An Empowerment-based Solution to Robotic Manipulation Tasks with Sparse Rewards

Anonymous Author(s)
Affiliation
Address
email

Abstract:
In order to provide adaptive and user-friendly solutions to robotic manipulation, it is important that the agent can learn to accomplish tasks even if they are only provided with very sparse instruction signals. To address the issues reinforcement learning algorithms face when task rewards are sparse, this paper proposes a novel form of intrinsic motivation that can allow robotic manipulators to learn useful manipulation skills with only sparse extrinsic rewards. Through integrating and balancing empowerment and curiosity, this approach shows superior performance compared to other existing intrinsic exploration approaches during extensive empirical testing. Qualitative analysis also shows that when combined with diversity-driven intrinsic motivations, this approach can help manipulators learn a set of diverse skills which could potentially be applied to other more complicated manipulation tasks and accelerate their learning process.

Keywords: Reinforcement Learning, Robotic Manipulation, Intrinsic Motivation, Empowerment

1 Introduction

Real-world robotic manipulation tasks are diverse yet often complicated. An ideal robotic agent should be able to adapt to new environments and learn new tasks by exploring on its own, instead of requiring intensive human supervision. The traditional task and motion planning approach to robotic manipulation [1] typically requires a significant amount of domain-specific prior knowledge, and acquiring this knowledge often involves intensive human engineering. On the other hand, reinforcement learning (RL) agents have demonstrated impressive performances in scenarios with well-structured environment and dense reward signals [2, 3]. However, learning-based approaches to manipulation typically only work well when the reward function is dense or when expert demonstrations are available. This is because when the state and action space is high-dimensional and the reward signal is sparse, RL agents could potentially spend a long time exploring the state space without getting any reward signal. Therefore, RL has seen less success in tasks with unstructured environments like robotic manipulation where the dynamics and task rewards are less intuitive to model. Designing task-specific dense reward functions to simplify the sparse-reward RL problem has been a common solution for manipulation problems, but in most practical applications, hand designing dense reward functions for every robot in every task and every environment is infeasible and might bias the agent's behavior in a suboptimal way [4]. Inverse reinforcement learning approaches seek to automate reward definition by learning a reward function from expert demonstrations, but inevitably demand a significant amount of task-specific knowledge and place considerable expert data collection burden on the user [5]. Recent advances in meta-learning allow agents to transfer learned skills to other similar tasks [6, 7], but a large amount of prior meta-training data across a diverse set of tasks is required, which also becomes a burden if a lot of human intervention is needed. Therefore, effectively solving sparse reward problems from scratch is an important capability that will allow RL agents to be applied in practical robotic manipulation tasks.

In this paper, we propose an empowerment-based intrinsic exploration approach that allows robots to learn manipulation skills with only sparse extrinsic rewards from the environment. Empowerment is

Submitted to the 4th Conference on Robot Learning (CoRL 2020). Do not distribute.

an information-theoretic concept proposed in an attempt to find local and universal utility functions which help individuals survive in evolution by smoothening the fitness landscape [8]. Through measuring the mutual dependence between actions and states, empowerment indicates how confident the agent is about the effect of its actions in the environment. In contrast to novelty-driven intrinsic motivations which encourage the agent to explore actions with unknown effects, empowerment emphasizes the agent's "controllability" over the environment and favors actions with predictable consequences. We hypothesize that empowerment is a more suitable form of intrinsic motivation for robotic manipulation tasks where the desired interactions with environment objects are typically predictable and principled. Imagine a robot interacting with a box on the table. Intuitively, the undesirable behaviors of knocking the box onto the floor should generate higher novelty since it helps explore more states that haven't been visited, and the desirable behaviors of pushing the box or lifting the box up should generate higher empowerment because the effects of these actions are more predictable. Based on this intuition, we apply an empowerment-based intrinsic motivation to manipulation tasks with sparse extrinsic rewards and demonstrate that with the help of novelty-driven rewards at the beginning of training, neural function approximators can provide reasonable estimations of empowerment values. With extensive empirical testing on object-lifting and pick-and-place tasks in simulation environments, we show that this empowerment-based approach outperforms other state-of-the-art intrinsic exploration methods when the extrinsic task rewards are sparse. Although the concept of empowerment has previously been discussed in the context of RL [9], to the author's best knowledge, this paper is the first successful demonstration of the effectiveness of empowerment in terms of assisting RL agents in learning complicated robotics tasks with sparse rewards.

2 Related Work

Reinforcement learning for sparse reward tasks has been been extensively studied from many different perspectives. Curriculum learning [10] is a continuation method that starts training with easier tasks and gradually increases task difficulty in order to accelerate the learning progress. However, many curriculum-based methods only involve a small and discrete set of manually generated task sequences as the curriculum, and the automated curriculum generating methods often assume known goal states or prior knowledge on how to manipulate the environment [11, 12] and bias the exploration to a small subset of the tasks [13]. Through implicitly designing a form of curriculum to first achieve easily attainable goals and then progress towards more difficult goals, Hindsight Experience Replay (HER) is the first work that allows complicated manipulation behaviors to be learned from scratch with only binary rewards [4]. However, when the actual task goal is very distinct from what random policies can achieve, HER's effect is limited. As mentioned in [4], HER is unable to allow manipulators to learn pick-and-place tasks without using demonstration states during training.

