Learning Reward Structure with Subtasks in Reinforcement Learning

Shuai Han[1], Mehdi Dastani[1], and Shihan Wang[1]

Utrecht University, Netherlands {s.han, m.m.dastani, s.wang2}@uu.nl

Reinforcement learning (RL) has been applied in a variety of domains, such as traffic signal control [1, 2], chemical structure prediction [3, 4], and radio resource management [5, 6]. The successful training of RL agents often relies on reward functions that are designed based on domain knowledge. Such reward functions allow agents to receive immediate reward signals. Without those handcrafted signals, the sparse rewards can result in RL algorithms suffering from low sample efficiency [7, 8]. Numerous methods have been proposed to enhance sample efficiency of RL in sparse-reward environments, such as building goal-conditioned reinforcement learning (GCRL) to provide intrinsic rewards [9, 10, 11], applying hierarchical reinforcement learning (HRL) for improving credit assignment [12, 13, 14], or employing reward machines (RM) to expose the structure of reward functions [15, 16, 17].

In many scenarios, accomplishing a task involves the sequential completion of multiple subtasks. Especially in sparse-reward settings where immediate feedback is scarce, evaluating action selection relies heavily on precise information about past subtask completions and their specific order. However, previous methodologies, such as GCRL and HRL, do not incorporate precise information about the sequential order of subtasks into their policy learning frameworks. On the other side, the recently proposed RM specify the reward function structure as an automaton [15, 18], which provides crucial information about the sequential nature of subtasks. To construct a reward machine for a given set of subtasks, previous methods have proposed to use automata learning to infer a automaton to describe and exploit the reward function structure [19, 20]. However, learning an exact automaton from trace data is a NP-complete problem [21]. Although heuristic methods can be used to speed up the learning [19], inferring an automaton that is representative to the reward structure relies on trace data which is collected by an adequate exploration. When the exploration of agents is inadequate, the automaton derived from the incomplete trace data could be either inaccurate or partial, which leads to the RL algorithm learning sub-optimal policies or even failing to learn.

Aiming at improving sample efficiency of RL in the above-mentioned sparse-reward scenarios that involve sequential completion of multiple subtasks, we propose a novel algorithm, which we call Automatically Learning to Compose Subtasks (ALCS). It automatically learns the structure of the reward based on a given set of subtasks (i.e. constituting the minimal domain knowledge of

This paper was accepted by the 27th European Conference on Artificial Intelligence (ECAI 2024).

the task). The key idea of ALCS is to learn the best sequences of subtasks to achieve the learning task. To accomplish this, we develop a framework with two-level hierarchy of policy learning. The low-level policy learns to take the next action toward completing a given subtask, while the high-level policy learns to specify a subtask to be achieved next. There are two main characteristics of the high-level policy learning. One is that the next subtask is selected based on the exact sequence of completed subtasks, which considers precise information about subtask sequences during learning. Another characteristic is that at the end of an episode, the subtasks selected by the high-level policy are modified based on the subtasks actually achieved by the low-level policy. This is necessary to consider the impact of all achieved subtasks on the reward gains so that those achieved subtasks can be reinforced as the policy selection. We verify the performance of our method on 8 sparse-reward environments. The results show that when the difficulty of tasks increases, our method produces a significant improvement over the previous most sample-efficient methods.

Evaluations

We first compare our method with baselines on 8 environments from *OfficeWord* and *MineCraft* domains to validate the superiority of ALCS. The results are shown in Figure 1. The results show that algorithms that cannot utilize information about subtasks already performed, such as Q learning, are unable to learn the optimal policy in these domains. Our method significantly outperforms the baseline methods (including the state-of-the-art methods JIRP and DeepSynth) in all environments except Coffee where the reward structure is simple and easy to explore. However, when the reward structure of the task becomes complex, the sample efficiency of ALCS can significantly outperform all other methods.

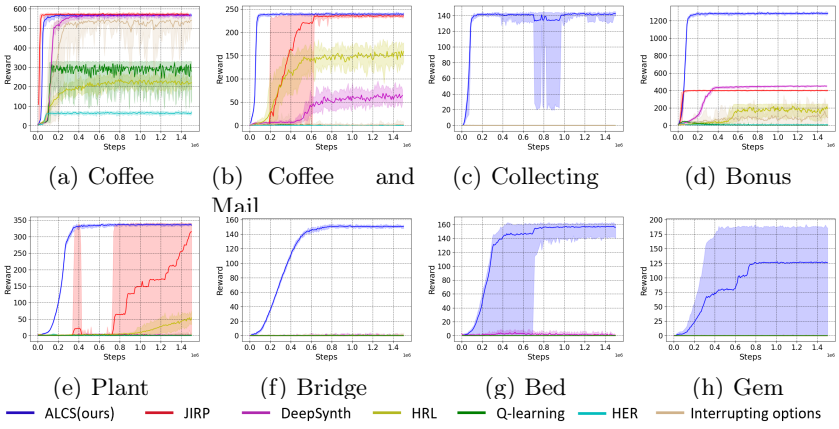

Fig. 1. Learning curves of various RL algorithms on 8 environments from *OfficeWord* and *MineCraft* domains.

