Sample-Efficient Reinforcement Learning for Robot to Human Handover Tasks

Trevor Barron, Heni Ben Amor
School of Computing, Informatics, and Decision Systems Engineering
Arizona State University
Tempe, AZ 85281
tpbarron@asu.edu, hbenamor@asu.edu

Abstract

While significant advancements have been made recently in the field of reinforcement learning, relatively little work has been devoted to reinforcement learning in a human context. Learning in the context of a human adds a variety of additional constraints that make the problem more difficult including an increased importance on sample efficiency and the inherent unpredictability of the human counterpart.

In this work we used the Sparse Latent Space Policy Search algorithm and a linear-Gaussian trajectory approximator with the objective of learning optimized, understandable trajectories for object handovers between a robot and a human with very high sample-efficiency. We present an analysis of various learning scenarios and provide results for each.

Keywords: Trajectory optimization, human-robot interaction, dimensionality reduction

Acknowledgements

This work was supported in part by the NSF I/UCRC Center for Embedded Systems and from NSF grant #1361926.

1 Introduction

Reinforcement learning (RL) algorithms have been shown to solve highly challenging tasks. Harnessing these capabilities for human-assistive technologies will lead to important breakthroughs in domestic and healthcare robotics.

For example, a robot assistant may learn how to seamlessly handover objects to human partners with specific personal preferences, limited postural range, or physical disabilities. Fig. 1 shows a robot assistant offering an apple to a person with an arm fracture. By finding an individualized solution for the task, the robot could increase the likelihood of success and improve quality of life for the human. Unfortunately, standard reinforcement learning methods are ill-suited for human-in-the-loop learning scenarios due to their inherent sample complexity required to reach a reasonable solution. Both human and robot would have to jointly execute a task for long periods of time before finding a reasonable solution.

Human-centered applications of RL therefore require sample-efficient methods that can learn with a limited budget of trials, e.g., less than 100 trials. In this paper we discuss a sample-efficient RL method that is particularly well suited for human-in-the-loop learning. The rationale underlying our algorithm is that policy parameters are often correlated. Individual joints can be grouped into synergies. A combination of a small number of synergies, can lead to a large range of different possible movements. In humans, such synergies reduce the dimensionality of the control task and, in turn, reduce the cognitive effort during learning and execution [1].

Leveraging this insight, we factorize robot control policies into a matrix of synergies, as well as a low-dimensional control policy. The synergies and control policies are updated within a reinforcement learning loop thus exhausting the information provided by each trial. The result is a sample-efficient policy search method for motor skill tasks that involve the coordination of multiple joints. In human-robot collaboration experiments, the robot autonomously learned optimal handover strategies within 30-40min.

We evaluate the approach in different robot to human handover settings involving a static and mobile robot, different postural ranges of the human partner, along with three different cost functions that we propose for reinforcement learning in a human context. We emphasize that the setup is not specific to handover tasks and that the paradigm should generalize to additional scenarios.

Figure 1: A Baxter robot learning to handover an apple to a person with an arm fracture. Adaptation is performed through sample-efficient reinforcement learning.

2 Methodology

2.1 Sparse Latent Space Policy Search

Policy search methods try to find an optimal policy for an agent which acts in an uncertain world with an unknown world model. Such methods use a parametrized policy represented by $\pi_\lambda(\mathbf{a}_t|\mathbf{s}_t, t)$ with parameters λ. The goal is to identify the values for λ that maximize the expected return of the policy:

$$J(\lambda) = \int_{\mathbb{T}} p_\lambda(\tau) R(\tau) d\tau \tag{1}$$

where the expectation integrates over all possible trajectories τ in the set $\mathbb{T}$ of time steps. Each trajectory $\tau = [\mathbf{s}_{1:T}, \mathbf{a}_{1:T}]$ is the result of repeatedly applying policy π_λ in all states $\mathbf{s}$ the agent encounters.