Hierarchical reinforcement learning (HRL) approaches utilize temporal abstraction [14] or information asymmetry [15, 16] to introduce inductive biases for learning complicated tasks and transferable skills. Frameworks that combine multiple different tasks through a high level task selection policy [5, 17] have also shown effectiveness for learning sparse reward tasks. Intrinsic exploration approaches, instead, augments the reward signals by adding task-agnostic rewards which encourage the agent to explore novel or uncertain states [18]. Many approaches in the theme of intrinsic exploration have been proposed to alleviate the burden of reward engineering when training RL agents: visit counts and pseudo-counts [19] encourage the agent to explore states that are less visited; novelty-based approaches [20, 21] motivate the agent to conduct actions that lead to more uncertain results; reachability-based approaches [22] gives rewards to the observations outside of the explored states that take more environment steps to reach; diversity-driven approaches [23, 24] learn skills using a maximum entropy policy to allow for the unsupervised emergence of diverse skills; and information gain [9, 25, 26] encourages the agent to explore states that will improve its belief about the dynamics. However, count-based and uncertainty-based exploration methods often can't distinguish between task-irrelevant distractions and task-related novelties, and the high computational complexity largely restricts the application of existing information-theoretic methods in practical robotic manipulation tasks. The approach proposed in this paper falls under the category of information-theoretic intrinsic exploration, and we provide insight into reasonable approximations that can make the computation of information-theoretic quantities feasible when the state and action spaces are continuous and high-dimensional with complex mutual dependencies. Extensive experiment results demonstrate the effectiveness of these approximations as well as the superiority of the proposed approach over existing intrinsic exploration approaches in robotic manipulation scenarios.

3 Preliminaries

3.1 Mutual Information

Definition Mutual information (MI) is a fundamental quantity for measuring the mutual dependence between random variables. It quantifies the amount of information obtained about one random variable through observing the other. For a pair of continuous variables X and Y, MI is defined as:

$$\mathcal{I}(X;Y) = \iint p_{XY}(x,y) \log \frac{p_{XY}(x,y)}{p_X(x)p_Y(y)}\, dx\, dy = \mathbb{E}_{XY}\Big[\log \frac{p_{XY}}{p_X p_Y}\Big], \tag{1}$$

where $p_X(x)$ and $p_Y(y)$ are the marginal probability density functions for X and Y respectively, and $p_{XY}(x,y)$ is the joint probability density function. MI is also often expressed in terms of Shannon entropy [27] as well as Kullback-Leibler (KL) divergence:

$$\mathcal{I}(X;Y) = \mathcal{H}(X) - \mathcal{H}(X|Y) = \mathcal{H}(Y) - \mathcal{H}(Y|X) = D_{KL}(p_{XY}||p_X p_Y), \tag{2}$$

where $\mathcal{H}(X)$ and $\mathcal{H}(Y)$ are the marginal entropies, $\mathcal{H}(X|Y)$ and $\mathcal{H}(Y|X)$ are conditional entropies, $\mathcal{H}(X,Y)$ is the joint entropy, and $D_{KL}(p_{XY}||p_X p_Y)$ denotes the KL-divergence between the joint distribution and the product of the marginal distributions.

Conditional MI measures the mutual dependency between two random variables conditioned on another random variable. For continuous variables X, Y and Z, conditioned MI can be written as:

$$\mathcal{I}(X;Y|Z) = \iiint \log\Big(\frac{p_{X|Y,Z}(x|y,z)}{p_{X|Z}(x|z)}\Big) p_{X,Y,Z}(x,y,z)\, dx\, dy\, dz, \tag{3}$$

where $p_{X,Y,Z}(x,y,z)$ is the joint probability density function, and $p_{X,Y|Z}(x,y|z)$, $p_{X|Y,Z}(x|y,z)$, $p_{Y|X,Z}(y|x,z)$, $p_{X|Z}(x|z)$ and $p_{Y|Z}(y|z)$ are conditional probability density functions.

Computation In general, the computation of MI is intractable. Exact computation of MI is only tractable for discrete random variables and a limited family of problems with known probability distributions [28]. Traditional algorithms for MI maximization, e.g. the Blahut-Arimoto algorithm [29], don't scale well to realistic problems because they typically rely on enumeration. Therefore, researchers often maximize a lower bound of MI instead of computing its exact value.

The variational lower bound derived from the non-negativity of KL-divergence, shown in Equation 4, is one of the most commonly used lower bounds for MI in the RL community:

$$\begin{aligned}\mathcal{I}(X;Y) &= \mathbb{E}_{XY}\Big[\log \frac{p(x|y)\cdot q(x|y)}{p(x)\cdot q(x|y)}\Big] = \mathbb{E}_{XY}\Big[\log \frac{q(x|y)}{p(x)}\Big] + \mathbb{E}_{XY}\Big[\log \frac{p(x|y)}{q(x|y)}\Big] \\ &= \mathbb{E}_{XY}\Big[\log \frac{q(x|y)}{p(x)}\Big] + \mathbb{E}_{Y}\Big[D_{KL}(p(x|y)||q(x|y))\Big] \geq \mathbb{E}_{XY}\Big[\log \frac{q(x|y)}{p(x)}\Big].\end{aligned} \tag{4}$$

Other variational lower bounds of MI have also been derived based on a broader class of distance measures called f-divergence [30, 31, 32]. KL-divergence and Jensen-Shannon (JS) divergence are two special cases of f-divergence. Based on the relationship between MI and KL-divergence shown in Equation 2, a lower bound of MI is derived in [28]:

$$\mathcal{I}_{KL}(X;Y) \geq \sup_{T\in\mathcal{T}} \mathbb{E}_{p_{XY}}[T] - \mathbb{E}_{p_X p_Y}[e^{T-1}]. \tag{5}$$

The JS definition of MI is closely related to the MI we defined in Equation 1, and its lower bound can be derived as [26]:

$$\begin{aligned}\mathcal{I}_{JS}(X;Y) &= D_{JS}(p_{XY}||p_X p_Y) \\ &\geq \sup \mathbb{E}_{p_{XY}}[\log 2 - \log(1+e^{-T})] - \mathbb{E}_{p_X p_Y}[D^*_{JS}(\log 2 - \log(1+e^{-T}))] \\ &= \sup \mathbb{E}_{p_{XY}}[-\mathrm{sp}(-T)] - \mathbb{E}_{p_X p_Y}[\mathrm{sp}(T)] + \log 4,\end{aligned} \tag{6}$$

where $D^*_{JS}(u) = -\log(2-\exp(u))$ is the Fenchel conjugate of JS-divergence, and $\text{sp}(u) = \log(1+\exp(u))$ is the soft plus function. Note that Equation 6 is not a lower bound for the MI we defined in Equation 1, but since the two MIs are closely related, it is also often used to estimate the MI defined in Equation 1. In this paper, we refer to the variational lower bound in Equation 4 as VLB, the lower bound based on KL-divergence in Equation 5 as KLD, and the lower bound for JS-divergence based mutual information in Equation 6 as JSD.

