Towards Energy-Efficient AI: Learning Generic Assignment Skills with Unsupervised Reinforcement Learning for Network Slicing in Distributed 6G Networks

Navideh Ghafouri , John S. Vardakas, Kostas Ramantas, and Christos Verikoukis

***Abstract*—Since the 6th Generation (6G) of wireless networks is expected to deliver advanced services and address the evolving demands of future applications, making it a highly complex and dynamic system. Managing this complexity requires innovative Artificial Intelligence-based approaches that balance scalability, efficiency, and robustness. This work investigates decentralized, multi-level architectures for 6G networks and introduces an energy-efficient Reinforcement Learning (RL)-based automation framework for edge domain management and Network Slicing. The core objective is to mitigate network complexity by exploiting scalable and generalizable RL techniques. To this end, we propose a pre-training phase based on unsupervised RL to uncover effective resource allocation skills at the network edge. The proposed method is domain-agnostic, enabling applicability across diverse edge environments. By supporting various service types, it enhances scalability, decentralization, and operational efficiency. Our implementation demonstrates that the discovered skills comprehensively span the resource allocation space for each service category, thereby simplifying the NS in the network domains. Finally, the proposed approach is compared to other RL-based techniques and is shown to improve energy efficiency by using less computational resources.**

I. Introduction

Network Slicing (NS) has become a fundamental yet challenging task in network management and orchestration, largely due to the growing number of layers in current and emerging wireless architectures. As service provisioning and NS expand in the 6th Generation (6G), the heightened dynamism and robustness of 6G will further intensify the complexity of management and orchestration. A promising way to address these challenges is through modular and multi-level system models. However, while decentralization, scalability, and modularity are critical for 6G, they also add to the intricacy and demands of NS.

Additionally, the real-time operation of 6G entities and their orchestration introduce real-world challenges that necessitate the integration of Artificial Intelligence (AI) into future wireless networks. These challenges arise from the need to manage highly dynamic environments, where factors such as user mobility, traffic fluctuations, and network topology can change rapidly. Coordinating the behavior of distributed components across the Radio Access Network (RAN), core, and edge computing infrastructure requires fast, intelligent decision-making to meet the ultra-low latency, high reliability, and energy efficiency targets of 6G. Traditional rule-based systems struggle to cope with this level of complexity and adaptability. In contrast, AI techniques offer the ability to learn from data, optimize resource allocation, predict network conditions, and autonomously manage services. By enabling proactive and adaptive control, AI becomes essential for achieving the scalability, flexibility, and real-time responsiveness required in next-generation networks.

Reinforcement Learning (RL) has emerged as a promising strategy for network management and orchestration, with various decentralized approaches, including Multi-Agent RL (MARL) and Hierarchical RL (HRL). In MARL, multiple agents either collaborate or compete to achieve a goal, whereas in HRL, a single agent decomposes the task into a hierarchy of subtasks to solve complex problems more efficiently.

Recently, AI-driven techniques, supported by network softwarization, have been widely explored for network orchestration, management, and NS [1], [2], [6], [7]. However, sophisticated network orchestration models often face practical limitations due to their complexity. As 6G becomes more tangible, scalability and efficiency must be prioritized when developing new solutions. Furthermore, the environmental impact of networks is a critical consideration, making it essential to adopt ML-based management approaches that minimize computational overhead and improve Energy Efficiency (EE).

In this study, we aim to enable a scalable and efficient management and orchestration technique for 6G networks. We propose a novel framework to reduce computational resources and enhance efficiency in automated edge network domains. The framework includes an offline pre-training phase in which an unsupervised RL agent discovers optimal assignment skills for domain controllers. This agent identifies effective sequences of resource-allocation skills for various service types. In the next phase, the domain controller leverages these pre-trained skills to manage NS within edge domains.

The key motivations for employing the proposed unsupervised RL-based pre-training phase for 6G orchestration include:

- *Supporting Modular Network Architecture*: The 6G system model consists of multiple network domains with

Navideh Ghafouri is with Iquadrat Informatica S. L., Barcelona, Spain.
John S. Vardakas is with Iquadrat Informatica S. L., Barcelona, Spain, and Dept. of Informatics, UOWM, Greece.
Kostas Ramantas is with Iquadrat Informatica S. L., Barcelona, Spain.
Christos Verikoukis is with the ISI/ATH and CEID, University of Patras, Greece.

similar structures but different configurations. To improve scalability and efficiency while reducing complexity, we explore automation techniques that apply across multiple domains rather than being restricted to a single environment.

