

Learning Cost Functions for Motion Planning of Human-Robot Collaborative Manipulation Tasks from Human-Human Demonstration

Jim Mainprice and Dmitry Berenson

Robotics Engineering Program, Worcester Polytechnic Institute
jmainprice@wpi.edu, dberenson@cs.wpi.edu

1 Introduction

Factory automation has revolutionized manufacturing over the last 50 years, but there is still a large set of manufacturing tasks that are tedious or strenuous for humans to perform. We believe that human-robot collaboration can enable safe and effective task execution while reducing tedium and strain of the human. In this paper we address an important step toward human-robot collaboration: learning a cost function for the robot motion planner which accounts for the human-robot interaction constraints that arise when sharing the workspace with a human.

Motion planning for robot manipulation in close proximity to humans poses two main challenges: first on the algorithmic techniques to be used and second on the definition of safety and comfort metrics. Our prior work Mainprice et al. (2011) has focused on considering the human's safety, visibility, and musculoskeletal comfort directly in the robot configuration space and solving motion planning using sampling-based and stochastic-optimization techniques. More recently Dragan and Srinivasa Dragan and Srinivasa (2013) have proposed a legibility metric inspired by the psychology of action interpretation, which was applied to robot motion planning. Our recent work Mainprice and Berenson (2013), incorporates early prediction of human motion in a dynamic motion planner capable of minimizing the interference with the human and falling back quickly to a feasible task by reasoning on a prediction of human workspace occupancy. All these approaches enforce human-robot interaction constraints by defining cost functions which the motion planner then uses to produce "human-aware" motions. Defining such a function is generally assumed to be easier than directly specifying the sequence of commands. However balancing the importance of each term in the function to obtain a consistent robot behavior is generally performed manually and can be quite unintuitive. In this work we hypothesize that it is possible to mimic human-motion generation with a dynamic motion planning process. Thus we believe that it is possible to learn a cost function that encodes the desired objectives of collaborative manipulation from a library of human-human collaborative manipulation motions

In the next Section we first describe the Inverse Opti-



Figure 1: Human-human experiment (right) and sampling of manipulation motions for inverse optimal control (left).

mal Control (IOC) or Inverse Reinforcement Learning (IRL) Abbeel and Ng (2004) method that enables us to recover the cost function from a linear combination of abstract feature functions. We then describe a set of features, inspired from our previous work Mainprice et al. (2011, 2012) on "human-aware" motion planning, to produce collaborative manipulation motions. In Section 3, we present initial results that illustrate the capacity of the IOC algorithm to learn a cost function in a static setting using the features described in Section 2. We have ran a human-human experiment (see Figure 1) and aim to use the collected data to learn a cost function for human-robot collaborative tasks from real demonstrations with the presented framework in the near future.

2 Approach

In order to learn a cost function for motion planning of collaborative manipulation tasks we proceed in two phases. First we gather a library of human-human collaborative motions segmented in elementary reaching motions (i.e., from a resting configuration to a grasping configuration). IOC is then performed using the segmented motions as demonstrations. In a second phase we use the learnt cost function inside a replanning loop to account for the human partner motions in real time.

Inverse Optimal Control Algorithm

We wish to perform IOC of human upper-body motions. Such motions can be represented as time-parametrized curves τ in the human's configuration space. Because these motions are inherently high-dimensional (in this work we consider 15 DoFs), global optimality maybe unfeasible. Hence we make use of the *path integral inverse reinforcement learning* algorithm Kalakrishnan et al. (2013),

which can deal with high-dimensional continuous state-action spaces, and only requires local optimality. We consider an active human, whose trajectories are used as demonstrations, and a passive human which is considered part of the context associated with that motion. A context comprises the passive human configuration as well as the locations of objects in the scene. As we aim to recover cost functions for a dynamic motion planning process, we then segment each demonstration into several smaller trajectories by advancing the active human along the trajectory and modifying the context with the current configuration of the passive human.

The original inverse optimal control problem solved by the *path integral inverse reinforcement learning* algorithm aims to recover a cost function composed of a control cost term and a general (i.e., configuration dependent) cost term that can be combined with a terminal cost term, which we do not use here. Our formulation of the problem considers linearly parametrized cost functions. Each feature function is user defined and models a desired property. A feature function penalizes motions which do not respect the associated property, see Section 2 for a description of all features. The cumulative cost $C(\tau)$, and feature count $\Phi(\tau)$ of a trajectory are defined as follows:

$$C(\tau) = w^T \Phi(\tau), \Phi(\tau) = \begin{bmatrix} G(\tau) \\ A(\tau) \end{bmatrix}$$

where Φ is a multi valued feature function defined by the user, w are weights associated with the features which the algorithm attempts to learn. A is a term enforcing smoothness (i.e., control cost) and G a general term of the form:

$$G(\tau) = \int_{t=0}^T \phi(q_t) dt \simeq \sum_{i=1}^N \phi(q_i) \delta t$$

where q_i is the configuration at index i along the trajectory.