3.2 Empowerment

Empowerment is an information-theoretic quantity that measures the value of the information an agent obtains in the action-observation sequences it experiences during the reinforcement learning process [9]. It is defined as the maximum mutual information between a sequence of K actions $\mathbf{a}$ and the final state $\mathbf{s}'$, conditioned on a starting state $\mathbf{s}$:

$$\mathcal{E}(\mathbf{s}) = \max_{\pi} \mathcal{I}^{\pi}(\mathbf{a}, \mathbf{s}'|\mathbf{s}) = \max_{\pi} \mathbb{E}_{p(s'|a,s)\pi(a|s)} \left[\log \left(\frac{p(\mathbf{a}, \mathbf{s}'|\mathbf{s})}{\pi(\mathbf{a}|\mathbf{s})p(\mathbf{s}'|\mathbf{s})} \right) \right], \tag{7}$$

where $\mathbf{a} = \{a_1, \ldots, a_K\}$ is a sequence of K primitive actions leading to a final state $\mathbf{s}'$, $\pi(\mathbf{a}|\mathbf{s})$ is exploration policy over the K-step action sequences, $p(\mathbf{s}'|\mathbf{a}, \mathbf{s})$ is the K-step transition probability of the environment, $p(\mathbf{a}, \mathbf{s}'|\mathbf{s})$ is the joint distribution of actions sequences and the final state conditioned on the initial state $\mathbf{s}$, and $p(\mathbf{s}'|\mathbf{s})$ is the marginalized probability over the action sequence.

3.3 Intrinsic Curiosity Module

Intrinsic Curiosity Module (ICM) [20] is one of the state-of-the-art novelty-driven intrinsic exploration approaches that aims at learning new skills by performing actions whose consequences are hard to predict. It trains an inverse model g to learn a feature encoding ϕ that captures the parts of the state space related to the consequences of the agent's actions, so that the agent will focus on the relevant part of the environment and not get distracted by other details in the camera observations. It also learns the forward model f and uses the prediction error of the forward model as the intrinsic reward in order to facilitate the agent to explore the part of the state space where it can't predict the consequences of its own actions very well.

$$\text{Inverse Model: } \hat{a}_t = g(\phi(s_t), \phi(s_{t+1})); \quad \text{Forward Model: } \hat{\phi}(s_{t+1}) = f(\phi(s_t), a_t). \tag{8}$$

4 Approach: Empowerment-based Intrinsic Motivation

We hypothesize that empowerment would be a good candidate for augmenting the sparse extrinsic rewards in manipulation tasks because it indicates the amount of information contained in the action sequence $\mathbf{a}$ about the future state $\mathbf{s}'$. Through maximizing empowerment, we are effectively encouraging the agent to influence the environment in a predictable way, which is the desired behavior in most manipulation tasks. However, as a form of conditional MI for continuous variables, the computation of empowerment is especially challenging. This is because for conditional MI $\mathcal{I}(X;Y|Z)$ with continuous Z, estimating $\mathcal{I}(X;Y|Z)$ for all Z is approximately equivalent to estimating an infinite number of unconditional MIs. In this section, we discuss the approaches we take to make empowerment a feasible form of intrinsic motivation in practical robotic manipulation tasks.

4.1 Approximations to Simplify Empowerment Calculation

Mohamed and Rezende [9] suggest that the empowerment at each state in the state space can be calculated using an exploration policy $\pi(\mathbf{a}|\mathbf{s})$ that generates an open-loop sequence of K actions into the future (Equation 7), so that a closed-loop policy can be obtained by a planning algorithm using the calculated empowerment values. Although Mohamed and Rezende demonstrated the effectiveness of this approach in grid world environments, it is infeasible to precompute the empowerment values for all states in a high-dimensional, continuous state space. Therefore, we make a few approximations in order to make empowerment-based intrinsic motivation a practical approach. First, we use only one action step instead of an action sequence to estimate empowerment. Second, instead of constructing a separate exploration policy π to first compute empowerment and then plan

a closed-loop policy according to empowerment, we directly optimize the behavior policy ω using empowerment as an intrinsic reward in an RL algorithm. These two approximations mean that the agent will only be looking at the one-step reachable neighborhood of its current state to find the policy that leads to high mutual information. Despite sacrificing global optimality, this approach prioritizes the policy that controls the environment in a principled way so that more extrinsic task rewards can be obtained compared to using random exploration, which help resolve the fundamental issue in sparse reward tasks.

In addition to the above two approximations, it is also important to note that in robotic manipulation tasks, we are typically not interested in the mutual dependence between robot action and robot states, and we wish to avoid the robot trivially maximizing empowerment through motion of its own body. Therefore, we assume that the state space can be divided into intrinsic states $\mathbf{s}^{in}$ (robot states) and extrinsic states $\mathbf{s}^{ex}$ (environment states), and only extrinsic states are used as $\mathbf{s}'$ when calculating empowerment. Namely, the empowerment used in this paper is defined as:

$$\mathcal{E}(\mathbf{s}_t) \approx \mathcal{I}^{\omega}(\mathbf{a}_t, \mathbf{s}^{ex}_{t+1}|\mathbf{s}_t) = \mathcal{H}^{\omega}(\mathbf{a}_t|\mathbf{s}_t) - \mathcal{H}^{\omega}(\mathbf{a}_t|\mathbf{s}^{ex}_{t+1}, \mathbf{s}_t), \tag{9}$$

where ω is the behavior policy, and the relationship to Shannon Entropy is derived from Equation 2.