- *Supporting Task Diversity in Network Management*: For domain controllers, providing each network service type is a different task, since NS procedures vary by service type. We aim to develop controllers capable of managing multiple tasks efficiently without requiring complex or costly techniques. Our goal is to enable multi-service provisioning using a single, lightweight controller.
- *Optimizing the Decision-Making process in Complex Environments*: Network domains present complex decision-making challenges for intelligent controllers. By adding a pre-training phase that discovers useful assignment skills, we significantly reduce the controller's action space. Instead of evaluating all possible options, the controller selects from a refined set of optimized skills, increasing success rates and reducing computational costs.
- *Enhancing the Modularity and Scalability in 6G Orchestration*: The 6G system model is expected to incorporate hierarchical orchestration levels. Developing a practical assignment skill discovery fits as an offline phase in the multi-level management and ensures efficiency in system-wide orchestration and management.
- *Improving Generalization and Cost Efficiency*: Our proposed approach is scalable and cost-effective. The generic unsupervised RL-based skill discovery mechanism allows seamless automation of multiple network domains. The same controller, trained with pre-discovered skills, can be deployed across different domains with minimal scalability concerns.

Fig. 1 illustrates the key aspects targeted for improvement and how each contributes to enhancing EE.

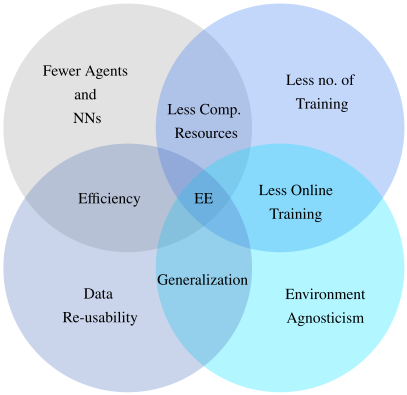

Fig. 1. The proposed approach's contributing factors to EE

II. A Multi-layer and Decentralized AI-driven Network System

While the robustness and dynamic nature of 6G necessitate decentralized and modular network architectures, managing and orchestrating such systems can become increasingly complex. To address this challenge, we propose the integration of a decentralized and modular network system with a decentralized and automated management and orchestration framework. As illustrated in Fig. 2, our architecture consists of two key components.

The upper section of Fig. 2 represents the management and orchestration layer, which is responsible for the overall network management, including slice orchestration, service provisioning, and resource allocation. This layer is typically deployed in the cloud, leveraging its abundant resources and accommodating the less time-sensitive nature of high-level orchestration tasks.

The lower section of Fig. 2 depicts the AI-enabled infrastructure, spanning multiple network domains, including the edge and far-edge. Edge domains, mostly located in the fog tier, provide computational resources, radio resource blocks, and enable localized optimization through near-real-time analytics and control. The far edge encompasses end-user equipment, IoT devices, and potentially the radio units themselves.

While the management and orchestration layer governs the network holistically, each domain must support autonomous, localized management. This decentralized architecture is critical to avoiding dependence on a central controller, which could otherwise become a bottleneck or single point of failure. Since this work focuses specifically on edge domain slicing and automation, we proceed by analyzing the architectural and functional aspects of individual network domains in more detail.

As illustrated in Fig. 2, each edge domain comprises a set of resource pools (e.g., storage and radio resources) that must be intelligently allocated to serve end-users and applications in the far edge. In addition to the resource pools, an edge domain typically includes one or more intelligent controllers and acts as a resource provider to tenants located at the far edge. Given the vast deployment of 6G networks, it is inevitable that multiple instances of these domains will co-exist. This highlights the need for automated intra-domain management, where each domain autonomously manages its resources to ensure scalability and efficiency. The domain orchestrator (controller) in Fig. 2 is responsible for assigning network slices to fulfill specific service demands. Given the dynamic nature of 6G, this domain controller cannot operate as a static entity. Instead, it must seamlessly integrate into the overall management and orchestration framework while maximizing resource utilization within the domains.

In the following section, we explore relevant ML-driven approaches and propose an offline intelligent primary-level resource quantization framework for network domains.

