

Dynamic Preference Multi-Objective Reinforcement Learning for Internet Network Management

DongNyeong Heo¹, Daniela Noemi Rim¹, Heeyoul Choi¹

¹Computer Science and Electrical Engineering, Handong Global University, Pohang, South Korea.
dnheo@handong.ac.kr, danielarim@handong.ac.kr, hchoi@handong.ac.kr

Summary

This paper proposes a novel reinforcement learning (RL) approach for internet network management (NM) that facilitates the RL agent can handle dynamic preference scenario. Traditional RL-based NM methods typically use fixed preferences to optimize multiple objectives like quality of service (QoS) and computing resource usage. However, in real-world scenarios, preferences of the multiple objectives can be changed dynamically due to factors such as network overloads or server failures. We present a method that allows RL agents to adapt to dynamic preferences during testing. Our experiments show that the proposed method significantly improves generalizability across various network preferences compared to previous RL methods, offering a more efficient and flexible solution for NM.

Background

Internet network service provider administrates the computer network with operating several modules. Representatively, virtualized network function (VNF) auto-scaling (AS) [1] and power management (PM) [2] modules operate to deploy/withdraw VNF instance on a computing server based on multiple objectives, including maintaining high QoS and minimizing computing resources. Due to the emergence of large-scale internet network, reinforcement learning (RL) based agent was proposed to approximate the NM modules [3,4]. The previous RL agent [1] has trained to optimize the objectives using static preferences that weigh the multiple objectives' relative importance based on proximal policy optimization (PPO) method [5]. However, in practical scenarios, these preferences change in response to various conditions, such as network overloads or server shutdowns. In results, the traditional fixed preference RL (FP-RL) agents struggle to adapt to these various preference settings, except the setting that the agent has trained on. Therefore, there is a need for an RL agent capable of adapting to dynamic preference settings, improving/preserving its performance under diverse conditions without requiring additional training or multiple agents trained for different preferences.

Methodology

We extend the PPO algorithm to include dynamic preferences. By incorporating preferences as an additional input into the policy network's state input, we allow the agent to adjust its

actions according to diverse network conditions. Specifically, the input consists of the network state (e.g., server load, traffic latency) and preferences (e.g., QoS vs. resource usage). During the training, the RL agent is trained on a given preference distribution rather than fixed values, allowing it to generalize more effectively across different network status. Additionally, we propose a numerical method to estimate an optimal preference distribution with respect to generalizability. In this work, we applied this dynamic preference RL (DP-RL) agent to the two network management modules: AS and PM. We note the AS module regards two objectives, QoS and resource usage, and PM module additionally regards power consumption on top of the AS module's objectives. Therefore, AS and PM modules contain 1 (for resource usage) and 2 (for resource usage and power consumption) relative preferences, respectively.

For experiments, we used the simulation datasets that are based on two typical network topologies: Internet2 and mobile edge computing (MEC), and real-world network traffic data [1]. Following the conventional method of the previous work [1], the VNF instance deployment is firstly determined by a dynamic programming method, then it is randomly perturbed to be sub-optimal for AS and PM modules' training.