4.2 Maximizing Empowerment using Mutual Information Lower Bounds

Neural function approximators have become powerful tools for numerically estimating conditional MIs for continuous random variables [9, 28, 26]. However, in most RL scenarios, since exact distributions are typically unavailable and numerical estimation through sampling is required, computation of high-dimensional conditional MI remain challenging. As mentioned in Section 3.1, a common practice is to maximize a lower bound of MI instead of its exact value. We test the performance of the three MI lower bounds introduced in Section 3.1 on distributions with know conditional MI and provide detailed experiment results Appendix A. We conclude that, in terms of estimating the conditional MI of the continuous random variables we tested on, VLB performs the best in all cases and KLD performs the worst in most cases. However, same conclusion may not be drawn for high-dimensional distributions with complex mutual dependencies. In the manipulation tasks in this paper, we noticed that JSD is the best performer on Fetch and VLB is the best performer on PR2, hence we will only report the results with the corresponding best performer in each environment.

4.3 Combination with ICM to Facilitate Empowerment Computation

Another challenging issue with empowerment-based RL is that well-balanced data are not easy to obtain at the beginning of training. If we initialize the RL agent with a random policy, it will highly likely explore much more of the empty space than regions with object interactions because the interaction-free part of the state space is often much larger. However, since $\mathbf{a}_t$ and $\mathbf{s}^{ex}_{t+1}$ are independent without interactions, the training data fed into the empowerment estimation network will be strongly biased towards the zero empowerment regions, which makes it very difficult to train accurate estimation models. Therefore, it is crucial that enough training data in the interacting part of the state space can be obtained at the beginning of training in order to get accurate estimations of empowerment. We achieve this through combining empowerment with ICM using adaptive coefficients, which initially place more weight on ICM to ensure enough well-balanced data are fed to the empowerment estimation networks, and then switches more weight to empowerment to encourage the robot to learn controllable behaviors. Figure 1 summarizes the proposed empowerment-based intrinsic motivation approach, and Appendix B elaborates on the algorithm implementation details.

5 Empirical Evaluation

5.1 Environment Setup

In order to compare the performance of the empowerment-based intrinsic motivation with other state-of-the-art intrinsic motivations, we created four object-lifting tasks with different object shapes in OpenAI Gym [33] and Gazebo, as shown in Figure 2. The Gym environment uses a Fetch robot with a 25D state space and a 4D action space, and the Gazebo environment uses a PR2 robot with a 38D state space and an 8D action space. We also use the FetchPickAndPlace-V1 task provided

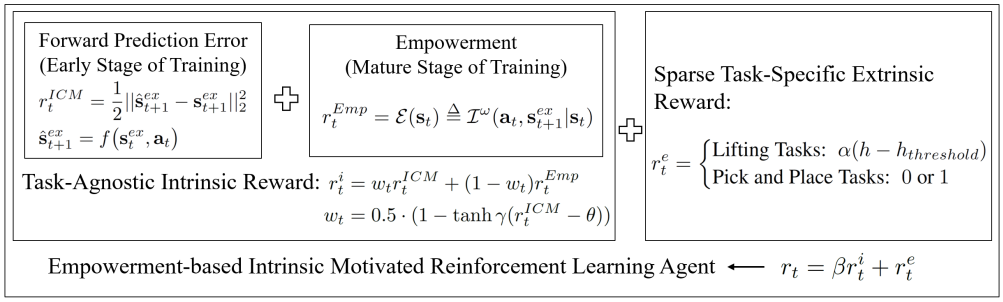
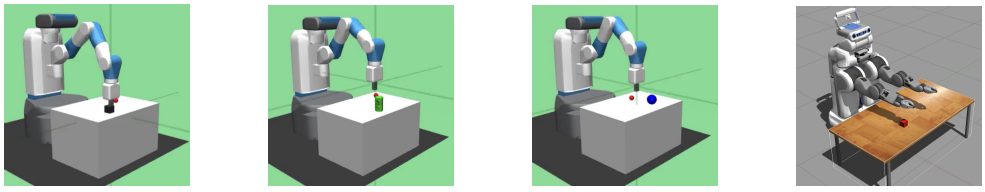


Figure 1: Overview of the empowerment-based intrinsic motivation approach



(a) Fetch with a box (b) Fetch with a cylinder (c) Fetch with a sphere (d) PR2 with a box

Figure 2: Simulation environments

in Gym in order to compare with HER because HER requires a goal-conditioned environment. In the four object-lifting tasks, the goal is to lift up the object, and the extrinsic reward is only given when the object's height is above a threshold. In the pick-and-place task, the reward is given when the distance of the object to the goal pose is within a threshold. We use Proximal Policy Optimization (PPO) [34] as the RL agent for all experiments. Experiments on the Fetch robot use 60 parallel environments for training, and PR2 experiments use 40 due to its higher CPU requirement. Implementation details including hyperparameters and task rewards are provided in Appendix B.

5.2 Experiment Results

In this section, we provide experiment results that compare the proposed empowerment-based intrinsic motivation approach with other state-of-the-art algorithms, including ICM [20], exploration via disagreement [21] (referred to as Disagreement in this paper) and HER [4]. We use our implementation of ICM and Disagreement, and use the OpenAI Baselines implementation [35] for HER. In both ICM and Disagreement, we also make the same assumption as in the empowerment implementation that the state space can be divided into intrinsic states and extrinsic states, and only the prediction error or variance of the extrinsic states contribute to the intrinsic rewards. We run HER with 2 MPI processes with 30 parallel environments each to make sure it is equivalent to the 60 parallel environments in other experiments. Other parameters for HER are set to default. All the results in the Fetch environment are averaged over 10 different random seeds, and the results in the PR2 environment are averaged over 2 random seeds due to limited computation resources.

