A Differentiable Policy for Shared Autonomy

Yoojin Oh[1,2], Hangbeom Kim[3], Marc Toussaint[1] and Jim Mainprice[1,2,4]

yoojin.oh@ipvs.uni-stuttgart.de, hang.beom.kim@ipa.fraunhofer.de

marc.toussaint@informatik.uni-stuttgart.de, jim.mainprice@ipvs.uni-stuttgart.de

[1]Machine Learning and Robotics Lab ; MLR ; Stuttgart University ; Stuttgart, Germany.

[2]Humans to Robots Motion Research Group ; HRM ; Stuttgart University ; Stuttgart, Germany.

[3]Department of Machine Vision and Image Processing; Fraunhofer IPA ; Stuttgart, Germany.

[4]Autonomous Motion Department, Max Planck Institute for Intelligent Systems ; Tübingen, Germany

***Abstract*— In this paper we present a framework for the teleoperation of pick and place tasks. We define a *shared control* policy that allows to blend between direct user control and autonomous control based on user intent inference. One of the main challenges in shared autonomy systems is to define the arbitration function, which decides when to let the autonomous agent take over. In this work, we propose a differentiable policy model that integrates motion generation, user intent inference and arbitration. Full differentialbilty of the policy is desirable to further train the shared autonomy system using Reinforcement Learning (RL). We present initial results teleoperating a gripper in a virtual environment using pre-training and hand tuning of the arbitration function. Our results demonstrate the efficacy of the approach when the intent inference module is trained on a task similar to the one performed at test time. Our results also shed light on limitations that we believe demonstrate the need for a shared autonomy RL setup.**

I. INTRODUCTION

Traditional approaches to robot teleoperation [1] rely on the user assigning low-level (*direct control*) or mid-level (*traded control*) commands to be performed by the robot. Difficulties originate both from limited situation awareness and the discrepancy between the human and robot morphology. This can lead to human operational errors, which can be compensated by extensive prior practice. One way to mediate these difficulties is *shared control*, combining human intelligence with the robot's autonomy.

Shared autonomy is an active field of research [2], [3], [4], [5]. These systems usually rely on: 1) inferring the user intent and 2) providing assistance when the *confidence* with respect to the intent is high enough. Blending between the user and robot policies is performed through *arbitration*, which is difficult to caracterize in practice [2]. Previous works have proposed to solve arbitration by using hindsight optimization to solve an POMDP [5], or through deep RL [4] in discrete action spaces. In this work we aim to extend this approach to continuous action spaces, by incorporating ideas from guided policy search [6] with the aim to train shared autonomy systems end-to-end.

Thus, we propose a differentiable graph model of the policy, see Figure 1, that integrates motion generation, user intent inference and arbitration. We train the motion generation module to mimic an optimal controller and the intent inference module to predict goal distributions using data collected in a direct control phase. Note that the training scheme is independent of the policy model and that the training proposed in this paper can be viewed as the pre-training step of a RL setup.

This research was supported in part by the System Mensch alliance and the Max-Planck-Society. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding organizations.

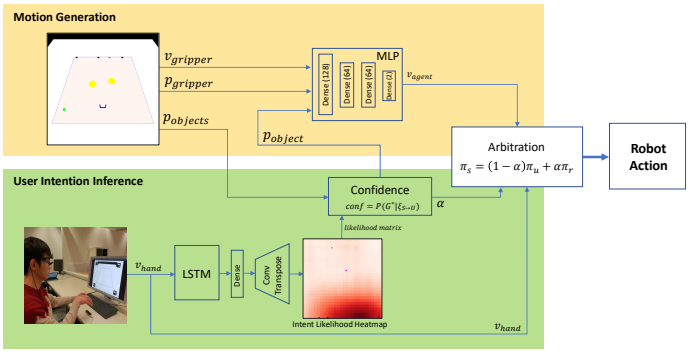

Fig. 1. Policy Architecture. Virtual environment with 4 target objects (top left). User interacting with the setup (bottom left). The arbitration module blends between autonomous motion generation and direct control of the human.

To assess the efficacy of the approach, we defined a teleoperation task where a user controls a virtual gripper using velocity commands v_{hand} using hand motion. We performed a pilot user experiment and compare completion times on two different tasks (with and without obstacles).

II. POLICY ARCHITECTURE

Our shared autonomy policy $\pi_s = (1-\alpha)\pi_u + \alpha\pi_r$, blends the robot policy π_r (i.e., motion generation) and the user policy π_u (i.e., direct control). In our experiments these policies map positions to velocities. The arbitration α is computed using the confidence that the robot has of the intent inference makes at any time.

a) Motion generation: We train a Multi-Layer Perceptron (MLP) using 24K trajectories to mimic trajectories generated from a Gauss-Newton trajectory optimizer [7]. This step is similar to the first step of guided policy search (i.e., approximate optimal pick-and-place motion given the positions of the object to grasp and the position to place). Thus, given the user intention p_{object}, we can infer the optimal action v_{agent}.

