Hierarchical Reinforcement Learning Framework for End-to-End Network Slice Lifecycle Management

Dannis Rease
School of Computing, Clemson University, Clemson, SC, USA.
rosedennis@clemson.edu

Ranald Jeacebs
Department of Computer Science, Colorado State University, Fort Collins, CO, USA.
ronald.work@colostate.edu

Abstract

Network slicing has emerged as a cornerstone of 5G and beyond mobile systems, enabling the provisioning of customized logical networks over a shared physical infrastructure. The end-to-end lifecycle management of network slices—spanning preparation, commissioning, operation, and decommissioning—poses a complex sequential decision-making problem under uncertainty. Traditional orchestration approaches rely on heuristics and reactive control loops that struggle to meet dynamic quality-of-service requirements and resource efficiency goals across multiple administrative domains. This paper presents a hierarchical reinforcement learning framework for autonomous end-to-end slice lifecycle management. The proposed architecture decomposes the management plane into multiple temporal and functional abstraction levels, each handled by a reinforcement learning module operating on distinct timescales. A high-level meta-controller selects long-term slice objectives and policy options, intermediate controllers manage lifecycle state transitions and cross-domain coordination, while low-level controllers execute fine-grained resource allocation and fault mitigation actions. The framework leverages the options formalism and policy gradient methods to enable sample-efficient learning and scalable decision-making. We discuss architectural trade-offs, infrastructure implications, and system-level design choices that govern robustness, sustainability, and multi-tenant fairness. The paper further examines policy and governance dimensions, including trustworthiness, auditability, and regulatory compliance, arguing that hierarchical structure naturally accommodates human-intelligible control points and policy constraints. By integrating lifecycle-wide optimization with hierarchically abstracted control, the proposed framework addresses key shortcomings of flat reinforcement learning approaches and charts a path toward fully automated, zero-touch slice management in future communication networks.

Keywords

network slicing, lifecycle management, hierarchical reinforcement learning, 5G and beyond, resource orchestration, quality of service, policy gradient, closed-loop automation.

1. Introduction

The proliferation of heterogeneous services in fifth-generation (5G) networks has driven the adoption of network slicing as a fundamental architectural paradigm [1]. By enabling multiple isolated logical networks on a common physical substrate, slicing allows tailored performance guarantees for enhanced mobile broadband, ultra-reliable low-latency communications, and massive machine-type communications [2]. The 3rd Generation Partnership Project (3GPP) defines a management architecture for network slicing that spans the entire lifecycle from design and preparation to commissioning, operation, and decommissioning [3]. Complementary standardization efforts, such as those by the European Telecommunications Standards Institute (ETSI) for network function virtualization (NFV) and the Zero-touch network and Service Management (ZSM) group, provide frameworks for orchestration and automation [4]. Despite these advances, operationalizing end-to-end slice lifecycle management remains a formidable challenge because slice instantiation, scaling, healing, and termination must be continuously coordinated across radio access, transport, and core network domains under fluctuating demand patterns and resource constraints.

Traditional management systems rely on static policies, threshold-based triggers, and human-in-the-loop interventions, none of which can match the agility required in dense, multi-tenant deployments. Reinforcement learning (RL) has attracted considerable attention as a means to achieve closed-loop automation in network slicing. Deep reinforcement learning (DRL), in particular, has been applied to slice resource allocation and traffic engineering with promising results [5], [6]. Proximal policy optimization (PPO) has become a popular policy gradient algorithm due to its stability and sample efficiency in continuous control tasks [7]. However, flat RL controllers face inherent limitations in environments characterized by long temporal horizons, composite actions, and the necessity of multi-timescale reasoning. Hierarchical reinforcement learning (HRL) addresses these limitations by introducing temporal abstraction through sub-policies and higher-level decision modules [8]. Later advances such as the options framework and feudal networks have demonstrated improved learning efficiency and structured exploration in complex domains [9], [10]. Preliminary explorations of HRL in networking domains, including virtual network function placement, suggest benefits in scalability and policy reusability [11].

While single-level DRL approaches have been shown effective for slice-level quality-of-service (QoS) assurance, as demonstrated in recent work employing PPO for 5G slice QoS management [12], such designs cannot easily extend to the holistic lifecycle perspective that involves sequential phase transitions, cross-domain interactions, and long-term policy adaptation. This paper proposes a unified hierarchical reinforcement learning framework that decomposes the end-to-end slice management problem into multiple levels of abstraction, each learning to optimize objectives at its own timescale. The framework embeds lifecycle state logic, cross-domain coordination, resource orchestration, and fault management within a coherent multi-level control hierarchy. The remainder of the paper presents the system architecture, analyzes structural trade-offs, and articulates the broader implications for infrastructure design, sustainability, fairness, and governance.