Figure 3(a)-(c) compare the performance of our approach with ICM, Disagreement, and PPO without any intrinsic reward in the object-lifting tasks with a Fetch robot, and Figure 3(d) compares our approach with ICM and Disagreement in box-lifting tasks with a PR2 robot. In the Fetch environment, the cylinder lifting task is much more difficult compared to box lifting and sphere lifting, thus we use a larger scale α for extrinsic lifting reward. Similarly, we also use a larger α for the box-lifting task with the PR2 robot since this environment is much higher-dimensional and hence more difficult for an RL agent. From Figure 3(a)-(c) we can see that the reward curve for PPO without any intrinsic reward remains almost zero, which proves that sparse reward tasks are very challenging for vanilla RL algorithms. In all four environments, our empowerment-based approach is able to help the robot achieve higher lifting rewards faster than other approaches we compared with. The Disagreement approach is able to perform better in the box lifting task with the Fetch robot after training for a long time, but it performs much worse than the other two intrinsic motivations in the cylinder and sphere lifting tasks. Another finding from Figure 3(a)-(c) is that the advantage of the

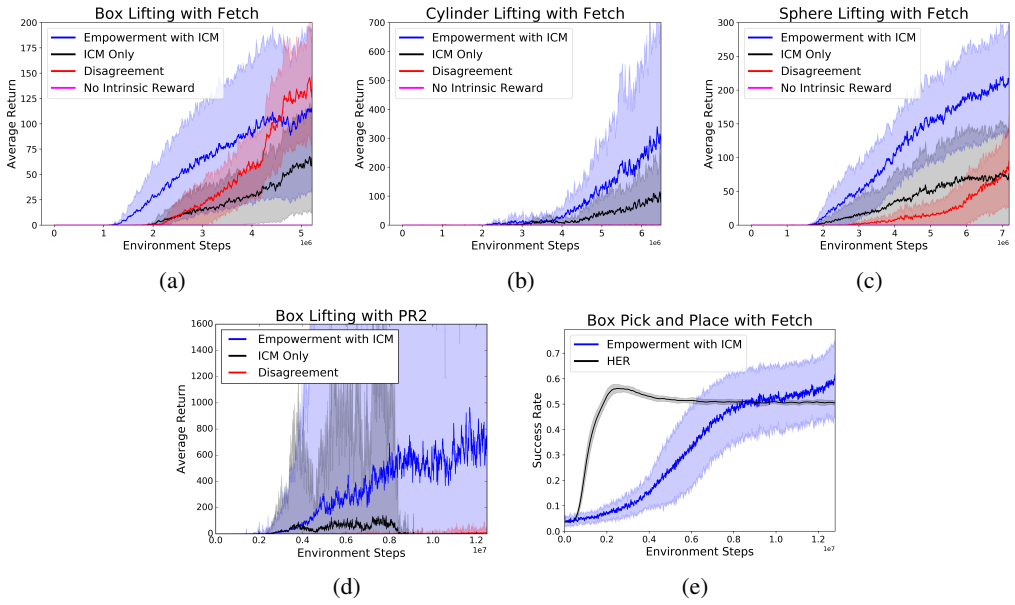


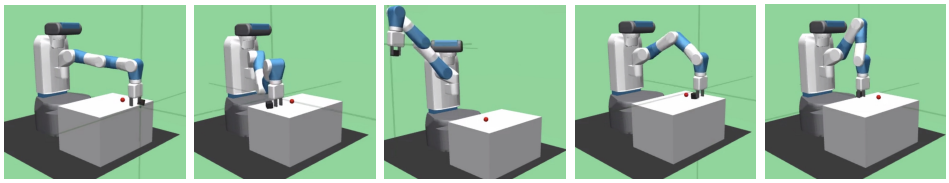
Figure 3: Experiment results. (a)-(d) compare the performance of the proposed empowerment-based approach (referred to as empowerment with ICM since ICM is used to help training the empowerment prediction networks) with ICM and Disagreement in object lifting tasks, and (e) compares the proposed empowerment-based approach with HER in pick-and-place tasks. The solid lines represent the mean and the shadow areas represent the 95% confidence intervals.

empowerment-based intrinsic motivation is much more obvious in the cylinder and sphere lifting tasks compared to the box lifting tasks. We hypothesize that this is because the ability of "controlling" the object is much more important when there are round surfaces, since these objects are more difficult to pick up and also more likely to randomly roll around when novelty is the only intrinsic motivation. In fact, in the cylinder lifting task, our empowerment-based intrinsic motivation is the only approach that allows the agent to learn to directly pick up the cylinder from the top without knocking it down first, whereas agents trained with ICM will knock down the cylinder and then pick up radially. In Figure 3(d), although the confidence intervals are wider due to the smaller number of runs, we can still get the similar conclusion that our approach shows the best performance.

Figure 3(e) compares the empowerment-based intrinsic motivation with HER in the Fetch pick-and-place environment. We can see that the average success rate of HER goes up much faster than the empowerment approach, but it stays at about 0.5 even after a long time of training. In fact, none of the 10 runs of HER has reached a success rate of 0.6 or above. In contrast, although the empowerment approach is slower in the initial learning phase, in 3 out of 10 runs it has learned to lift up the object and reach the goals in the air accurately and quickly, and the success rate stays at about 1 in these tests. This is because in the Gym FetchPickAndPlace-V1 task, half of the goals are sampled from on the table and half are sampled in the air, thus agents that only learned to push can still reach the goals close to the tabletop and receive a success rate of about 0.5, but only agents that actually learned to pick and place will reach a success rate of 1.0.

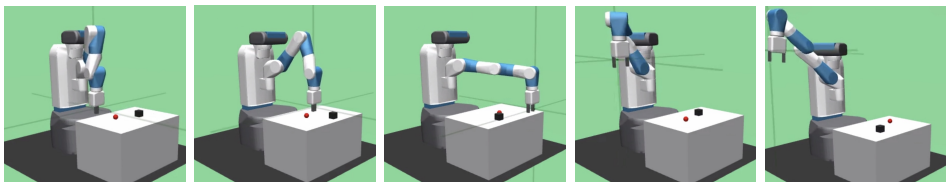
6 Application: Learning a Diverse Set of Skills

Besides its advantage in solving sparse reward RL tasks, another driving force for research on intrinsic motivation is its potential in unsupervised skill discovery. Many HRL frameworks allow RL agents to learn policies of different levels so that high-level policies only need to focus on the skill-space that low-level controllers provide instead of the raw state-space. However, the skills an end-to-end HRL system can learn are limited and they often require guidance from human-designed "curricula" [14, 5, 17]. In contrast, skills discovered by intrinsic motivations can reduce HRL frameworks' dependence on human engineering and potentially enable them to learn more complicated



(a) (b) (c) (d) (e)

Figure 4: Qualitative performance of the proposed empowerment-based intrinsic motivation when combined with the diversity-driven DIAYN [23] approach in the box lifting task with a Fetch robot. (a)-(e) show the different skills learned when the number of skills in DIAYN is set to 5.