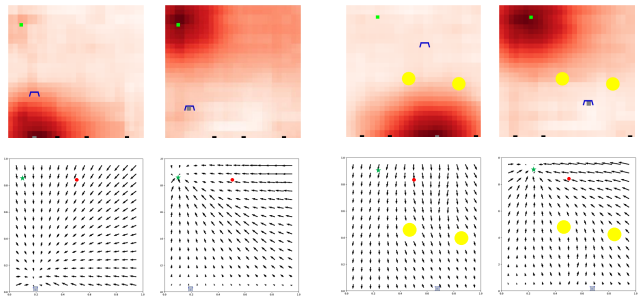

Fig. 2. Predicted intent likelihood 28x28 heatmap (top) and robot policy (bottom) for pick-and-place tasks in an environment with no obstacles (two left columns) and environment with obstacles (two right columns).

b) User intention inference: We train a Long Short-Term memory (LSTM), followed by a convolution transpose layer to predict the intent likelihood as a heatmap grid representing the environment (see Figure 2). The intent prediction module is trained on direct control data from 19 participants (approx. 1400 episodes), where the goal is known.

c) Arbitration: Based on the intent likelihood heatmap, intensity scores for each object in p_{object} can be extracted. The scores for the past n timesteps are accumulated and we take the highest softmax value of the accumulated scores as $p(G^*|\xi_{S\rightarrow U})$, which we use as confidence [2]. The object with highest probability is used as target. The cosine similarity between v_{agent} and v_{hand} is multiplied to the confidence to compensate for incorrect intent inference. Arbitration α is proportional to the confidence and set to one when it exceeds a threshold.

III. Preliminary Experiment and Results

We created a 2-dimensional pick-and-place environment in simulation as shown in the top-left image in Figure 1. The environment is displayed in a tilted angle to generate a perspective view as if the user is looking at a teleoperation environment from a camera mounted on a robot.

We performed a user experiment with 8 subjects asked to perform pick-and-place manipulation task using their hand motion, i.e. reaching out the hand to grab object and pulling back hand to retrieve object. The motion data was captured using a Leap Motion hand gesture sensor. Subjects performed the experiment for two different modes, (i.e. direct and shared control), for a sequence of 12 episodes twice. The sequence included random order of direct and shared control and the second sequence consists of the opposite order episodes such that the subjects test the same episode for both control modes. Subjects were allowed to practice each mode for a very short time before the test. The experiment was repeated with and without obstacles. Figure 3 shows the result of average completion times for both control modes and environments.

For the environment without obstacles, the average completion time with shared control is shorter than with direct control. This shows that shared control helped the subject

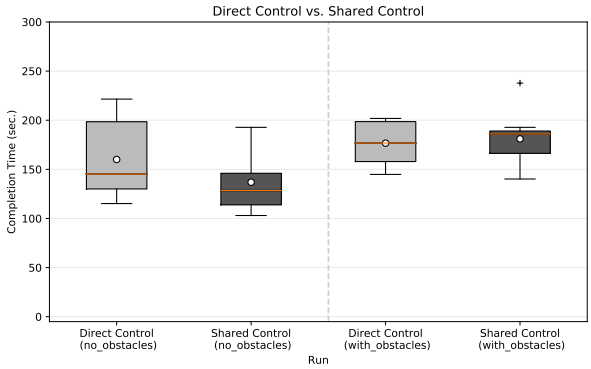

Fig. 3. Comparing the average completion time of across all episodes for direct control and shared control, with and without obstacles.

perform the task faster whereas in direct control the subjects had to be more cautious when controlling the gripper.

For the environment including obstacles, the average completion time with shared control is slightly longer than with direct control. Since the intent inference was trained in an environment without obstacles and it takes into account the past trajectory to predict the future goal, the curve motions intended only to avoid an obstacle made by the participants often lead to wrong predictions. This error in the prediction made the system assist for the wrong goal, and as a consequence subjects had to fight the controller to make the gripper move towards the subjects' true intention, which suggest the need to adapt the prediction module to the task.

In this work we introduced a differentiable policy for shared autonomy and preliminary results using the proposed framework. Our experiments show that even with high confidence, the predicted result may not be what the user really wants. It is difficult to simply define a relationship between confidence and arbitration, thus our goal is to express a formalism for shared control which embeds intent inference and arbitration that is able to be trained end-to-end.

References

[1] C. Phillips-Grafflin, H. B. Suay, J. Mainprice, N. Alunni, D. Lofaro, D. Berenson, S. Chernova, R. W. Lindeman, and P. Oh, "From autonomy to cooperative traded control of humanoid manipulation tasks with unreliable communication," *Journal of Intelligent & Robotic Systems*, vol. 82, no. 3-4, pp. 341–361, 2016.

[2] A. D. Dragan and S. S. Srinivasa, "A policy-blending formalism for shared control," *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 790–805, 2013.

[3] S. Nikolaidis, Y. X. Zhu, D. Hsu, and S. Srinivasa, "Human-Robot Mutual Adaptation in Shared Autonomy." *HRI*, 2017. [Online]. Available: https://dblp.org/rec/conf/hri/NikolaidisZHS17

[4] S. Reddy, A. D. Dragan, and S. Levine, "Shared autonomy via deep reinforcement learning," *arXiv preprint arXiv:1802.01744*, 2018.

[5] S. Javdani, H. Admoni, S. Pellegrinelli, S. S. Srinivasa, and J. A. Bagnell, "Shared autonomy via hindsight optimization for teleoperation and teaming," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 717–742, May 2018. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364918776060

[6] S. Levine and V. Koltun, "Guided Policy Search." *ICML*, 2013. [Online]. Available: https://dblp.org/rec/conf/icml/LevineK13

[7] J. Mainprice, N. Ratliff, and S. Schaal, "Warping the workspace geometry with electric potentials for motion optimization of manipulation tasks," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 3156–3163.