III. ML-based Techniques for Decentralized Network Management

RL is a well-known ML-based technique for solving real-world problems involving an intelligent agent that learns through interactions with its environment. In the context of decentralized management problems, MARL is a relevant RL-based approach, which can be cooperative, competitive, or a hybrid of both. It is important to note that variations in information sharing, coordination mechanisms, and observation

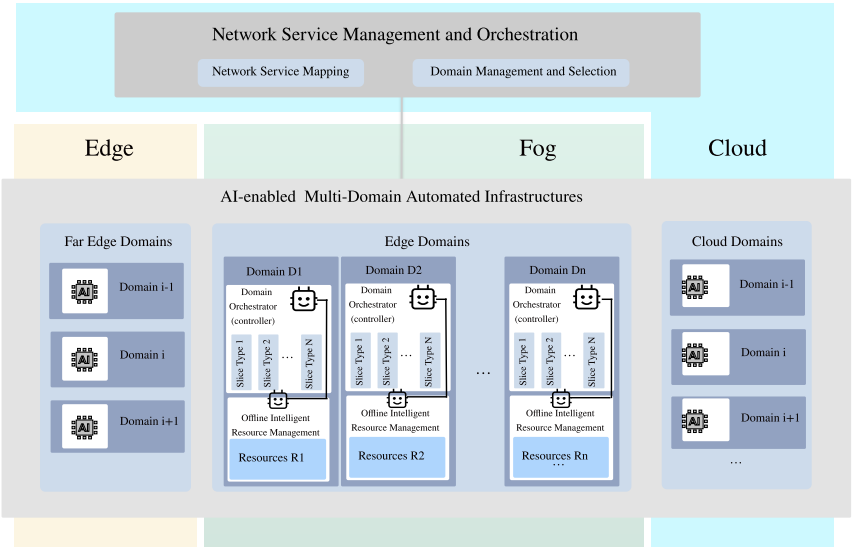

Fig. 2. Network´s general framework with a close look to the edge domains

levels can lead to different RL strategies. However, MARL often faces scalability and complexity challenges, as the presence of multiple agents significantly increases computational overhead [6].

On the other hand, Federated Learning (FL) is a decentralized ML approach in which multiple distributed servers or devices collaborate to train an ML model without sharing their local data. Although FL reduces data privacy concerns and minimizes raw data transmission, scalability issues remain due to the need for a central aggregation node. As the number of participants increases, the complexity of managing updates and coordinating learning escalates. Additionally, device heterogeneity, data inconsistency, and communication overhead pose further challenges in large-scale FL deployments [5].

While distributed RL and FL are promising for 6G network management and orchestration, in this work our focus is on minimizing complexity and maximizing efficiency. A practical management strategy should be realistically scalable and implementable in real-world systems. Given the inherent complexity of 6G networks and the anticipated challenges in NS and service provisioning, we explore strategies that strike an optimal balance between efficiency and complexity reduction. Achieving these goals requires improvements in both training and operational performance. Therefore, ML-based approaches that generalize well to new tasks and environments, without requiring retraining for every scenario, are preferable for complex systems.

Transfer Learning (TL) is one such ML technique that leverages previously acquired knowledge to improve performance on related tasks. TL consists of a pre-training phase, where a model learns generalized features, enabling it to adapt to multiple tasks without retraining from scratch. Similarly, Meta-Learning improves the learning process by making models more adaptable to new tasks with minimal data, thus enhancing overall efficiency [10].

Although TL offers efficiency by reusing trained models for multiple tasks, our problem necessitates the deployment of intelligent agents for network domain automation. As dictated by the nature of the problem, domain controllers must dynamically interact with their domains, taking sequential actions to allocate available resources for service provisioning. To achieve a beyond state-of-the-art efficiency, we propose integrating a pre-training phase using unsupervised RL. This enables domain controllers to leverage previously learned knowledge, improving decision-making while enhancing efficiency and adaptability. We introduce an offline unsupervised RL-based assignment skill discovery framework to optimize network domain management. By incorporating a pre-training phase, our approach enables simple controllers to efficiently provide multiple service types, since the complexity of the tasks and environment has been reduced for the agent.

IV. The Unsupervised RL-based Assignment Skill Learning Primary Phase

Designing a reward system tailored to a specific task and environment can effectively guide an RL agent to learn an optimal policy. However, manually crafting reward functions becomes impractical and costly when managing a network of distributed intelligent entities with various tasks. Furthermore, while 6G networks are inherently heterogeneous, distributed, and dynamic, they still follow unified global objectives. These global objectives require a long-horizon sequence of actions in multiple levels. Thus, single-objective agents with fixed goals appear less practical. This makes unsupervised RL a compelling approach for enabling adaptability and efficiency in 6G automation.

