# Matching Options to Tasks using Option-Indexed Hierarchical Reinforcement Learning

Extended Abstract

Kushal Chauhan
Google Research
kushalchauhan@google.com

Soumya Chatterjee
Google Research
soumyach@google.com

Akash Reddy
IIT Madras
ee17b001@smail.iitm.ac.in

Aniruddha S
IIT Madras
ep18b029@smail.iitm.ac.in

Balaraman Ravindran
IIT Madras
ravi@cse.iitm.ac.in

Pradeep Shenoy
Google Research
shenoypradeep@google.com

## ABSTRACT

The options framework in Hierarchical Reinforcement Learning breaks down overall goals into a combination of simpler tasks (options) and their policies, allowing for abstraction in the action space. Ideally, options can be reused across different goals; indeed, this is necessary to build a continual learning agent that can effectively leverage its prior experience. Previous approaches allow limited transfer of pre-learned options to new task settings. We propose a novel option indexing approach to hierarchical learning (OI-HRL), where we learn an affinity function between options and items present in the environment. With OI-HRL, we effectively reuse a large library of pre-trained options in zero-shot generalization at test time by restricting goal-directed learning to relevant options alone. We develop a meta-training loop that learns the representations of options and environments over a series of HRL problems by incorporating feedback about the relevance of retrieved options to the higher-level goal. Our model is competitive with oracular baselines and substantially better than a baseline with the entire option pool available for learning the hierarchical policy.

## KEYWORDS

Hierarchical Reinforcement Learning; Option Indexing

**ACM Reference Format:**
Kushal Chauhan, Soumya Chatterjee, Akash Reddy, Aniruddha S, Balaraman Ravindran, and Pradeep Shenoy. 2023. Matching Options to Tasks using Option-Indexed Hierarchical Reinforcement Learning: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

## 1 INTRODUCTION

Suppose an agent encounters a new task in a new environment. Without prior experience, the agent will need to explore and interact with the environment using primitive actions as it learns to navigate and accomplish various goals. For the agent to effectively use hierarchical reinforcement learning, it has to first *partition* the overall task into appropriate subtasks, and then learn how to accomplish subtasks in an appropriate sequence. A naive search in such combinatorial spaces can be computationally prohibitive. We focus on a continual learning agent that solves multiple related tasks in a given domain, and leverages prior experience to accomplish new tasks in an efficient manner. We propose using options (subtasks & associated policies) as a repository of knowledge that is transferred from prior tasks to the target task. While the problem of transfer, using options in particular, and HRL in general, has been explored in the past [2, 4, 5, 9], our proposed setting has not received much attention in the literature: efficient retrieval of *relevant options* for a new HRL task, from a large library of pre-learned options. This capability is crucial in sparse reward settings where the hierarchical policy can only afford to explore necessary parts of the search space; including even a few irrelevant options can significantly impact convergence rates or even task completion.

Our primary challenge is to determine relevant options for a particular task without first solving the task itself. Our key insight is that the state of the environment, and the actions enabled by items in the environment [3, 7], provide substantial information about what tasks are *achievable*. Based on this insight, we propose Option-Indexed Hierarchical Reinforcement Learning (OI-HRL), which solves new goals in a known domain by first fetching the relevant options from the option library, then constructing a hierarchical policy using the fetched options. Our proposal has three major ideas: (1) an option representation based on the frequent cooccurrence of option sets, (2) an affinity score between a given environment state and the options relevant to goals achievable in that environment, and (3) a meta-training loop, which learns this affinity score by solving a series of related HRL tasks in a given problem domain. We sketch the outline of our proposal & experiments here; for a more detailed treatment, please consult the extended version [1].

## 2 APPROACH

Consider a kitchen with food items, utensils, and appliances. An agent can accomplish goals such as picking up a bowl, picking up an egg, picking up a pan, cracking an egg, cooking on a stove, making an omelet, slicing apples, placing on a table, etc. We use the term *task* to refer to the accomplishment of a specific *end goal*. We distinguish tasks into two types: base tasks, which do not have any dependencies, and composite tasks, which depend upon earlier goals having been achieved first, in a specific order. We assume the agent has access to a set of pre-trained option policies $O = \{O_1, O_2, \ldots O_k\}$, one for completing each base task.

*Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.*

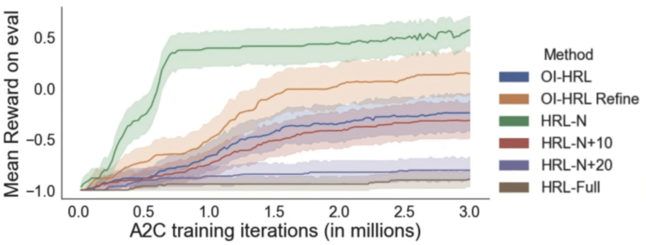

**Figure 1: Mean reward vs A2C training iterations.**

When an environment is initialized, a composite task $t_i$ is chosen, and the environment is filled with the necessary items (and a few random distractors). The chosen goal task is used to get a MDP $\mathcal{M}_{t_i} = (\mathcal{S}, \mathcal{A}, \mathcal{R}_{t_i}, \mathcal{P})$ where $\mathcal{S}$ and $\mathcal{A}$ are the set of states and actions, $\mathcal{R}_{t_i}$ and $\mathcal{P}$ are the reward and transition functions. Here the reward function is sparse, with the reward being non-zero only when the task $t_i$ is completed. When encountering a new task, our agent leverages its experience to retrieve the relevant options from the option library. Subsequently, the agent learns an HRL policy using only the retrieved options. If the retrieved set of options is sufficient for accomplishing the goal, the HRL policy (eg. learned by Advantage Actor Critic method [8]) will succeed in completing the goal. Furthermore, the fewer extraneous options retrieved, the faster the HRL policy can be learned.

For retrieving options, we learn an *affinity*/similarity measure between options and the initial state of the environment. Based on this affinity, we retrieve a subset of options from the base option set $\mathcal{B}$. We maintain an index $\Psi$ and a Query Generation Network (QGN) $\mathcal{N}$. The index $\Psi \subset \mathbb{R}^d \times O$ stores options $O$ together with a key vector $\psi(O_i) \in \mathbb{R}^d$ for option $O_i$, i.e. $\Psi = \{(\psi(O_i), O_i)\}$. The QGN generates a query $\mathbf{q} \in \mathbb{R}^d$ based on objects initially present in the environment ($s_0$). This query is transformed using $\hat{\mathbf{p}}_i = \text{Softmax}(\mathbf{q}^\intercal \psi(O_i))$ to produce a probabilistic ranking over options, from which the top-p options are retrieved [6].

**Meta-training:** We iterate over sampled tasks $t_i$. At each step an MDP $\mathcal{M}_{t_i}$ is created from the sampled task; the QGN and option indices are used to select a subset of options $\widehat{O}$ as described, alongside retrieval probabilities $\hat{\mathbf{p}}$ for all options. An HRL policy is learned over options $\widehat{O}$ using A2C [8], and used to sample a set of successful trajectories $\Lambda$. We calculate $\mathbf{y}$, the normalized frequencies of option usage in $\Lambda$, and update the retrieval networks based on $(\mathbf{y}, \hat{\mathbf{p}})$.

$$\mathcal{L}(\mathbf{y}, \mathbf{p}) = -\frac{1}{k}\sum_{i=1}^{k} \mathbf{y}_i \log(\mathbf{p}_i)$$

**Indexing & retrieval:** We propose two distinct ways of meta-learning the parameters of the QGN $\mathcal{N}$ and option embeddings $\psi(O_i)$– *(1) Pretrained Index:* We learn a co-occurrence-based representation for each option using option sequences that the agent encounters in the meta-training phase, and use them as the (fixed) option indices $\psi(O_i)$, *(2) Learned Index:* We meta-learn both the parameters of the QGN $\mathcal{N}$ and option embeddings $\psi(O_i)$ in an end-to-end manner. In the former approach, we update $\mathcal{N}$ alone; in the latter, we update both $(\mathcal{N}, \Psi)$, in the meta-training loop.

**Test-time adaptation:** In some cases, options selected using OI-HRL are not sufficient to complete the goal, and the HRL policy will necessarily fail to converge. However, the fetched set is still represents a potentially better starting point than learning a policy from scratch. In order to further improve performance in this scenario, the OI-HRL-Refine variant iteratively drops options that are infrequently used in successful (reward-finding) training runs, and, includes additional unchosen options from $O$ after a number of unsuccessful (incomplete) training runs, in the decreasing order of the selection probability predicted by the QGN. We see that this additional optimization further improves average reward rates, through increased task completion rates on the test set.