The IOC algorithm samples trajectories with low control features around each demonstration from a multivariate gaussian distribution with covariance R^{-1} , for a definition see Kalakrishnan et al. (2011). The standard deviation is tuned manually. The recovered weights are obtained by solving the following convex minimization problem using gradient descent:

$$\min_w - \sum_{i=1}^D \log \frac{e^{w^T \Phi_i}}{\sum_{k=1}^K e^{w^T \Phi_{i,k}}}$$

where D is the number of demonstrations and K the number of samples per demonstrations. For more details on the *path integral inverse reinforcement learning* algorithm, see Kalakrishnan et al. (2013).

Feature functions

We consider variants of the feature functions that have been introduced in previous, inspired from the *proxemics* theory Hall and Hall (1969), to account for human-robot interaction constraints.

Distance between human links The goal of this feature is to avoid collision risks. We consider multiple distances (e.g., wrist to wrist, wrist to elbow, wrist to pelvis).

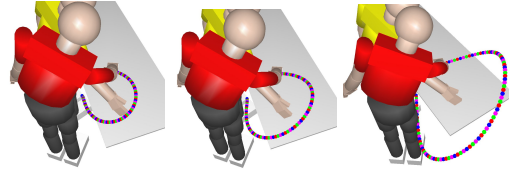


Figure 2: Three trajectories computed using the STOMP motion planner (a) with a user given weight (b) with the recovered weight (c) with a random weight vector.

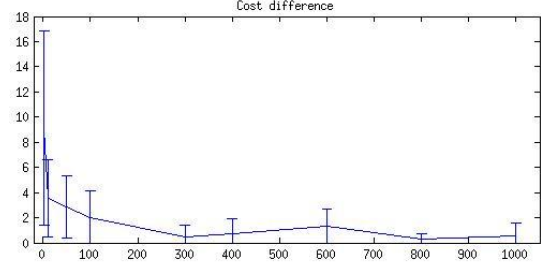


Figure 3: Difference in cost using the recovered weight vector and the original weight vector, function of the number of sample trajectories used by the IOC algorithm, planned on the example of Figure 2

Visibility Humans generally feel more comfortable when the other agent, robot or human, is manipulating within his/her field of view. To enforce this, we consider multiple points of interest on the active human body and measure their distances to the center of the field of view of the passive human.

Musculoskeletal effort This feature measures the comfort of the active human. It is evaluated using three functions introduced in Marler et al. (2005) : 1) a joint angle distance from a resting posture to the actual posture, 2) the potential energy of the arm by measured as the relative height of the arm and forearm, and 3) the distance to the joint limits.

Smoothness This feature ensures that the trajectory remains smooth. We measure, length of the configuration space trajectory, length of the task space trajectory, and sum of accelerations along the trajectory.

3 Results

In this section we present preliminary results illustrating the capacity of the algorithm to recover a cost function using *distance between links* and *smoothness* features. We first planned a trajectory with a user input weight vector using the STOMP algorithm Kalakrishnan et al. (2011), see Figure 2. We then used the IOC algorithm presented in Section 2 to recover a weight vector using this planned trajectory as a demonstration. Figure 3 shows the difference in cost using the original weights, between the original trajectory and the trajectories planned using the recovered weights as the number of samples used by the IOC algorithm increases. The results are averaged over 10 runs. The mean and standard

deviation decrease as the number samples increases which indicates the capacity of the algorithm to recover cost functions for the type of reaching motions we consider.

We are currently gathering a library of human-motions using a motion capture system and we will apply the approach described in this paper to the data and evaluate its efficacy.

References

- Abbeel, P., and Ng, A. Y. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, 1. ACM.
- Dragan, A., and Srinivasa, S. 2013. Generating legible motion. RSS.
- Hall, E. T., and Hall, E. T. 1969. *The hidden dimension*, volume 1990. Anchor Books New York.
- Kalakrishnan, M.; Chitta, S.; Theodorou, E.; Pastor, P.; and Schaal, S. 2011. STOMP: Stochastic trajectory optimization for motion planning. In *ICRA*.
- Kalakrishnan, M.; Pastor, P.; Righetti, L.; and Schaal, S. 2013. Learning objective functions for manipulation. In *IEEE International Conference on Robotics and Automation*.
- Mainprice, J., and Berenson, D. 2013. Human-robot collaborative manipulation planning using early prediction of human motion. In *IROS*.
- Mainprice, J.; Akin Sisbot, E.; Jaillet, L.; Cortés, J.; Alami, R.; and Siméon, T. 2011. Planning human-aware motions using a sampling-based costmap planner. In *ICRA*.
- Mainprice, J.; Gharbi, M.; Siméon, T.; and Alami, R. 2012. Sharing effort in planning human-robot handover tasks. In *ROMAN*.
- Marler, R. T.; Rahmatalla, S.; Shanahan, M.; and Abdel-Malek, K. 2005. A new discomfort function for optimization-based posture prediction. Technical report, SAE Technical Paper.