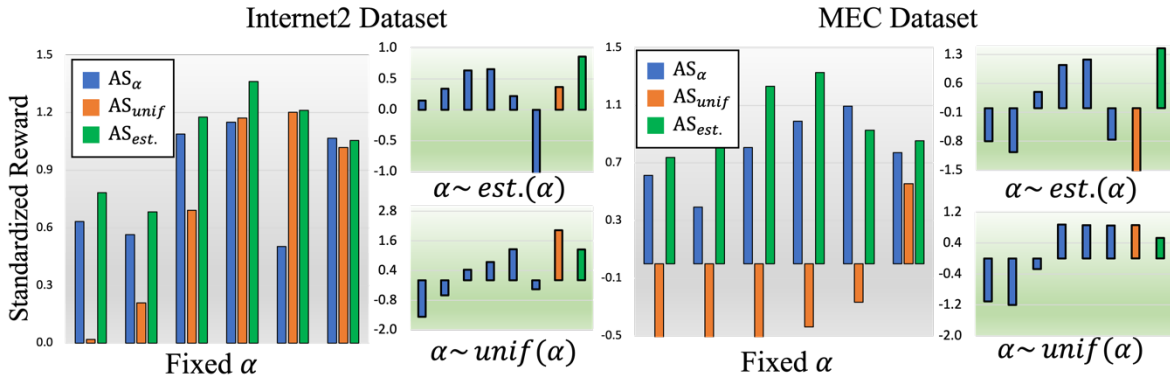


Fig. 1. Experiment results of AS agents. α refers to the relative preference value of the computing resource usage objective. *est.*(α) and *unif.*(α) mean the preference distributions of the estimated one and the arbitrary uniform, respectively. Standardized reward means the average reward relatively scaled by other comparable agents' reward results.

		Internet2 Dataset						MEC Dataset					
Fixed α	Fixed β	0.25	0.13	0.24	0.48	0.44	0.46	0.87	1.01	0.68	0.8	1.47	3.3
	Fixed β	-0.03	0.05	0.78	0.83	1.14	0.37	1.29	1.33	1.95	0.91	1.3	2.88
	Fixed β	-0.32	-0.16	0.2	0.55	0.23	0.21	3.05	1.95	1.52	0.68	4.04	1.06
		Fixed β						Fixed β					

Fig. 2. Experiment results of PM agents. α and β refer to the relative preference values of the computing resource usage and power consumption objectives, respectively. Each value means the difference of the standardized rewards of baseline and DP-RL agent. Positive means DP-RL agent is better than the baseline.

Results and Observations

For comparison, we trained the baseline FP-RL agents with different preferences. For our DP-RL agents, we trained them with the preference distribution estimated by our numerical

method. To check the advantage of our numerical method, we additionally trained DP-RL agent with an arbitrary uniform preference distribution in the AS task.

For the evaluation of AS agents, we conducted two types of tests that are static and dynamic preference tests for all of the agents. During the static preference test, we set the same fixed preference of a baseline RL agent's training setting. Analogously, during the dynamic preference test, we set the preference changes at every episode according to the distributions that are the DP-RL agents' training setting. As shown in Fig. 1, DP-RL agent that is trained by the estimated distribution, $AS_{est.}$ (green), consistently perform as well or better than baselines, AS_{α} (blue) in the static preference tests (left graphs of each dataset). In the results of dynamic preference tests (right bottom/top graphs of each dataset), DP-RL agents demonstrate superior adaptability compared to the baselines. In addition, $AS_{est.}$ generalizes well to various preference settings, while the DP-RL agent trained by the arbitrary uniform distribution, AS_{unif} , leads to diminished performances across several settings.

For the evaluation of PM agents, we conducted the static preference test for all agents. Similar to the results of AS agents, as demonstrated in Fig. 2, DP-RL agent performs similar or better than the baselines in various fixed preference settings. Based on all our experiment results, our DP-RL method is suitable for training generalizable NM modules in various scenarios.

Conclusion

This study introduces a dynamic preference multi-objective reinforcement learning framework in NM domain. It enables RL agents to adapt to varying NM objectives in real-time. By training the agent with dynamic preferences, we enhance its generalizability and robustness in managing network services like AS and PM. The proposed approach not only improves/preserves performance under diverse conditions but also reduces the need for multiple specialized agents, providing a more efficient solution for complex NM tasks.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Education (NRF-2022R1A2C1012633)

References

- [1] Seo, N., Heo, D., Hong, J., Kim, H. G., Yoo, J. H., Hong, J. W. K., and Choi, H., 2022. Updating VNF deployment with Scaling Actions using Reinforcement Algorithms. *23rd Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pp. 1-4.
- [2] Abbas, K., Hong, J., Van Tu, N., Yoo, J. H., and Hong, J. W. K., 2022. Autonomous DRL-based energy efficient VM consolidation for cloud data centers. *Physical Communication*, 55, 101925.
- [3] Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., and Uhlig, S., 2014. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1), pp.14-76.
- [4] Mijumbi, R., Serrat, J., Gorricho, J. L., Bouten, N., De Turck, F., and Boutaba, R., 2015. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications surveys & tutorials*, 18(1), pp.236-262.
- [5] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O., 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.