Our goal is to create a specific policy search algorithm for collaborative tasks that converges on an optimal policy with less than 100 executions. This goal can be achieved by reducing the dimensionality of the reinforcement learning process. Using latent variable estimation techniques, Luck, et. al. [2] derive a policy search variant that uncovers the low-dimensional manifold of solutions during the search process. To this end, we factorize the action at time step t according to:

$$\mathbf{a}_t^{(m)} = \left(\mathbf{W}^{(m)}\mathbf{Z}_t + \mathbf{M}^{(m)} + \mathbf{E}_t^{(m)}\right)\phi(\mathbf{s}_t, t), \tag{2}$$

where $m \in [1, 2, \ldots, M]$ defines the number of the group and $\mathbf{a}_t^{(m)} \in \mathbb{R}^{D_m \times 1}$ the D dimensions of the action vector for a specific time step t. The vector $\mathbf{a}_t$ is a linear projection of the feature vector $\phi(\mathbf{s}_t, t) \in \mathbb{R}^{p \times 1}$. The mean matrix is given by $\mathbf{M} \in \mathbb{R}^{D \times p}$ and the exploration noise by the entries of $\mathbf{E} \in \mathbb{R}^{D \times p}$.

The above factorization can be seen as a low-dimensional policy that is projected into a higher-dimensional space through the projection matrix $\mathbf{W}$. This yields a class of policy search algorithms that simultaneously searches for a

lower-dimensional policy, as well as the projection matrix $\mathbf{W}$ that embeds the low-dimensional latent signal into the high-dimensional space of robot control parameters. Using such a *latent space* approach for robot learning results in highly sample efficient policy search methods that can learn from a small number of executions in the real-world.

2.2 Policy Representation

We use a linear-Gaussian time-dependent trajectory representation similar to a Dynamic Motor Primitive (DMP) [3]. However, in contrast to DMPs, we do not employ a forcing function. A trajectory is represented by a set of weights and their respective basis functions which are spaced equally over the time of the trajectory.

In order to ensure invariance with respect to the position and orientation of the user, the linear-Gaussian policy is defined in the coordinate frame of the human partner (as opposed to the joint space of the robot). The trajectory can be transformed to robot space and then converted to joint space by using inverse kinematics. Defining the policy in the human frame permits a simpler generalization, as the trajectory is specified relative to human movements. Additionally, constraints on human posture can also be easily encoded in this space. Finally, representing policies in the coordinate space of the human permits training over fewer parameters than using joint coordinates.

Before running RL, imitation learning is used to derive an initial policy. We provided a few samples (ten in our experiments) defining a reasonable handover trajectory. Consequently, the trajectory representation is fit to these samples.

2.3 Reward Functions

Subsequently, we introduce various reward functions that are well-suited for human-robot interaction scenarios.

Optimizing Distance: Perhaps the most intuitive objective when learning a handover task is the distance between the robot's end-effector and the human's hand. This turned out to be a quite effective method but was susceptible to learning unintuitive trajectories since the only reward signal was the final distance. Accordingly, we try two additional methods to resolve the jerkiness in the trajectory while still optimizing for distance. Formally, the cost function we minimize is

Symbol	*Meaning*
t	Total number of time steps in a trajectory
d	Total number of degrees of freedom
b	Total number of basis functions. Also the number of weights per DOF.
Ψ	The set of Gaussian basis functions spaced over time in $\mathbb{R}^{b \times t}$
W	The weights on the basis functions that define a trajectory in $\mathbb{R}^{b \times d}$
H	A matrix representing the position of the human in each DOF over time in $\mathbb{R}^{t \times d}$

Table 1: Notation used to describe our cost functions

$$D(W, H) = \frac{1}{2} \sum_i \|\boldsymbol{\psi}_i^T W - \boldsymbol{h}_i\|^2 e^{-(t-i)}. \tag{3}$$

The notation used throughout the paper can be found in Tab. 1. We add an exponential scaling term that gives a higher weight to time steps close to the final position.

Optimizing Jerk: To combat some of the jerky, counterintuitive trajectories that can result from optimizing only for distance we also try optimizing over a combination of closeness to the human's hand and minimal trajectory jerk. This motivation behind this objective stems from an experimentally validated theory that humans make arm movements with minimal jerk trajectories [4]. The jerk cost for a trajectory is the sum of squared jerk over time averaged over degrees of freedom.

$$S(W) = \frac{1}{2d} \sum_i \left(\frac{\delta^3 \boldsymbol{\psi}_i^T W}{\delta t^3} \right)^2 \tag{4}$$

The total cost is then a linear combination of the jerk cost and the distance cost. We perform a grid search to find an approximate minimum of $C(W) = D(W, H) + \lambda S(W)$ and find an approximate solution at $\lambda = 0.5$.