2. Background and Related Work

Network slicing is realized through the coordinated instantiation of virtualized network functions, software-defined connectivity, and dedicated resource partitions across a shared infrastructure. The 3GPP management reference model defines four lifecycle phases: preparation, commissioning, operation, and decommissioning, each involving distinct information elements, performance metrics, and state transitions [3]. In parallel, ETSI NFV Management and Orchestration (MANO) and the ZSM framework introduce hierarchical management domains and cross-domain integration enablers, yet they stop short of prescribing fully autonomous decision-making algorithms [4]. Emerging architectures such as the O-RAN Alliance specifications further disaggregate control by introducing near-real-time and non-real-time RAN intelligent controllers that can host machine learning applications

[13]. These developments provide fertile ground for embedding learned control policies, but the complexity of coordinating decisions across different timescales and administrative domains calls for more structured learning approaches.

Reinforcement learning has been widely studied for individual slice management tasks. DRL-based dynamic resource allocation, admission control, and traffic steering solutions have demonstrated significant performance gains in simulation environments [5], [6]. PPO-based controllers offer improved stability and monotonic policy improvement [7], and have been specifically tailored for slice QoS assurance [12]. Nonetheless, these solutions treat the management problem as a flat Markov decision process, wherein a single policy network must directly map high-dimensional state observations to fine-grained configuration commands. Such flat formulations suffer from the curse of dimensionality, sparse reward signals over long horizons, and difficulty in reusing learned behaviors across different slice types and lifecycle stages.

Hierarchical reinforcement learning introduces temporal abstraction, allowing agents to reason at multiple spatial and temporal granularities. The options framework formalizes the notion of temporally extended actions, or options, each defined by an initiation set, a sub-policy, and a termination condition, and enables a high-level policy to select among options while low-level policies execute primitive actions [8]. Deep hierarchical models, such as the hierarchical-DQN that learns sub-goals through intrinsic motivation, and feudal networks that use manager-worker architectures with latent goal representations, have shown superior performance on long-horizon tasks [9], [10]. In the networking domain, HRL has been applied to service function chain embedding and VNF resource allocation, demonstrating that hierarchical decomposition improves convergence speed and policy generalizability [11]. These successes motivate the extension of HRL principles to the holistic slice lifecycle management problem, where decisions range from hours-long slice planning to millisecond-scale packet scheduling.

3. Hierarchical Reinforcement Learning Architecture for Slice Lifecycle

The proposed framework structures the slice management control plane into three hierarchically nested reinforcement learning agents: a high-level meta-controller, one or more mid-level lifecycle coordinators, and distributed low-level resource controllers. The high-level meta-controller operates on business and operational timescales, receiving aggregate state information about infrastructure capacity, service level agreement (SLA) portfolios, and market context, and selecting among a set of pre-trained option policies that represent slice templates, scaling strategies, and inter-slice isolation policies. Each option chosen by the meta-controller remains active for extended intervals measured in minutes to hours, during which a corresponding mid-level lifecycle coordinator assumes control over the evolution of a particular slice instance or a group of related slices.

The mid-level coordinator is responsible for managing lifecycle state transitions: from preparation to commissioning, through active operation with elasticity and healing, and ultimately to decommissioning. Its observations include slice-specific key performance indicators, resource utilization ratios, traffic forecasts, and alarm events. Its actions correspond to invoking specific orchestration workflows, such as scaling a virtualized radio access network distributed unit, migrating a user plane function instance across edge clouds, or triggering a recovery procedure after a service degradation. The coordinator's policy is itself structured as a finite state machine augmented with option sub-policies that handle each lifecycle phase, enabling the agent to learn phase-appropriate behavior while reusing common sub-policies across slices with similar performance profiles.

Low-level controllers execute fine-grained, real-time configuration actions at sub-second timescales. These may include adjusting radio resource block allocations, tuning queue management parameters, or steering traffic flows through different network paths. Each low-level controller operates within a well-defined scope—such as a single network domain or a single VNF instance—and receives dense reward signals computed from instantaneous throughput, latency, and packet loss metrics. The hierarchical decomposition empowers the system to cope with the combinatorial explosion of the global action space; the meta-controller's decision is concerned with a compact set of high-impact policy options, while the low-level controllers solve focused, domain-specific optimization subproblems.