(a) (b) (c) (d) (e)

Figure 5: Different skills learned with DIAYN [23] without the empowerment-based intrinsic motivation in the box lifting task with a Fetch robot when the number of skills is set to 5.

tasks. Ultimately, we hope the empowerment-based intrinsic motivation proposed in this paper can also be incorporated into a HRL framework and contribute to the learning of complicated manipulation skills, such as opening a container and stacking objects inside. In order to see what type of skills an agent can learn with our approach, we provide preliminary qualitative results combining empowerment and the Diversity is All You Need (DIAYN) approach [23] in the "Fetch with a box" environment. Figure 4 and 5 compare the skills learned by combining empowerment and DIAYN as the intrinsic reward and the skills learned with only DIAYN as the intrinsic reward. From Figure 5 we can see that without an intrinsic motivation that drives the agent to control the object, the skills learned through a purely diversity-driven approach are not meaningful in terms of solving manipulation tasks because they don't involve interactions with the object. In comparison, Figure 4 demonstrates the potential of this combined intrinsic reward in terms of learning a set of meaningful manipulation skills, including pushing the object to different directions and lifting the object up.

7 Discussion

In this paper we present a novel intrinsic motivation for robotic manipulation tasks with sparse extrinsic rewards that leverages recent advances in both mutual information maximization and intrinsic novelty-driven exploration. Through maximizing the mutual dependence between robot actions and environment states, namely the empowerment, this intrinsic motivation helps the agent to focus more on the states where it can effectively "control" the environment instead of the parts where its actions cause random and unpredictable consequences. Despite the challenges posed by conditional mutual information maximization with continuous high-dimensional random variables, we are able to successfully train neural networks that make reasonable predictions on empowerment with the help of novelty-driven exploration methods at the beginning of the learning process. Empirical evaluations in different robotic manipulation environments with different shapes of the target object demonstrate the advantages of this empowerment-based intrinsic motivation over other state-of-the-art solutions to sparse-reward RL tasks. In addition, we also combine this approach with diversity-driven intrinsic motivation and show that the combination is able to encourage the manipulator to learn a diverse set of ways to interact with the object, whereas with the diversity-driven rewards alone the manipulator is only able to learn how to move itself in different directions. In future work, we hope to apply this empowerment-based intrinsic motivation in a HRL framework that can utilize it to learn a diverse yet meaningful set of manipulation skills, so that the HRL agent can ultimately accomplish more complicated tasks that existing approaches can't learn from scratch without reward shaping or demonstrations, such as opening a container and stacking objects inside.

References

[1] L. P. Kaelbling and T. Lozano-Pérez. Integrated task and motion planning in belief space. *The International Journal of Robotics Research*, 32(9-10):1194–1227, 2013.

[2] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In *2018 International Conference on Intelligent Robots and Systems (IROS)*, pages 4238–4245.

[3] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning*, pages 651–673, 2018.

[4] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. P. Abbeel, and W. Zaremba. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, pages 5048–5058, 2017.

[5] M. Riedmiller, R. Hafner, T. Lampe, M. Neunert, J. Degrave, T. Wiele, V. Mnih, N. Heess, and J. T. Springenberg. Learning by playing solving sparse reward tasks from scratch. In *International Conference on Machine Learning*, pages 4341–4350, 2018.

[6] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135, 2017.

[7] R. Chitnis, L. P. Kaelbling, and T. Lozano-Pérez. Learning quickly to plan quickly using modular meta-learning. In *2019 International Conference on Robotics and Automation*.

[8] A. S. Klyubin, D. Polani, and C. L. Nehaniv. Empowerment: A universal agent-centric measure of control. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 128–135.

[9] S. Mohamed and D. J. Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, 2015.

[10] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[11] C. Florensa, D. Held, M. Wulfmeier, M. Zhang, and P. Abbeel. Reverse curriculum generation for reinforcement learning. In *Conference on Robot Learning*, pages 482–495, 2017.

[12] R. Wang, J. Lehman, J. Clune, and K. O. Stanley. Paired open-ended trailblazer (poet): Endlessly generating increasingly complex and diverse learning environments and their solutions. *arXiv preprint arXiv:1901.01753*, 2019.

[13] S. Sukhbaatar, Z. Lin, I. Kostrikov, G. Synnaeve, A. Szlam, and R. Fergus. Intrinsic motivation and automatic curricula via asymmetric self-play. In *International Conference on Learning Representations*, 2018.

[14] P.-L. Bacon, J. Harb, and D. Precup. The option-critic architecture. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[15] A. Galashov, S. Jayakumar, L. Hasenclever, D. Tirumala, J. Schwarz, G. Desjardins, W. M. Czarnecki, Y. W. Teh, R. Pascanu, and N. Heess. Information asymmetry in KL-regularized RL. In *International Conference on Learning Representations*, 2019.

[16] A. Goyal, R. Islam, D. Strouse, Z. Ahmed, H. Larochelle, M. Botvinick, S. Levine, and Y. Bengio. Transfer and exploration via the information bottleneck. In *International Conference on Learning Representations*, 2019.

[17] C. Colas, P.-Y. Oudeyer, O. Sigaud, P. Fournier, and M. Chetouani. Curious: Intrinsically motivated modular multi-goal reinforcement learning. In *International Conference on Machine Learning*, pages 1331–1340, 2019.

[18] Y. Kim, W. Nam, H. Kim, J.-H. Kim, and G. Kim. Curiosity-bottleneck: Exploration by distilling task-specific novelty. In *International Conference on Machine Learning*, pages 3379–3388, 2019.