3 EXPERIMENTS

We ran all of our experiments on the Rethink Robotics Baxter robot mounted on a Clearpath Robotics Ridgeback omnidirectional base. We use Kinect skeleton tracking to monitor the human position. We run several experiments analyzing various reward functions for human robot interaction along with eight different scenarios that validate our robot to human handover approach. First we describe the various objectives and then each scenario. Four human subjects performed each of the eight scenarios.

3.1 Human-In-The-Loop Experiments

We run a total of 32 experiments with four human subjects: with and without the mobile base, with the human sitting and standing, and with and without a sling simulating a lack of mobility. All of our experiments used eight iterations and 10 interactions per iteration. In all human experiments we optimize only for distance from the end-effector to the human's left hand. This results in 90 training interactions (20 in the first iteration).

4 Results

We achieve good convergence in all scenarios and often find a good solution in less than five iterations, or about 50 interactions. Table 2 shows specific results per task per subject. Due to noise in the Kinect skeleton tracking and variability in the human we cannot expect to reach a distance of zero. Averaged over all trials, we reach a final distance of approximately twenty centimeters, which accounting for these sources of error, is well within the range of a grasp target even for someone with a mobility impairment.

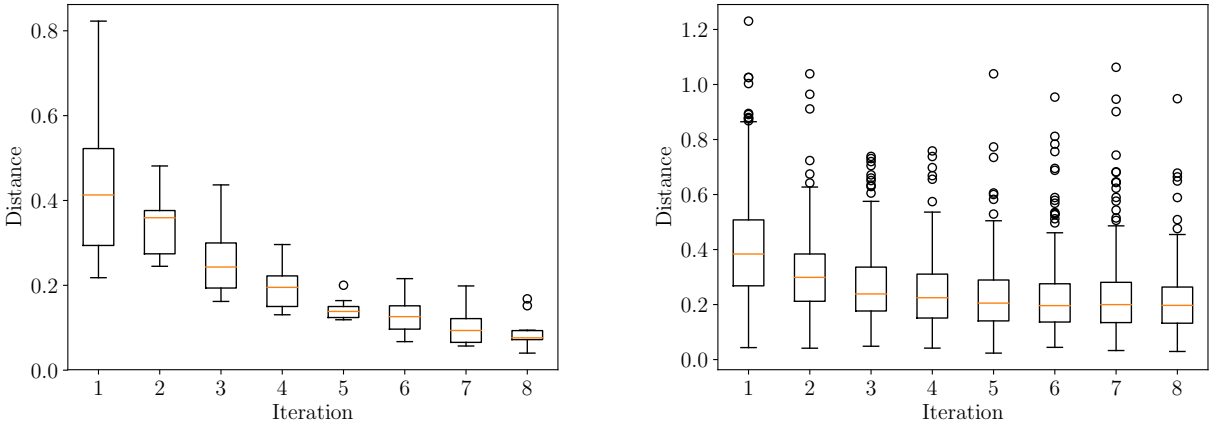

Figure 2: Distribution of distances calculated from a single experiment (left) and all performed experiments (right) at each iteration. We can see that even averaged over all experiments the distance to the user is shrinking.

Fig. 2 (left) shows the variance in distance from within a single RL experiment. During the initial trials, high exploration produces a range of different distance values. However, as learning progresses, the distance shrinks to below 10cm.

While we used only the distance cost in the optimization, the trend of the other costs throughout training is still interesting. We find that optimizing for distance still results in reasonably low jerk but appears to have little to no effect on correlation. During the first iteration exploration is high and results in very large jerk costs. After the first policy update, we achieve a similar jerk cost to the initial policy and it does not decrease further.

Scenario	*Subject #1*	*Subject #2*	*Subject #3*	*Subject #4*
Non-Mobile				
├─Standing				
│ ├─No-Sling	0.124	**0.063**	**0.040**	0.095
│ └─Sling	0.204	0.144	*0.205*	0.150
└─Sitting				
├─No-Sling	**0.092**	0.118	0.070	0.152
└─Sling	0.177	0.203	0.116	0.363
Mobile				
├─Standing				
│ ├─No-Sling	0.142	0.224	0.174	0.144
│ └─Sling	*0.342*	*0.313*	0.181	*0.374*
└─Sitting				
├─No-Sling	0.110	0.202	0.168	0.119
└─Sling	0.200	0.097	0.113	**0.054**

Table 2: Results per individual per task measured in meters between the endeffector and the human's hand. The best trial for each subject is bolded; the worst trial is italicized.

