—such as a logarithmic utility for elastic traffic to ensure proportional fairness, step-function penalties for latency-bound slice violations, and cost terms for energy consumption. The reward design critically shapes emergent behaviors, and must encapsulate both per-tenant contractual obligations and operator-level objectives like total spectral efficiency and energy proportionality. A recent study demonstrated that proximal policy optimization can effectively guarantee quality-of-service in 5G network slicing by adapting resource allocation under dynamic load conditions, underscoring the algorithm's aptitude for this domain [11]. However, transferring policies trained purely in the twin environment to the physical network remains nontrivial due to the inevitable gap between simulated and real dynamics, referred to as the reality gap. Addressing this gap requires domain randomization during twin-based training, periodic online fine-tuning using real network feedback, and policy validation guards that halt unsafe actions before execution.

The system design must also accommodate the operational reality that the agent is not a standalone module but part of a larger network management ecosystem with legacy equipment, proprietary scheduler implementations, and multi-vendor configurations. A practical deployment pathway embeds the deep reinforcement learning policy as a recommendation engine that proposes slice configurations to the operations support system, with a deterministic rule-based fallback mechanism that takes over if the proposed configuration is rejected by safety monitors. This human-in-the-loop or system-in-the-loop paradigm reduces deployment risk and aligns with telecom operational practices that mandate gradual transition to fully autonomous control. Over time, as confidence in the policy grows, the level of automation can be increased from open-loop recommendation to closed-loop enforcement, a staged approach that mirrors the autonomous network levels defined by industry bodies. The digital twin serves as the validation crucible at each stage, recording all decisions and their outcomes to support root cause analysis and continuous policy refinement.

4. System Deployment, Infrastructure, and Scalability

Deploying a digital twin-based slice optimization framework across a nationwide mobile network involves significant infrastructure planning at the intersection of radio access, edge data centers, and centralized cloud facilities. The twin's model granularity must be chosen to match the available computing budget: a per-cell high-fidelity ray-tracing propagation model may be infeasible to maintain in real time for tens of thousands of cells, whereas a coarser statistical channel model might suffice for inter-cell slice budget decisions. A pragmatic approach adopts a tiered twin fidelity, where the inner loop for sub-second scheduling uses lightweight neural surrogates trained offline on high-fidelity simulation outputs, while the outer loop for minute-scale inter-slice coordination runs a more detailed system-level simulator on a cloud-based twin. This design exploits the natural timescale separation present in mobile networks, where slice resource budgets change slower than per-user scheduling decisions. The resulting system bears resemblance to hierarchical reinforcement learning architectures, albeit with the twin serving as the surrogate environment at both levels.

Scalability concerns also extend to the data pipeline that continuously feeds radio network information, core network counters, and energy measurements into the twin. The ingestion system must handle millions of measurement reports per second, necessitating stream-processing frameworks and intelligent sampling strategies that preserve statistical distributions while reducing volume. Furthermore, the twin's database must support time-travel queries and versioning to allow training of the reinforcement learning agent on consistent snapshots. These requirements push the digital twin toward a design grounded in event sourcing and command query responsibility segregation patterns, where the state of the twin is derived from an immutable log of events. In addition to simplifying auditing, this design enables replay-based training and debugging, which are invaluable for diagnosing unexpected policy behaviors before they affect live traffic. The orchestration layer mediating between the twin, the learning agent, and the physical network must implement exactly-once semantics for action execution to prevent duplicate resource grants that could arise from retransmissions in a failure-prone control plane.

The choice between a centralized agent and a distributed multi-agent learning setup raises further infrastructure trade-offs. A centralized agent has a global view facilitated by the twin but introduces a single point of failure and may suffer from the curse of dimensionality as the network size grows. Conversely, fully decentralized agents at each edge site can make fast local decisions but face the challenge of coordinating to avoid inter-slice interference, a problem often addressed through federated learning or consensus mechanisms. An intermediate federated digital twin architecture, where local twins periodically synchronize model parameters with a global twin while maintaining local control loops, appears promising. In this model, the reinforcement learning policy can be trained in a distributed manner using federated policy gradients, with the twin serving as the local environment simulator. This arrangement inherently addresses data sovereignty and privacy concerns because raw telemetry data need not leave the operator's regional data center, aligning with emerging data localization regulations and reducing backhaul load.

5. Governance, Policy, and Sustainability

Introducing autonomous slice optimization agents guided by digital twins raises profound governance challenges that extend beyond technical performance to regulatory, economic, and ethical dimensions. Network slicing inherently involves multiple tenants—virtual operators, vertical industries, emergency services—each with distinct rights and expectations. The optimization agent must not only maximize aggregate efficiency but also enforce inter-tenant isolation and contractual fairness. From a policy standpoint, the digital twin provides an unprecedented opportunity for transparent governance: all slice allocation decisions can be traced back to twin states, reward function components, and policy network outputs, creating an auditable record that regulators could inspect to detect anti-competitive behavior or violations of net neutrality principles [18]. Embedding such transparency requires that the twin's model and the agent's policy be interpretable to domain experts, which may motivate the use of attention-based neural architectures or post-hoc explanation methods that highlight which features drove a particular resource grant. The governance framework should also define liability boundaries when an autonomous agent makes a decision that causes a service outage; the operator, the software vendor, and the tenant need clear contractual instruments that delineate responsibility.