All agents in the hierarchy employ policy gradient methods, with PPO serving as a common learning backbone due to its trust region properties and compatibility with continuous and discrete action spaces [7], [12]. The meta-controller's option-level policy is trained using cumulative rewards that aggregate the performance of low-level controllers over the lifetime of each option, thus providing a coherent credit assignment mechanism across temporal scales. The framework makes extensive use of the options formalism: each high-level action selects an option that persists until a learned termination condition is met, at which point the meta-controller may switch to a different option. This design naturally accommodates irregular and event-driven lifecycle events, such as sudden traffic surges or infrastructure failures, while preserving the learning efficiency gained through temporal abstraction.

State representations are carefully engineered at each level to avoid information overload. The meta-controller ingests processed forecasts, topology digests, and aggregated SLA compliance statistics, while low-level controllers observe raw or lightly processed telemetry streams from their local domains. The hierarchical structure thus imposes a deliberate information bottleneck that filters out irrelevant high-frequency noise at the meta-controller level, improving learning stability and generalization. Reward shaping across levels aligns local objectives with global system goals: low-level controllers are encouraged to maximize domain-level QoS satisfaction, mid-level coordinators receive rewards based on slice-wide SLA compliance and resource cost, and the meta-controller is rewarded for overall operator utility, which may combine revenue, energy efficiency, and regulatory compliance metrics.

4. End-to-End Lifecycle Management Integration

Integrating the hierarchical RL framework into the full slice lifecycle requires mapping the abstract control levels onto the 3GPP and ETSI ZSM management planes. During the preparation phase, the meta-controller evaluates candidate slice designs by simulating the expected performance of candidate option policies under demand forecasts. It approves or rejects slice creation requests based on predicted resource feasibility and alignment with long-term operator objectives. Once a slice is accepted, the mid-level coordinator initiates commissioning by invoking NFV orchestrator workflows to instantiate network services, configure connectivity, and allocate initial resource quotas.

In the operational phase, the mid-level coordinator continuously monitors slice health and triggers scaling, migration, or reconfiguration actions through option-level decisions. These decisions are implemented by low-level controllers that execute fine-grained resource adjustments. The hierarchical design enables coordinated scaling actions across multiple domains—for example, simultaneously expanding radio access capacity and backhaul bandwidth while migrating a core network function to maintain end-to-end latency guarantees. The termination condition of an operational option may be triggered by a SLA violation threshold, a scheduled maintenance window, or a change in user demand patterns, at which point the meta-controller re-evaluates the active policy.

Cross-domain coordination is a particularly challenging aspect of lifecycle management, because radio, transport, and core network segments are often administered by separate organizational units or even different operators. The framework accommodates such federated deployments by allowing each domain to run its own low-level controller and potentially its own mid-level coordinator, while a centralized or logically centralized meta-controller orchestrates inter-domain option selection. Information exchange between domains is limited to aggregated KPIs and option-level coordination signals, reducing the communication overhead and preserving domain autonomy. The meta-controller's options may encapsulate inter-domain service level specification templates that constrain the behavior of domain-local agents, thereby enforcing end-to-end consistency without centralized micro-management.

Decommissioning is treated as a deliberate lifecycle phase rather than a simple teardown. The mid-level coordinator learns to plan decommissioning sequences that carefully drain traffic, terminate network functions, and release resources in a manner that avoids degrading the QoS of remaining slices. The meta-controller can reuse learned decommissioning policies across different slice types, further reducing the learning burden. The entire lifecycle thus forms a closed loop in which the outcomes of past slice instances inform the preparation decisions for future demands, enabling continuous improvement of the operator's slice portfolio over time.

5. Architectural Trade-offs and Infrastructure Implications

The introduction of a hierarchical RL framework into production network slicing infrastructures entails a series of architectural trade-offs. Centralizing the meta-controller simplifies global optimization and inter-slice coordination but creates a potential scalability bottleneck and a single point of failure. A fully distributed hierarchy, in which multiple meta-controller instances coordinate through consensus protocols, improves resilience and geographical scaling at the cost of increased coordination complexity and delayed convergence. An intermediate approach uses a logically centralized meta-controller with physically replicated inference nodes synchronized via eventually consistent state stores, balancing consistency and availability. The choice among these deployment models must account for the operator's risk tolerance, jurisdictional data sovereignty constraints, and the latency characteristics of the wide-area network connecting management nodes.

Sustainability considerations are increasingly critical in network operations. The proposed hierarchy can integrate carbon-aware and energy-efficiency objectives directly into the reward functions of high-level and mid-level agents. The meta-controller can learn to steer traffic toward slices that leverage energy-proportional hardware or renewable-powered edge sites, while low-level controllers apply micro-sleep strategies in radio units during idle slots. However, optimizing for energy efficiency may conflict with strict latency guarantees, necessitating careful reward calibration and constraint enforcement through safe reinforcement learning techniques [15]. The hierarchical structure permits the expression of hard safety constraints at the meta-controller level, which can disable option choices that would lead to sustained SLA violations or excessive power consumption, thus providing a guardrail for lower-level exploration.