Reference

[1] Chacha Chen, Hua Wei, Nan Xu, Guanjie Zheng, Ming Yang, Yuanhao Xiong, Kai Xu, and Zhenhui Li. Toward A thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 3414–3421, 2020.

[2] Qize Jiang, Minhao Qin, Shengmin Shi, Weiwei Sun, and Baihua Zheng. Multi-agent reinforcement learning for traffic signal control through universal communication method. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pages 3854–3860, 2022.

[3] Hwanhee Kim, Soohyun Ko, Byung Ju Kim, Sung Jin Ryu, and Jaegyoon Ahn. Predicting chemical structure using reinforcement learning with a stack-augmented conditional variational autoencoder. *J. Cheminformatics*, 14(1):83, 2022.

[4] Luca A. Thiede, Mario Krenn, AkshatKumar Nigam, and Alán Aspuru-Guzik. Curiosity in exploring chemical spaces: intrinsic rewards for molecular reinforcement learning. *Mach. Learn. Sci. Technol.*, 3(3):35008, 2022.

[5] Mohammad Zangooei, Niloy Saha, Morteza Golkarifard, and Raouf Boutaba. Reinforcement learning for radio resource management in RAN slicing: A survey. *IEEE Commun. Mag.*, 61(2):118–124, 2023.

[6] James Delaney, Steve Dowey, and Chi-Tsun Cheng. Reinforcement-learning-based robust resource management for multi-radio systems. *Sensors*, 23(10): 4821, 2023.

[7] Marcin Andrychowicz, Dwight Crow, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. pages 5048–5058, 2017.

[8] Abhishek Gupta, Aldo Pacchiano, Yuexiang Zhai, Sham Kakade, and Sergey Levine. Unpacking reward shaping: Understanding the benefits of reward engineering on sample complexity. *Advances in Neural Information Processing Systems*, 35:15281–15295, 2022.

[9] Meng Fang, Tianyi Zhou, Yali Du, Lei Han, and Zhengyou Zhang. Curriculum-guided hindsight experience replay. *Advances in neural information processing systems*, 32, 2019.

[10] Minghuan Liu, Menghui Zhu, and Weinan Zhang. Goal-conditioned reinforcement learning: Problems and solutions. *arXiv preprint arXiv:2201.08299*, 2022.

[11] Cédric Colas, Tristan Karch, Olivier Sigaud, and Pierre-Yves Oudeyer. Autotelic agents with intrinsically motivated goal-conditioned reinforcement learning: a short survey. *Journal of Artificial Intelligence Research*, 74: 1159–1199, 2022.

[12] Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. Hierarchical reinforcement learning: A comprehensive survey. *ACM Computing Surveys (CSUR)*, 54(5):1–35, 2021.

[13] Mehdi Zadem, Sergio Mover, et al. Goal space abstraction in hierarchical reinforcement learning via set-based reachability analysis. In *2023 IEEE International Conference on Development and Learning (ICDL)*, pages 423–428. IEEE, 2023.

[14] Tejas D Kulkarni, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

[15] Rodrigo Toro Icarte, Toryn Q. Klassen, Richard Anthony Valenzano, and Sheila A. McIlraith. Reward machines: Exploiting reward function structure in reinforcement learning. *J. Artif. Intell. Res.*, pages 173–208, 2022.

[16] Alberto Camacho, Rodrigo Toro Icarte, Toryn Q. Klassen, Richard Anthony Valenzano, and Sheila A. McIlraith. LTL and beyond: Formal languages for reward function specification in reinforcement learning. In *Proceedings of the Twenty-Eighth IJCAI*, pages 6065–6073, 2019.

[17] Hippolyte Bourel, Anders Jonsson, Odalric-Ambrym Maillard, and Mohammad Sadegh Talebi. Exploration in reward machines with low regret. In *International Conference on Artificial Intelligence and Statistics*, pages 4114–4146, 2023.

[18] Cyrus Neary, Zhe Xu, Bo Wu, and Ufuk Topcu. Reward machines for cooperative multi-agent reinforcement learning. In *Proceedings of the 2021 International Conference on Autonomous Agents and Multiagent Systems*, pages 934–942, 2021.

[19] Zhe Xu, Ivan Gavran, Yousef Ahmad, Rupak Majumdar, Daniel Neider, Ufuk Topcu, and Bo Wu. Joint inference of reward machines and policies for reinforcement learning. In *Proceedings of the Thirtieth International Conference on Automated Planning and Scheduling*, pages 590–598, 2020.

[20] Mohammadhosein Hasanbeig, Natasha Yogananda Jeppu, Alessandro Abate, Tom Melham, and Daniel Kroening. Deepsynth: Automata synthesis for automatic task segmentation in deep reinforcement learning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, pages 7647–7656, 2021.

[21] E. Mark Gold. Complexity of automaton identification from given data. *Inf. Control.*, 37(3):302–320, 1978.