Sustainability considerations are increasingly critical as mobile networks account for a non-trivial fraction of global electricity consumption, projected to grow with densification and higher bandwidth demands. A digital twin-based optimization loop can explicitly incorporate energy metrics into the reward function, allowing the agent to learn policies that dynamically deactivate carriers or steer traffic to more energy-efficient bands during low-load periods. However, operating the digital twin itself consumes substantial computational resources, particularly if high-fidelity simulations run continuously. A life-cycle carbon analysis must balance the energy savings achieved in the physical network against the added footprint of twin computation and model training. Early assessments of information and communication technology electricity use indicate that moving toward carbon-aware computing, where training and twin updates are scheduled during periods of high renewable energy availability, can significantly reduce net emissions [17]. Additionally, the design of the agent's neural network architecture and training frequency should be co-optimized with the twin's fidelity to avoid a situation where marginal gains in slicing efficiency are outweighed by the carbon cost of the optimization infrastructure itself. Operators adopting such systems should publish sustainability metrics alongside traditional key performance indicators to maintain public accountability.

The social dimension of fairness cannot be reduced solely to utilitarian metrics. In crisis scenarios, the slice optimization agent must be capable of overriding normal economic objectives to prioritize emergency services and public safety communications. Encoding such prioritization requires a hierarchical reward architecture where the primary reward tier enforces non-negotiable safety constraints, a secondary tier maintains fairness among commercial tenants, and a tertiary tier pursues efficiency. Policy-makers and standards bodies must collaborate to define these tiers in a technology-neutral manner that remains consistent across vendors. The digital twin can serve as a virtual testbed to evaluate the societal impact of alternative prioritization policies under extreme load scenarios, making the trade-offs explicit before they are needed in a real emergency.

6. Robustness and Fairness in Slice Optimization

Robustness of the optimization system encompasses resilience to twin model inaccuracies, adversarial inputs, and non-stationary environment dynamics. The digital twin, no matter how elaborate, is an approximation of the physical network, and systematic biases in channel modeling or traffic prediction can cause the reinforcement learning agent to converge to policies that perform poorly when deployed. A robust architecture must quantify the uncertainty of twin predictions and propagate it to the agent's decision-making process, for instance by using distributional reinforcement learning that models the entire return distribution rather than a point estimate. This allows the agent to distinguish between states where the twin is confident and those where it is not, and to adopt more conservative actions in high-uncertainty regimes. Additionally, the system must guard against adversarial perturbations, such as an attacker injecting carefully crafted measurement reports to mislead the twin, which in turn could cause the agent to starve a slice or cause congestion. Defensive strategies include anomaly detection on telemetry streams, robust aggregation of measurements from redundant sensors, and adversarial training of the agent in the twin environment with worst-case perturbations [13].

Fairness among slices is a multi-faceted requirement that involves both procedural fairness—the process by which resource decisions are made—and distributive fairness—the outcomes in terms of service quality experienced by tenants. Simple proportional fairness formulations used in many resource allocation studies may fail to protect small tenants that require a baseline amount of resources even when their demand is low relative to larger tenants. The optimization framework must therefore support minimum resource guarantees and maximum bounds, effectively a form of hard slicing within soft slicing management. The deep reinforcement learning agent can handle such constraints via reward shaping that imposes large penalties for guarantee violations, or through constrained policy optimization techniques that strictly enforce expected costs. The digital twin aids in fairness analysis by providing a counterfactual simulation environment: one can replay a history of traffic with modified fairness parameters and observe how different tenant satisfaction metrics evolve without impacting the live network. This capability transforms fairness from a static configuration into a continuously monitored and tuned system property, enabling operators to offer differentiated slice products with verifiable service level agreement compliance.

A related concern is the temporal consistency of slice allocations under the digital twin and reinforcement learning pair. Frequent reconfiguration, while potentially optimal in the short term, can degrade tenant experience if applications cannot adapt to rapidly fluctuating resources. The agent's policy can be regularized to penalize large reallocation magnitudes, introducing hysteresis that stabilizes slice boundaries. The digital twin allows offline tuning of this hysteresis coefficient by simulating the trade-off between stability and optimality, a task that would be highly disruptive to perform on a live network. End-users ultimately care about application-level quality of experience, which depends on the entire chain of service functions. The twin can incorporate models of video streaming adaptation logic, industrial control loop dynamics, and augmented reality rendering pipelines to translate raw throughput and latency metrics into predicted user satisfaction scores, aligning the agent's reward signal with end-user utility rather than lower-level network metrics alone.