Robustness against adversarial conditions and rare events is another design imperative. HRL agents must be trained with domain randomization, adversarial scenario generation, and careful curriculum learning to withstand tail-latency events, flash crowds, and partial infrastructure failures. The hierarchical decomposition naturally isolates failure domains: a low-level controller failure remains contained within its limited scope, and the mid-level coordinator can trigger mitigation options independently. The meta-controller's option selection can incorporate uncertainty estimates and risk-sensitivity measures, enabling conservative behavior under high uncertainty. These mechanisms collectively improve the trustworthiness of autonomous management operations.

Infrastructure implications extend to the compute and networking resources required to host the control plane. Low-level controllers demand low-latency inference, potentially deployed on field-programmable gate arrays or smart network interface cards co-located with data plane nodes. Mid-level coordinators run on edge cloud platforms, while the meta-controller occupies a centralized or regional cloud. The communication fabric linking these components must support reliable, low-jitter message passing for state synchronization and option handshaking. Service mesh architectures and cloud-native design patterns, such as sidecar proxies and control-plane of control-planes, provide a natural substrate for deploying such multi-tier reinforcement learning systems. Operators must also plan for the lifecycle of the ML models themselves, including continuous training pipelines, model versioning, and canary deployments, all of which represent an operational overhead that must be justified by the resulting improvements in network autonomy and efficiency.

6. Policy, Fairness, and Governance Considerations

Network slicing operates at the intersection of technical orchestration and regulatory policy, making fairness among tenants a paramount concern. Without explicit fairness constraints, a pure utility-maximizing RL agent could allocate resources in ways that systematically disadvantage smaller tenants or delay decommissioning of low-revenue slices to the detriment of new entrants. The hierarchical RL framework can encode fairness objectives at the meta-controller level, such as max-min fairness or proportional fairness over slices, and translate these into option selection criteria that respect inter-tenant equity even when low-level controllers are optimized for domain-specific efficiency [17]. The mid-level coordinator can enforce per-slice resource budgets derived from the meta-controller's allocation decisions, and the option termination conditions may incorporate fairness-driven thresholds that prevent resource hoarding.

Governance of autonomous network management systems raises questions of accountability, transparency, and human oversight. The hierarchical architecture inherently offers a degree of interpretability because each level makes decisions with distinct scopes and on timescales that allow human operators to inspect, audit, and override actions before they propagate. The meta-controller's option choices and termination events form a sparse, high-level action trace that is far more amenable to post-hoc analysis than the continuous stream of low-level control decisions. Explainable reinforcement learning techniques [16] can be applied at the meta-controller level to generate natural language justifications for option selections based on forecasted traffic trends and SLA risks, thereby supporting regulatory audits and compliance with emerging AI governance frameworks.

Data privacy and inter-operator trust are additional policy dimensions. In a multi-domain slice spanning several administrative entities, the meta-controller may need access to aggregated telemetry from each domain to make coordinated decisions. Federated learning variants and secure multi-party computation can protect the confidentiality of raw telemetry while still enabling joint training of the meta-controller's policy. The options framework further supports data minimization: low-level controllers consume fine-grained data locally and emit only option-level performance summaries upward, reducing the exposure of sensitive subscriber-level measurement data. This alignment with privacy-by-design principles strengthens the framework's acceptability in jurisdictions with stringent data protection regulations.

Regulatory compliance also extends to service continuity obligations and emergency communication prioritization. The meta-controller can be explicitly programmed with statutory policy constraints that take precedence over learned options, for instance, guaranteeing minimum resources for public safety slices during disaster scenarios. Such hard constraints can be enforced by a deterministic policy overlay that filters the action set of the meta-controller, effectively creating a normative governance shell around the learned core. The hierarchical separation of concerns thereby harmonizes the flexibility of data-driven learning with the rigidity required by legal and societal mandates, a balance that flat RL architectures struggle to achieve.