Unsupervised RL refers to a class of RL methods where agents learn useful behaviors, representations, or models without access to task-specific reward signals. Instead, learning is driven by intrinsic objectives such as maximizing diversity, predictability, or information gain. The goal is to acquire general-purpose skills, structured policies, or state abstractions that can later be reused or adapted for a variety of downstream tasks, often with minimal additional training.

The primary advantage of applying unsupervised RL in 6G networks is that it enables agents to develop generalizable

unsupervised policies, allowing them to adapt to multiple tasks without predefined rewards. However, the impact of the environment on an agent remains significant. To ensure that agents can generalize across multiple tasks and network domains, they must undergo pre-training in multiple reward-free environments, environments that belong to the same domain class but feature different transition dynamics [4].

Unsupervised learning in RL poses challenges, as the agent must collect useful data in a pre-training phase without explicit rewards. Recent research suggests that skill discovery is a promising approach for pre-training, as it facilitates the learning of diverse behaviors that can later be fine-tuned or leveraged in HRL [8].

Skill discovery involves learning latent behaviors, referred to as skills, through an intrinsic reward system [9]. These skills are acquired without prior task knowledge, making them useful for adapting to a wide range of tasks [11]. Given its advantages, we propose integrating a skill discovery phase as the primary stage for edge domain automation in 6G.

The Diversity Is All You Need (DIAYN) algorithm [11] is designed to identify a diverse set of skills without relying on a handcrafted reward function. Instead, it maximizes the utility of the learned skill set, ensuring that the discovered skills are distinct and meaningful.

A latent-conditioned policy is one that conditions its action outputs on both the current state and a latent variable representing a skill. Unlike traditional policies, which map states to actions, DIAYN policies receive an additional latent skill variable, influencing the agent's decision-making process.

DIAYN maximizes the mutual information between states and skills, effectively empowering the controller agent by allowing it to explore a diverse set of behaviors. More specifically, DIAYN optimizes the predictability of the skill given the state, the entropy of the skill distribution, ensuring diverse behaviors,and the entropy of the policy, encouraging broader exploration.

By maximizing mutual information between states and skills, DIAYN enables agents to control and explore different states, ultimately resulting in a richer set of learned skills. The mutual information objective is mathematically represented as the entropy of the skill, minus the entropy of the skill given the state, encouraging distinct and meaningful skill representations.

DIAYN is implemented on top of the Soft Actor-Critic (SAC) algorithm [11], which is widely used for solving continuous control tasks in complex environments. SAC is an entropy-regularized RL algorithm, which means that while it seeks to maximize cumulative rewards, it simultaneously maintains a high policy entropy to encourage exploration.

SAC employs a stochastic policy, assigning probabilities to possible actions and sampling from these probabilities to ensure diverse action selection. By incorporating entropy regularization, SAC inherently supports exploration, which aligns well with DIAYN's objective of discovering distinguishable skills. Consequently, DIAYN's neural networks only need to maximize the predictability of a skill given a state, rather than crafting a predefined reward function [11].

In the proposed 6G edge domain automation framework, domain management begins with a pre-training phase consisting of unsupervised interactions with the environment to discover useful skills [4], [9]. These discovered skills can then be leveraged in multiple ways. Some of the possible applications in the higher levels of management are as follows:

1) Fine-tuning with supervised training: The primary policy undergoes additional supervised training with a reward function, which improves task-specific performance [4].
2) HRL deployment: The pre-trained skills are used as the lower level of HRL, enabling adaptive task execution in a modular and scalable fashion.
3) Basic controllers reusing useful data: By incorporating unsupervised RL-based skill discovery, our approach enables simpler controllers to efficiently handle multiple service types, significantly reducing complexity while improving adaptability and performance.

V. Our Proposed Approach: Generic unsupervised skill discovery for automation of distributed network domains

This section outlines the proposed primary phase of unsupervised skill discovery, which enables efficient service provisioning by the controller, as illustrated in Fig. 3.

Each network domain consists of four types of resource pools: computational resources (e.g., processors), transmission power, bandwidth, and memory. At the pre-training level, the unsupervised RL agent interacts directly with these resource pools to discover useful assignment skills without supervision. These skills are then leveraged by the domain controller, which is responsible for providing various network services to tenants.

Since each skill represents a set of resource assignments, the controller applies these skills to allocate network slices for tenants as required. Notably, the nature of tasks aligns with skills, as both involve resource allocation. Fig. 2(b) illustrates the domain management mechanism, progressing from left to right. In the offline pre-training phase, the unsupervised RL agent discovers useful assignment skills. Since network domains differ only in details, the discovered skills can be generalized and applied across all domains.