[19] H. Tang, R. Houthooft, D. Foote, A. Stooke, O. X. Chen, Y. Duan, J. Schulman, F. DeTurck, and P. Abbeel. # exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in neural information processing systems*, pages 2753–2762, 2017.

[20] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, pages 2778–2787, 2017.

[21] D. Pathak, D. Gandhi, and A. Gupta. Self-supervised exploration via disagreement. *arXiv preprint arXiv:1906.04161*, 2019.

[22] N. Savinov, A. Raichuk, D. Vincent, R. Marinier, M. Pollefeys, T. Lillicrap, and S. Gelly. Episodic curiosity through reachability. In *International Conference on Learning Representations*, 2019.

[23] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2019.

[24] A. Sharma, S. Gu, S. Levine, V. Kumar, and K. Hausman. Dynamics-aware unsupervised skill discovery. In *International Conference on Learning Representations*, 2020.

[25] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. De Turck, and P. Abbeel. Vime: Variational information maximizing exploration. In *Advances in Neural Information Processing Systems*, pages 1109–1117, 2016.

[26] H. Kim, J. Kim, Y. Jeong, S. Levine, and H. O. Song. Emi: Exploration with mutual information. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3360–3369, 2019.

[27] L. Paninski. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003.

[28] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm. Mutual information neural estimation. In *International Conference on Machine Learning*, pages 531–540, 2018.

[29] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[30] F. Liese and I. Vajda. On divergences and informations in statistics and information theory. *IEEE Transactions on Information Theory*, 52(10):4394–4412, 2006.

[31] X. Nguyen, M. J. Wainwright, and M. I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56 (11):5847–5861, 2010.

[32] S. Nowozin, B. Cseke, and R. Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pages 271–279, 2016.

[33] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym, 2016.

[34] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[35] P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, S. Sidor, Y. Wu, and P. Zhokhov. Openai baselines. https://github.com/openai/baselines, 2017.

[36] I. Gel'Fand and A. Yaglom. Calculation of amount of information about a random function contained in another such function. *Eleven Papers on Analysis, Probability and Topology*, 12: 199, 1959.

[37] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. Language modeling with gated convolutional networks. In *International conference on machine learning*, pages 933–941, 2017.

Appendices

A Comparison of Mutual Information Lower Bounds

We construct a set of distributions with known theoretical MI:

$$
\begin{aligned}
&Z \sim \mathcal{N}(0, \sigma_z^2),\ X = Z + e,\ e \sim \mathcal{N}(0, 1), \\
&Y = \begin{cases} Z + X \cdot Z + f & \text{if } Z > 0, \\ f & \text{if } Z \leq 0, \end{cases} \ f \sim \mathcal{N}(0, n^2).
\end{aligned} \tag{10}
$$

Based on the theoretical MI for bivariate Gaussian distributions [36], we can compute the conditional MI:

$$
\mathcal{I}(X; Y|Z) = \frac{1}{2} \log(1 + \frac{z^2}{n^2}). \tag{11}
$$

We conduct tests on the X, Y and Z random variables described above with $\sigma_z = 1$ and $n = 0.5$. We compare the performance of the three different estimation approaches introduced in Section 3.1 given different variable dimensions and different sizes of training data, and evaluate them using the root mean square error (RMSE) compared to the theoretical value of MI computed through Equation 11. We use a neural network with one hidden layer of 256 units as the MI estimator for each approach. We compare the performance of the three different estimation approaches given different variable dimensions and different sizes of training data, and the results are shown in Table 1. The performance of each estimation approach is evaluated based on the root mean square error (RMSE) compared to the theoretical value of MI computed through Equation 11.

Table 1: Comparison of Mutual Information Lower Bounds

Dimension	Theoretical Average MI	Training Data Size	Root Mean Square Error (RMSE)		
			VLB	KLD	JSD
1	0.2911	20000	0.0713	0.1661	0.1594
		40000	0.0424	0.1291	0.1242
		60000	0.0502	0.1509	0.1785
2	0.5821	20000	0.0974	0.3745	0.2578
		40000	0.1121	0.3517	0.3292
		60000	0.0942	0.2139	0.2105
3	0.8732	20000	0.1594	0.4825	0.4573
		40000	0.1508	0.4828	0.4573
		60000	0.1407	0.4129	0.3176
4	1.1643	20000	0.2222	0.5879	0.5406
		40000	0.1665	0.6092	0.4101
		60000	0.1611	0.4928	0.4326

From Table 1 we can see that the VLB has the lowest RMSE in all the test cases on this random variable set, whereas the KLD bound performs the worst in most cases. From the comparison between the RMSE and the absolute values of theoretical average MI we can see that it is possible to get a relatively accurate approximation of the conditional MI through numerical estimation when the mutual dependency between random variables are simple.

B Experiment Details

We implement the empowerment-based approach, the ICM approach and the Disagreement approach as intrinsic rewards with an on-policy implementation of PPO. We use on-policy PPO because intrinsic rewards are not "ground truth" rewards and their values are not very meaningful until the neural networks are trained to predict intrinsic rewards well. Since the estimation of conditional mutual information is very challenging and the empowerment networks typically take a long time

to get well trained, mixing up experiences with reward values predicted at different training steps in the same replay buffer will influence the overall performance and makes off-policy training very tricky. We use a three hidden-layer fully-connect neural network with (128, 64, 32) units in each layer for both the policy network and the value network, and set $\gamma = 0.99$ and $\lambda = 0.95$ in the PPO algorithm. We use the Adam optimizer with learning rate $2\mathrm{e}{-4}$. All experiments shown in this paper are conducted on a 10-core Intel i7 3.0 GHz desktop with 64 GB RAM and one GeForce GTX 1080 GPU.