7. Conclusion

This paper has presented a system-oriented exploration of digital twin-based network slice optimization using deep reinforcement learning, emphasizing architectural design, deployment practicalities, governance, fairness, and sustainability. The closed-loop paradigm, where a continuously synchronized digital twin enables safe and foresighted policy learning, represents a significant departure from conventional reactive network management. We have argued that the transition from algorithm simulation to operational infrastructure must surmount challenges of twin fidelity, latency, scalability, and multi-stakeholder accountability. By structuring the twin as a hierarchical, composable entity and integrating provenance-aware data pipelines, operators can balance the demands of real-time control with the rigor required for auditing and regulatory trust. The synergy between proximal policy optimization and the twin's predictive capabilities offers a path toward autonomous slice management that respects stringent service-level agreements and energy budgets. At the same time, the wider societal implications demand that fairness, emergency prioritization, and carbon accounting be designed into the system from the outset, not retrofitted. Looking forward, the convergence of digital twin standards, open radio access network interfaces, and federated learning infrastructures will lower the barrier for multi-operator twin federation, enabling cooperative slice optimization across administrative domains. Achieving this vision will require continued interdisciplinary collaboration among network engineers, machine learning researchers, policy-makers, and sustainability experts to ensure that the next-generation mobile infrastructure is not only high-performing but also trustworthy, equitable, and environmentally responsible.

References

1. Foukas, X., Patounas, G., Elmokashfi, A., & Marina, M. K. (2017). Network slicing in 5G: Survey and challenges. IEEE Communications Magazine, 55(5), 94–100.
2. 3GPP. (2019). System architecture for the 5G System (5GS); Stage 2 (Release 16). 3GPP TS 23.501.
3. Bariah, L., Muhaidat, S., Al-Dweik, A., & Coté, D. (2021). Digital twin for 6G: The network digital twin. IEEE Network, 35(5), 92–99.
4. Grieves, M., & Vickers, J. (2017). Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In Transdisciplinary perspectives on complex systems (pp. 85–113). Springer.
5. Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., Liang, Y. C., & Kim, D. I. (2019). Applications of deep reinforcement learning in communications and networking: A survey. IEEE Communications Surveys & Tutorials, 21(4), 3133–3174.
6. Sciancalepore, V., Samdanis, K., Costa-Perez, X., Capone, A., & Banchs, A. (2017). Mobile traffic forecasting for maximizing 5G network slicing resource utilization. In IEEE INFOCOM 2017.
7. Wei, Y., Peng, M., & Wang, C. (2021). Digital twin-enabled network slicing: A deep reinforcement learning approach. IEEE Transactions on Communications, 69(12), 8417–8433.
8. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.
9. D'Andreagiovanni, F., Mannaro, M., & Napolitano, F. (2018). 5G-EmPOWER: A software-defined networking platform for 5G radio access networks. In 2018 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN).

10. Caballero, P., Banchs, A., de la Oliva, A., & Giupponi, L. (2020). Network slicing games: Enabling customization in multi-tenant networks. IEEE/ACM Transactions on Networking, 28(2), 662–675.

11. Li, Q. (2026). QoS Assurance Mechanism for 5G Network Slicing Based on the Deep Reinforcement Learning PPO Algorithm. arXiv preprint arXiv:2605.03345.

12. Liu, Y., Peng, M., & Wang, C. (2021). Digital twin for 6G: A survey. China Communications, 18(9), 1–18.

13. Sagduyu, Y. E., Shi, Y., & Erpek, T. (2021). Adversarial deep learning for wireless communications. IEEE Communications Magazine, 59(9), 62–67.

14. Olwal, T. O., Djouani, K., & Kurien, A. M. (2020). Energy-efficient resource allocation in 5G network slicing: A deep reinforcement learning approach. IEEE Access, 8, 83695–83710.

15. Taleb, T., Frangoudis, P. A., Benkacem, I., & Ksentini, A. (2019). On supporting different QoS classes in 5G network slicing: A deep reinforcement learning approach. IEEE Transactions on Network and Service Management, 16(4), 1560–1575.

16. 3GPP. (2021). Management and orchestration; Provisioning. 3GPP TS 28.531.

17. Andrae, A. S. G. (2020). New perspectives on internet electricity consumption in 2030. Engineering Reports, 2(12), e12256.

18. Yrjölä, S., Ahokangas, P., & Matinmikko-Blue, M. (2020). Sustainability as a value driver in the 5G ecosystem: The regulatory framework perspective. Telecommunications Policy, 44(7), 101975.

19. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. ACM Computing Surveys, 54(6), 1–35.

20. Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital twin: Enabling technologies, challenges and open research. IEEE Access, 8, 108952–108971.