Adding a pre-training phase to discover the useful assignment tasks offers the following benefits:

- *Discretization of Resources*: In network slicing, resources are allocated based on an upper limit, a lower limit, or a combination of both. Each resource pool consists of an infinite range of possible values, yet only a subset is useful for network slicing. Thus, the observation space of the controller is sparse and infinite, making direct optimization difficult. Pre-trained skills discretize the resource space, eliminating the need for complex controllers while increasing the likelihood of successful slice assignments.
- *Generalization Across Network Domains*: Since network domains share structural similarities, discovering a set of useful skills removes the need for separate training in every domain. By defining skills as percentages rather

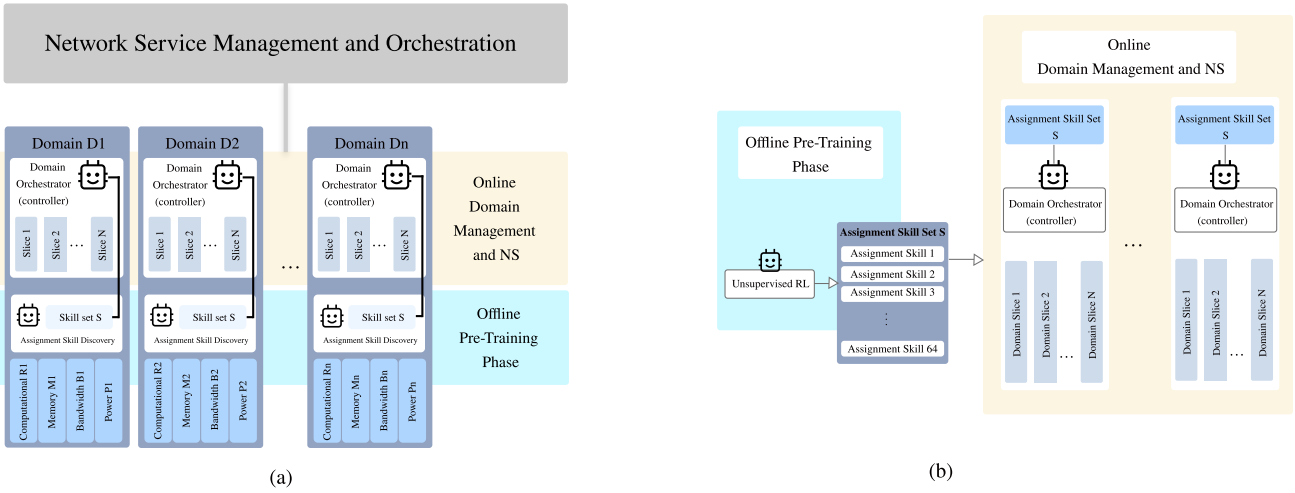

Fig. 3. Automation of edge domains: (a) Domains' management scheme in the system model, (b) A closer look to the offline and online phases

than absolute resource values, the discovered skills can adapt to domains with resource pools of varying sizes.

- *Handling Multiple Service Types Efficiently*: In addition to having multiple different domains, multiple service types exist in the network. This means that network domains serve multiple service types, which means that each controller must handle diverse tasks. A traditional RL agent trained for a single goal would be inefficient in such an environment. However, by discovering reusable skills, the nature of the problem shifts: the controller only needs to select a sequence of skills from a finite skill set until the constraints of the target task are met. This allows a simple, lightweight controller to efficiently manage multiple service types.
- *Scalability and Complexity Reduction*: This approach proposes one offline pre-training phase that enables the domain to be automated by a simple controller. This ensures that multiple edge domains can operate efficiently without a surge in complexity.
- *Multi-Level Management*: We envision the management of this decentralized system model to be in multi-levels. The discovered skills can be used in various ways, including by a simple controller, by fine-tuning the model, or by incorporating HRL for adaptive orchestration.

A. Implementation Details

The following paragraphs detail our implementation of the "Unsupervised RL-Based Assignment Skill Discovery for Network Edge Domains" framework.

The environment in our implementation is modeled as a network domain, as illustrated in Fig. 3. While this same environment includes the controller and handles the provisioning of resource slices to tenants, during the offline pre-training phase, the agent interacts only with the resource pools. This design choice reflects the fact that, in this phase, the goal is solely to discover unsupervised skills, without considering user requests or service-level constraints.

At a higher level, the controller will later observe user requests, network dynamics, and Service-Level Agreements (SLAs) to orchestrate service provisioning using the skills learned during pre-training. Each network slice is composed of four key resource types: computational capacity, memory, bandwidth, and power.