ICM Implementation In the ICM implementation, we train the forward model f by minimizing the forward loss:

$$\mathcal{L}_t^f = \frac{1}{2}||f\big(\mathbf{s}_t^{ex}, \mathbf{a}_t\big) - \mathbf{s}_{t+1}^{ex}||_2^2. \tag{12}$$

To compute the forward loss in the ICM approach, we use one 256-unit hidden layer in the network, and we didn't compute inverse loss because the observations in this paper are poses instead of images. The value of the forward loss $\mathcal{L}_t^f$ is also used as the ICM intrinsic reward:

$$r_t^{ICM} = \mathcal{L}_t^f, \tag{13}$$

and we normalize r_t^{ICM} using running average before summing it up with the extrinsic reward to get the final reward for training the RL agent:

$$r_t = 0.01\bar{r}_t^{ICM} + r_t^e. \tag{14}$$

Disagreement Implementation In the Disagreement approach, we use the same network structure as in ICM and use five of these networks as the ensemble to compute the disagreement reward. We compute the forward losses for each of the five forward models in the same way as Equation 12, and sum up the five forward losses as the total loss to train the forward models. The intrinsic reward is calculated as:

$$r_t^{Dis} = var\{\hat{\mathbf{s}}_{t+1}^{ex,1}, \ldots, \hat{\mathbf{s}}_{t+1}^{ex,5}\}, \tag{15}$$

where $\hat{\mathbf{s}}_{t+1}^{ex,1}$ through $\hat{\mathbf{s}}_{t+1}^{ex,5}$ are the forward predictions made by the five forward models. We also use running average to get the normalized disagreement intrinsic reward $\bar{r}_t^{Dis}$ and then sum it up with the extrinsic reward to get the final reward for training the RL agent:

$$r_t = 0.01\bar{r}_t^{Dis} + r_t^e. \tag{16}$$

Empowerment Implementation For the neural network that makes empowerment prediction in the PR2 environment, we apply Gated Linear Units (GLU) [37] to improve performance. We use a neural network with four GLU layers with 256 gates each and two hidden fully-connected layers with (128, 64) units to predict $p(\mathbf{a}_t|\mathbf{s}_{t+1}^{ex}, \mathbf{s}_t)$, and calculate empowerment with the variational lower bound. Namely, we use

$$r_t^{Emp} = \log p(\mathbf{a}_t|\mathbf{s}_{t+1}^{ex}, \mathbf{s}_t) - \log p(\mathbf{a}_t|\mathbf{s}_t) \tag{17}$$

as the empowerment intrinsic reward so that in expectation, the empowerment reward being maximized is equivalent to the empowerment defined in Equation 9. In the Fetch environment, we use a neural network with six hidden fully-connected layers with (512, 512, 216, 128, 64, 32) units to approximate the T function in Equation 6 and calculate empowerment with the JS-Divergence approximation. The loss function being minimized in order to train T network is:

$$\mathcal{L}_t^{Emp} = \mathrm{sp}(-T(\mathbf{a}_t, \mathbf{s}_t, \mathbf{s}_{t+1}^{ex})) + \mathrm{sp}(T(\tilde{\mathbf{a}}_t, \mathbf{s}_t, \mathbf{s}_{t+1}^{ex})) - \log 4, \tag{18}$$

where $\mathbf{a}_t$ is the true action executed at time step t and $\tilde{\mathbf{a}}_t$ is sampled from the policy, and the reward being maximized is:

$$r_t^{Emp} = T(\mathbf{a}_t, \mathbf{s}_t, \mathbf{s}_{t+1}^{ex}). \tag{19}$$

In our empowerment-based intrinsic motivation implementation, empowerment reward and ICM reward are combined through weight coefficients to ensure that the agent can collect enough data in the nonzero empowerment region to train the empowerment network well before it is used as the intrinsic reward. The weight coefficients used in this paper are:

$$\begin{aligned} w_t^{ICM} &= 0.5 \times (1 - \tanh(200(r_t^{ICM} - 0.12))), \\ w_t^{Emp} &= 1 - w_t^{ICM}, \end{aligned} \tag{20}$$

where r_t^{ICM} is the forward prediction error (computed through Equation 12 and 13) averaged from all the parallel environments at time step t. These weight coefficients make sure that at the beginning of training when the robot don't have much interaction with the object, the coefficient for ICM reward is near 1 and the coefficient for empowerment reward is near 0. After the average ICM reward reaches a certain threshold, which means the robot have learned to interact with the object and the empowerment network can obtain enough meaningful data to get well trained, the coefficient for ICM reward switches to near 0 and the coefficient of the empowerment reward switches to near 1. Then this intrinsic reward and extrinsic task reward are combined as the RL algorithm reward:

$$\begin{aligned} r_t^i &= w_t^{ICM} \bar{r}_t^{ICM} + w_t^{Emp} \bar{r}_t^{Emp}, \\ r_t &= 0.01 r_t^i + r_t^e, \end{aligned} \tag{21}$$

where $\bar{r}_t^{ICM}$ and $\bar{r}_t^{Emp}$ are normalized using running average.

Extrinsic Task Rewards In the box-lifting task and the pick-and-place task in the Fetch environment, the object is a cube with 0.05 m edges. In the cylinder-lifting environment, the height of the cylinder is 0.1 m and the radius is 0.03 m. In the sphere-lifting environment, the radius of the sphere is 0.04 m. In both the box-lifting and sphere-lifting task, the task reward is given as Equation 22 when the center of the grippers is less than 0.01 m away from the center of the object. In the cylinder-lifting task, the condition for giving task reward is the same, but the reward is given as Equation 23. In the pick-and-place task, the task reward is 1 when the object pose is within 0.05 m of the target pose, and 0 otherwise.

$$\text{Fetch with box or sphere: } r_t^e = 50 \cdot (h - 0.01), \tag{22}$$

$$\text{Fetch with cylinder: } r_t^e = 500 \cdot (h - 0.01), \tag{23}$$

In the box-lifting task in the PR2 environment, the object is a cube with 0.06 m edges, and the task reward is given as Equation 24 when both grippers are in contact with the object and the object height is at least 0.012 m above the tabletop.

$$\text{PR2 with box: } r_t^e = 500 \cdot (h - 0.012). \tag{24}$$