It is perhaps most instructive to examine when the algorithm performed very well or very poorly. Notice that three of the four subjects worst results were on tasks involving the mobile base. While a sample size of four is not sufficient to make a statistical conclusions we do believe the adding base movement (even only in one dimension) makes the task much more difficult. This relates to reward function design. Unless we hand engineer the reward to take into account both base movement and arm movement, an interaction could have excellent base movement but still receive a high cost due to poor arm movement. The opposite is also possible. Learning in this scenario requires that exploration early in training experience high value states, which becomes increasingly less likely as the dimensionality increases.

5 Subjective Observations

All of the human subjects in our experiments had prior experience working with robots. Since our emphasis in this work is on efficient learning in the context of a human we did not conduct a formal survey. Nevertheless we did gather some observations of the training experience. None of the subjects reported feeling unsafe during the process though all were familiar with the Baxter robot. Still, safety is a major concern when learning a human context. Our setup had no constraints that stopped the base from driving into the human or hitting the human mid trajectory. We did not experience any safety issues but these are important considerations. The most common observation was that the movement of the robot was unpredictable especially early during training with exploration is high.

6 Related Work

There is some existing work with ties to our own. DMPs are a very common means of trajectory representation in robotics and have been used in reinforcement learning applications but to our knowledge have not been used in a interactive context [5]. Likewise, imitation learning has become a common method of initializing policies, especially in robotics, to provide prior information and reduce training time [6]. The existing work in human-robot learning has largely been focused on learning distributions over trajectories using imitation learning. Ben Amor developed the Interaction Primitives framework, an extension of DMPs that also models the human movement and conditions the trajectory of the robot on the observation of the human [7],[8], [9]. Unlike an Interaction Primitive, our work only determines the trajectory based on the initial position of the human and does not condition the robot movement on additional observations of the human over time. This is the logical next step in our work. Finally, [10] also adapt handover trajectories to human limitations. Unlike our work, they also perform an estimation step over the rewards which are specified by the human.

7 Conclusion and Future Work

This work examines a sample efficient method for reinforcement learning in human-robot interaction, specifically handover tasks. While our approach does converge to good solutions, we feel the need to mention some significant limitations whose resolution we leave for future work. First, since each trajectory timestep is not conditioned on the state of the human, the current setup generalizes poorly with human movement. Second, since the algorithm adds random noise to the model weights for each interaction, the movement along the trajectory is not guaranteed to be time correlated. This often results in trajectories during exploration that are not intuitive to the human counterpart.

References

[1] M. Santello, M. Flanders, and J. Soechting, "Postural hand synergies for tool use," *The Journal of Neuroscience*, vol. 18, no. 23, 1998.

[2] K. S. Luck, J. Pajarinen, E. Berger, V. Kyrki, and H. B. Amor, "Sparse latent space policy search." in *AAAI*, 2016, pp. 1911–1918.

[3] S. Schaal, "Dynamic movement primitives-a framework for motor control in humans and humanoid robotics," in *Adaptive motion of animals and machines*. Springer, 2006, pp. 261–280.

[4] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *Journal of neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[5] M. P. Deisenroth, G. Neumann, J. Peters, *et al.*, "A survey on policy search for robotics," *Foundations and Trends® in Robotics*, vol. 2, no. 1–2, pp. 1–142, 2013.

[6] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[7] H. B. Amor, D. Vogt, M. Ewerton, E. Berger, B. Jung, and J. Peters, "Learning responsive robot behavior by imitation," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 3257–3264.

[8] H. B. Amor, G. Neumann, S. Kamthe, O. Kroemer, and J. Peters, "Interaction primitives for human-robot cooperation tasks," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2831–2837.

[9] M. Ewerton, G. Neumann, R. Lioutikov, H. B. Amor, J. Peters, and G. Maeda, "Learning multiple collaborative tasks with a mixture of interaction primitives," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1535–1542.

[10] A. Kupcsik, D. Hsu, and W. S. Lee, "Learning dynamic robot-to-human object handover from human feedback," *CoRR*, vol. abs/1603.06390, 2016. [Online]. Available: http://arxiv.org/abs/1603.06390