To maintain generality and applicability across different deployment scenarios, we represent both states and actions as percentage-based allocations rather than absolute values. This abstraction enables flexible and scalable learning across varying resource scales. Accordingly, the state and action space used in the pre-training phase is defined as follows:

$$\text{State space} = [2-10]\%, [2-10]\%, [2-10]\%, [2-10]\%$$
$$\text{Action space} = [1-5]\%, [1-5]\%, [1-5]\%, [1-5]\%$$

The observation space indicates that only 10 % of each resource type is utilized for skill discovery. In practice, no resource type can be zero for service provision. As a result, the lower bound of the state space is 2 %. The upper bound is set at 10 % to introduce distinct states per episode, simulating the slight differences between domains. This is because each episode starts with a different minimum and maximum percentage in the state space, which can simulate the slight differences existing in different domains.

Since the maximum allocation per skill is capped at 20 %, an upper bound higher than 20 would limit the number of time steps in each episode. Action can be a continuous number between 1 to 5 %. It is worth mentioning that the action space in this level consists of infinite real numbers, the discovered skills form a finite set of discrete actions for the higher-level controller.

We configure the agent to discover 64 distinct skills, a choice informed by empirical trials aimed at balancing sufficient state-space coverage with the discovery of only relevant and diverse behaviors. The number of skills is determined primarily by the number of resource types, as the agent in this phase only observes the values of the resource pools, independent of other domain complexities such as user behavior or network topology.

When the number of resource types increases, it becomes reasonable to increase the number of skills, as the space of

possible combinations expands accordingly. Additionally, the level of granularity (i.e., how finely the resource values are discretized) can also influence the optimal number of skills. As such, the number of skills may vary across implementations depending on the domain's design and resolution.

Nevertheless, the algorithm remains scalable, as the number of resource types in a network domain is typically limited, and increasing the number of skills does not introduce significant additional complexity during training or inference.

In this way, the infinite state space of the domain controller is reduced to a finite number of skills. Thus, while the controller state space will only include the most useful hand-picked assignment sequence, the action space will also be reduced to selecting a series of skills until the service is provided.

As discussed in the previous section, DIAYN employs an intrinsic reward mechanism and is fundamentally task-agnostic, which means that no explicit external reward is defined. Instead, the most distinct combination of values (distinct states) is discovered. However, the environment still requires termination conditions, or final states, to end each episode. In our setup, we define useful states as final states (i.e., when a known combination of four resource values is observed, the episode is terminated. Importantly, this does not compromise the task-agnostic nature of the DIAYN framework. The actual downstream task is to select appropriate resource values for network slice provisioning. Therefore, defining final states that reflect certain numerical patterns or thresholds does not interfere with this task. Instead, it simply provides structure to the exploration process since there are infinite states. A final state can occur either when at least one resource type reaches a predefined maximum threshold (e.g., 20% of the total available resources), or when the combination of allocated resources matches a known pattern of four values that indicates a distinct skill. Thus, if a skill matches any possible combination of an array consisting of 4 binary numbers, the mentioned skill can be useful for the controller agent. Therefore, the states are normalized and compared to all combinations of a binary array with four elements. If the normalized state is close to any combination (with the relative tolerance of $1e-01$ and absolute tolerance of $1e-02$), that state is a final state.

The 20% threshold ensures that the learned skills remain within reasonable operational bounds, promoting resource efficiency and fairness. This value can be adjusted depending on the desired level of granularity or resource sensitivity in a given deployment scenario.

The RL agent was trained for 25,000 episodes with a *learning rate* of $1e-5$, a *gamma* equal to 0.99, α equal to 0.01, and τ equal to 0.001. Each discovered skill is an array that represents the percentage allocation of power, bandwidth, memory, and computational resources. Although the skills are not visualized here due to space constraints, detailed analysis confirms that each skill is distinct and maintains an upper bound of 20%. Since we need to confirm that replacing the state space of the controller with the set of discovered skills does not limit its access for adequate service provisions, we investigate the discovered skills in the entire space of the state space.