## 3 EXPERIMENTS

We experiment on a number of diverse but interrelated food preparation tasks in AI2THOR, an interactive 3D environment where an agent can navigate around and interact with a variety of objects. In order to study the effect of selecting a subset of options, we compare OI-HRL with the following baselines having varying number of options made available to the HRL learner.

- **HRL-N**: HRL using the exact set of options required to complete the task. This is an oracular upper bound on performance for OI-HRL (uses privilaged information).
- **HRL-N+K**: like HRL-N but with k extra (irrelevant) options.
- **HRL-Full**: using the entire library of options $O$.

Figure 1 shows the mean reward obtained on a test task distribution by the A2C policy as a function of training steps. As test tasks widely vary in difficulty, we see substantial variance in mean rewards, even for the oracular HRL-N skyline. Adding 10 irrelevant options (HRL-N+10) causes a significant drop in asymptotic performance, and subdued growth over the first 1M iterations. Adding 20 extra options (HRL-N+20) results in little to no learning, and is close to the HRL-Full baseline. OI-HRL matches or exceeds the sample efficiency of HRL-N+10 for the first 1M iterations and achieves asymptotic performance comparable to HRL-N+10. Note that these baselines are oracular in nature and make use of privileged information, albeit with some added noise; in particular, task completion and reward is theoretically guaranteed using these baselines. Despite the absence of guarantees, OI-HRL remains competitive with these baselines. OI-HRL Refine improves the performance of OI-HRL further, significantly covering the performance gap between OI-HRL and the oracular HRL-N.

## 4 CONCLUSION

We presented OI-HRL, an option indexing approach towards large-scale reuse of learned options in HRL, where only a small fraction of options are relevant to the task at hand. We proposed a method by which we can infer option relevance based on the state of the environment and showed that these relevance or affinity scores could be effectively learned over a distribution of tasks in a meta-training framework. Our results show an exponential improvement in average reward rates for OI-HRL compared to baselines that include all available options, and relevant options plus some fixed number of irrelevant options. While we do not address the question of where the library of options comes from, it is natural to expect a continual learning agent will acquire such a library of skills over the course of its lifetime.

## REFERENCES

[1] Kushal Chauhan, Soumya Chatterjee, Akash Reddy, Balaraman Ravindran, and Pradeep Shenoy. 2022. Matching options to tasks using Option-Indexed Hierarchical Reinforcement Learning. https://doi.org/10.48550/ARXIV.2206.05750

[2] Carlos Florensa, Yan Duan, and Pieter Abbeel. 2017. Stochastic neural networks for hierarchical reinforcement learning. *ICLR* (2017).

[3] James J Gibson. 1977. The theory of affordances. *Hilldale, USA* 1, 2 (1977), 67–82.

[4] Karol Hausman, Jost Tobias Springenberg, Ziyu Wang, Nicolas Heess, and Martin Riedmiller. 2018. Learning an embedding space for transferable robot skills. In *ICLR*.

[5] Nicolas Heess, Greg Wayne, Yuval Tassa, Timothy Lillicrap, Martin Riedmiller, and David Silver. 2016. Learning and transfer of modulated locomotor controllers. *arXiv preprint arXiv:1610.05182* (2016).

[6] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*. https://openreview.net/forum?id=rygGQyrFvH

[7] Khimya Khetarpal, Zafarali Ahmed, Gheorghe Comanici, David Abel, and Doina Precup. 2020. What can I do here? A Theory of Affordances in Reinforcement Learning. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 5243–5253. https://proceedings.mlr.press/v119/khetarpal20a.html

[8] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous Methods for Deep Reinforcement Learning. In *Proceedings of The 33rd International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 48)*, Maria Florina Balcan and Kilian Q. Weinberger (Eds.). PMLR, New York, New York, USA, 1928–1937. https://proceedings.mlr.press/v48/mniha16.html

[9] Tianmin Shu, Caiming Xiong, and Richard Socher. 2018. Hierarchical and interpretable skill acquisition in multi-task reinforcement learning. *ICLR* (2018).