7. Conclusion

The end-to-end lifecycle management of network slices in 5G, beyond 5G, and future 6G systems demands decision-making capabilities that span multiple timescales, administrative domains, and operational constraints. This paper has presented a hierarchical reinforcement learning framework that decomposes the management plane into three interacting levels: a high-level meta-controller for long-term policy selection, mid-level lifecycle coordinators for slice state transitions, and low-level controllers for fine-grained resource orchestration. By grounding the design in the options formalism and leveraging policy gradient methods such as PPO, the framework achieves sample-efficient learning, scalable real-time control, and natural integration with existing orchestration architectures. Through structural analysis, the paper has illuminated trade-offs related to centralization, sustainability, robustness, and infrastructure deployment, highlighting how hierarchical abstraction creates opportunities for energy-aware optimization and fault isolation. Moreover, the discussion of policy and governance showed how the multi-level architecture embeds fairness guarantees, audit trails, and regulatory compliance mechanisms as first-class design elements rather than retrofitted constraints. As networks evolve toward zero-touch management paradigms, hierarchical reinforcement learning stands out as a principled approach to tame complexity while preserving the accountability and adaptability that society demands from critical communication infrastructures. Future work should address rigorous safety verification of learned options, standardization of inter-level interfaces, and large-scale trials in live production environments to validate the framework's performance under real-world operational stress.

References

1. Rost, P., Mannweiler, C., Michalopoulos, D. S., Sartori, C., Sciancalepore, V., Sastry, N., Holland, O., Tayade, S., Han, B., Bega, D., & Banchs, A. (2017). Network slicing to enable scalability and flexibility in 5G mobile networks. IEEE Communications Magazine, 55(5), 72–79.

2. Foukas, X., Patounas, G., Elmokashfi, A., & Marina, M. K. (2017). Network slicing in 5G: Survey and challenges. IEEE Communications Magazine, 55(5), 94–100.

3. 3GPP. (2020). 5G; Management and orchestration; Concepts, use cases and requirements (3GPP TS 28.530 version 16.2.0 Release 16). European Telecommunications Standards Institute.

4. ETSI. (2020). Zero-touch network and service management (ZSM); Reference architecture (ETSI GS ZSM 002 V1.1.1). European Telecommunications Standards Institute.

5. Li, R., Zhao, Z., Sun, Q., Chih-Lin, I., Yang, C., Chen, X., Zhao, M., & Zhang, H. (2019). Deep reinforcement learning for network slicing with heterogeneous resource requirements and time varying traffic. IEEE Transactions on Vehicular Technology, 68(12), 11496–11505.

6. Chen, Q., & Yu, F. R. (2020). Deep reinforcement learning for network slicing: A comprehensive survey. IEEE Communications Surveys & Tutorials, 23(1), 186–218.

7. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

8. Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. Artificial Intelligence, 112(1–2), 181–211.

9. Kulkarni, T. D., Narasimhan, K., Saeedi, A., & Tenenbaum, J. (2016). Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. Advances in Neural Information Processing Systems, 29, 3675–3683.

10. Vezhnevets, A. S., Osindero, S., Schaul, T., Heess, N., Jaderberg, M., Silver, D., & Kavukcuoglu, K. (2017). FeUdal networks for hierarchical reinforcement learning. Proceedings of the 34th International Conference on Machine Learning, 70, 3540–3549.

11. Huang, S., Li, J., & Li, X. (2021). Hierarchical reinforcement learning for network function virtualization resource allocation. IEEE Internet of Things Journal, 8(5), 3812–3822.

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

13. O-RAN Alliance. (2021). O-RAN Architecture Description v4.0. O-RAN Working Group 1.

14. Moubayed, A., Refaey, A., & Shami, A. (2020). Machine learning in network slicing—A survey. IEEE Access, 8, 70344–70367.

15. Garcia, J., & Fernandez, F. (2015). A comprehensive survey on safe reinforcement learning. Journal of Machine Learning Research, 16(1), 1437–1480.

16. Puiutta, E., & Veith, E. M. S. P. (2020). Explainable reinforcement learning: A survey. In A. Holzinger, P. Kieseberg, A. M. Tjoa, & E. Weippl (Eds.), International Cross-Domain Conference for Machine Learning and Knowledge Extraction (pp. 77–95). Springer.

17. Chen, D., Jiang, T., & Liu, Y. (2020). Fair resource allocation for network slices in 5G radio access networks with deep reinforcement learning. IEEE Transactions on Vehicular Technology, 69(9), 10197–10210.

18. Zhang, L., Afolabi, I., & Taleb, T. (2020). A survey on artificial intelligence for network slicing in beyond 5G: Techniques, open issues, and opportunities. IEEE Access, 8, 211014–211043.

19. Saad, W., Bennis, M., & Chen, M. (2020). A vision of 6G wireless systems: Applications, trends, technologies, and open research problems. IEEE Network, 34(3), 134–142.

20. Lu, Z., Lei, T., & Xu, J. (2022). Hierarchical reinforcement learning for intent-driven network slicing. IEEE Transactions on Cognitive Communications and Networking, 8(1), 180–193.