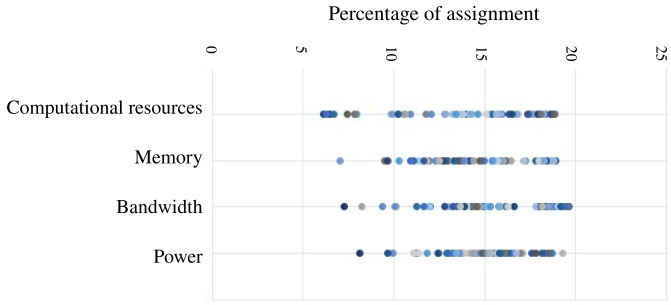

Fig. 4. Coverage of the discovered skills across the resources of each domain

While Fig. 4 embeds the skill values for each resource type, it also demonstrates that a finite number of skills have extended across approximately 5 to 20 percent of the infinite state space of each resource pool. This significant simplification of the environment and action space allows unknown network slicing tasks to be managed with minimal complexity, computational overhead, and energy consumption (which are typically required for complicated RL algorithms) across different domains at a higher level.

VI. Energy Efficiency

With the rapid expansion of wireless network applications and their increasing roles in our lives, green communication has become a critical concern in the telecommunication society. As a broad concept, green communication refers to environmentally friendly networking, where every stage, from design and implementation to management, is optimized to minimize environmental impact. Although there are numerous design strategies and technologies that enhance the energy efficiency of wireless networks, efficient management and orchestration are equally crucial.

Softwarization, virtualization, and AI provide a transformative opportunity for flexible and energy-efficient resource management by reducing the idle time of hardware network components. However, AI-driven network management requires computational resources, which in turn utilize hardware and energy. This makes the development of scalable, efficient, and generic AI-based solutions both challenging and necessary in the context of tailored service provisioning.

Table I presents a comparison between our proposed solution and the state-of-the-art RL techniques in terms of EE. The comparison considers key factors introduced in Fig. 1, such as the number of deployed agents, neural networks, and training iterations, which directly impact computational resource demands, as well as data reusability, environment independence, and generalization, which are critical for scalability.

Table I demonstrates that our approach achieves superior re-usability and generalization compared to other methods for automating service provisioning in network domains. Furthermore, after evaluation and hyperparameter optimization, our method requires fewer training runs and a smaller number of neural networks, resulting in significantly reduced computational overhead and enhanced energy efficiency. These advantages become even more pronounced when the framework is scaled across multiple network domains. Moreover, Fig. 5 visualizes how our proposed approach is superior by

TABLE I
COMPARISON OF ASPECTS AFFECTING EE IN DIFFERENT RL TECHNIQUES

Approach	Data Reuse	Scalability (w.r.t. services)	Env. Dependency	Generalization Across Domains	No. of Trainings*	No. of Agents	Neural Networks	Comp. Resources
DIAYN + Controller	High	High	Low	High	~ 2 (DIAYN + controller)	~ 2 (DIAYN + controller)	~ 4 (DIAYN + controller)	Moderate
DIAYN + HRL	High	High	Moderate	High	~ 1	1	~ p+ (p = policy levels)	Moderate to High
MARL (Co-operative)	Moderate	Low	Moderate	Moderate	n (one per service/agent)	~ n	~ n	High
MARL (Competitive)	Low	Very Poor	High	Low	~ n^2 (train for every pair of services/agents)	~ n	> n	Very High
MARL (Mixed)	Moderate	Moderate	Low to Moderate	Moderate	~ n << n^2	~ n+	> n	High
MORL (Preference-Based)	Moderate	Moderate	Moderate	Moderate	~ k (preference vectors)	1 agent with multi-objective input	1 << k	Moderate to High
Traditional HRL	Low to Moderate	Low	High	Low	~ nZ + 1 (Z = subpolicies per task)	1 agent with task-specific subpolicies	~1 controller + Σ subpolicy NNs	High

* Final training, after research-quality evaluation and determining the optimal hyperparameter values, without considering different seeds.

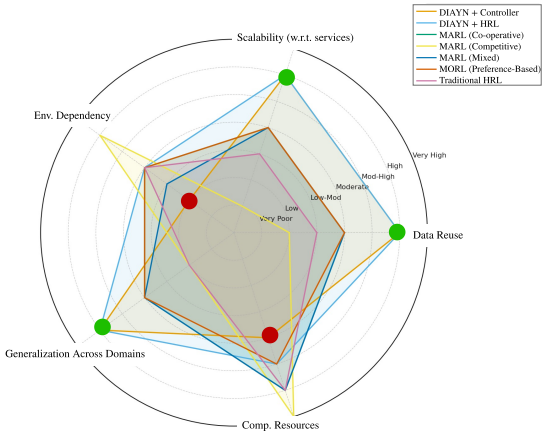

Fig. 5. Comparison of RL techniques across key aspects in Fig.1

having lower comparative values in the red points (environment dependency and computational resources) and higher comparative values in the green points (data reuse-ability, generalization, and scalability).

VII. CONCLUSIONS AND FUTURE STEPS

With 6G rapidly approaching, this research focuses primarily on practical, scalable, and energy efficient AI-based network management. Following a brief review of edge RL techniques suitable for decentralized network automation, we introduce a modular and decentralized management framework. We then propose an offline pre-training phase for unsupervised RL-based skill discovery, aimed at enhancing efficiency and scalability in network management.

Although this work explores the application of unsupervised skill discovery for domain controllers and HRL-based orchestration, the full implementation of these higher-level management strategies is not included. The offline unsupervised skill discovery can be leveraged in three ways: directly by a domain controller, through fine-tuning the pre-trained model, or as part of an HRL-based management framework for service provisioning.

This research can be extended in future work by implementing and evaluating higher-level controllers, further exploring their role in enabling scalable, efficient, and intelligent orchestration in 6G networks.

ACKNOWLEDGEMENT

This work has received funding from the European Union under the ADROIT6G project (Grant agreement ID: 101095363), by the SUNRISE-6G project (Grant agreement ID: 101139257), and by the H.F.R.I project ENABLE-6G (ID:16294).

REFERENCES

[1] X. Zhang, P. Han, C. Feng, T. Ma and L. Guo, "Multi-Dimensional Resource Orchestration Toward Edge Intelligence in 6G Networks," in IEEE Communications Magazine, vol. 61, no. 12, pp. 46-52, Dec 2023.
[2] N. Ghafouri, J. S. Vardakas, K. Ramantas, and C. Verikoukis, "Rl-based high-level radio unit clustering and distributed unit assignment in user-centric cell-free mmimo for oran-based 6g," in Proc. IEEE ICC 2024.
[3] T. Huang, W. Yang, J. Wu, J. Ma, X. Zhang, and D. Zhang, "A survey on green 6g network: Architecture and technologies," IEEE access, vol. 7, pp. 175 758–175 768, 2019.
[4] M. Mutti, M. Mancassola, and M. Restelli, "Unsupervised reinforcement learning in multiple environments," in Proc. AAAI Conference on AI, 2022.
[5] Y. C. Lai, C. Y. Wang and F. T. Chien, "When Green Learning Meets Federated Learning: Toward Distributed Learning with Low Complexity and Model Heterogeneity," ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Korea, Republic of, 2024, pp. 7615-7619.
[6] N. Ghafouri, J. S. Vardakas, K. Ramantas, and C. Verikoukis, "A multi-level deep rl-based network slicing and resource management for o-ran-based 6g cell-free networks," IEEE Transactions on Vehicular Technology, 2024.
[7] ——, "High-level service type analysis and MORL-based network slice configuration for Cell-Free-based 6G Networks", IEEE Trans. Veh. TEch., accepted, 2024.
[8] Bozi c, P. Tadi c, and M. Nikoli c, "Intrinsically motivated option learning: a comparative study of recent methods," in Proc. IEEE TELFOR 2021.

[9] J. RHIM, E. Yang, and T. Kim, “Efficient task adaptation by mixing discovered skills,” in First Workshop on Pre-training: Perspectives, Pitfalls, and Paths Forward at ICML 2022.
[10] R. Upadhyay, R. Phlypo, R. Saini and M. Liwicki, ”Sharing to Learn and Learning to Share; Fitting Together Meta, Multi-Task, and Transfer Learning: A Meta Review,” IEEE Access, 2024.
[11] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine, “Diversity is all you need: Learning skills without a reward function,” arXiv preprint arXiv:1802.06070, 2018.

BIOGRAPHIES

Navideh Ghafouri is a PhD candidate at the Polytechnic University of Catalunya and working as a junior researcher in Iquadrat Informatica S.L. Her current focus is on the management of future wireless networks by deploying RL techniques.

John S. Vardakas received his Ph.D from the ECE Dept., University of Patras, Greece in 2012. His research interests include teletraffic engineering, performance analysis / simulation of communication networks and smart grids.

Kostas Ramantas received his Ph.D in 2012 from CEID, University of Patras, Greece. His research interests are in modeling and simulation of network protocols; and scheduling algorithms for QoS provisioning.

Christos Verikoukis received the Ph.D. degree from UPC, Barcelona, Spain, in 2000. He is currently an Associate Professor at the CEID, University of Partas, Greece.