

# Show Me What You Can Do: Capability Calibration on Reachable Workspace for Human-Robot Collaboration

Xiaofeng Gao<sup>1</sup>, Luyao Yuan<sup>1</sup>, Tianmin Shu<sup>2</sup>, Hongjing Lu<sup>3</sup>, Song-Chun Zhu<sup>1</sup>

**Abstract**—Aligning humans’ assessment of what a robot can do with its true capability is crucial for establishing a common ground between human and robot partners when they collaborate on a joint task. In this work, we propose an approach to calibrate humans’ estimate of a robot’s reachable workspace through a small number of demonstrations *before* collaboration. We develop a novel motion planning method, REMP (Reachability-Expressive Motion Planning), which jointly optimizes the physical cost and the expressiveness of robot motion to reveal the robot’s motion capability to a human observer. Our experiments with human participants demonstrate that a short calibration using REMP can effectively bridge the gap between what a non-expert user thinks a robot can reach and the ground-truth. We show that this calibration procedure not only results in better user perception, but also promotes more efficient human-robot collaborations in a subsequent joint task. The supplementary material is available at [https://xfgao.github.io/paper/IROS\\_2021\\_CapCali\\_Sup.pdf](https://xfgao.github.io/paper/IROS_2021_CapCali_Sup.pdf).

## I. INTRODUCTION

One of the main challenges in Human-Robot Interaction is that the capacity of the robot perceived by the human partner may not be consistent with its actual capacity [1], [2], [3]. Such discrepancy may lead to overuse or misuse of the robot. For instance, as shown in Figure 1a, when incorrectly estimating the robot’s capacity, the efficiency of the collaboration will be greatly impaired [4]. Particularly, in an ad-hoc teaming setting [5] where humans do not have prior experience with their robot partners, the consequence caused by such discrepancy could be detrimental to the team collaboration [6].

In this work, our key insights to address this challenge are two-fold: i) humans’ perception of the capability of a robot can be calibrated by observing its behavior, e.g., robot demonstrating its motion trajectories in pursuit of certain goals, and ii) calibrating the perceived robot capability improves the quality of subsequent human-robot collaboration.

We focus on a case study as shown in Figure 1, where a human user and a robot share the same workspace, and they must take turns picking up all objects in the shared workspace as fast as possible. The robot can only reach part of the workspace due to its mechanical limits, the human partner needs to pick up the objects that the robot can not reach in order to achieve maximum efficiency in completing this joint task. We introduce capability calibration as shown in Figure 1b, where we allow the robot to show a small number of demonstrations for reaching a target location.

After watching each demonstration, the human can estimate the robot’s capability accordingly. The goal is to come up with motion plans to pragmatically demonstrate the robot’s capability by modeling how humans would update their beliefs based on what they observe from the demonstrations.

For achieving a sample-efficient capability calibration, we propose REMP (Reachability-Expressive Motion Planning), a novel planning algorithm that models perceived robot capability as a human user’s belief over a robot’s reachable workspace, and integrates the belief update into motion planning by introducing an additional cost in trajectory optimization. As a result, REMP can generate a series of expressive trajectories for different robots to showcase their reachability to human users. We conducted a user study in which participants i) first observed several robot demonstrations, then ii) estimated where the robot could reach, and finally iii) proceeded to work with the same robot in a joint task: picking up all objects in the shared workspace as fast as possible. We find that i) REMP can significantly increase the accuracy of humans’ estimate of robot capability, ii) the subsequent human-robot collaboration benefits from a successful calibration, iii) users perceive the robot as more predictable and reliable, and report it is easier to understand the robot’s capability.

## II. RELATED WORK

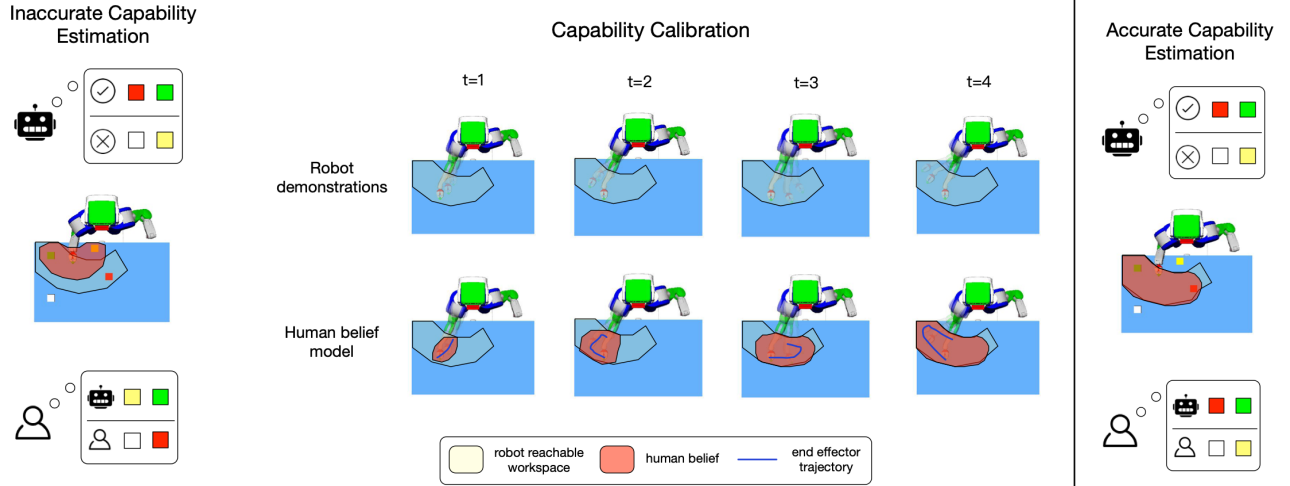
**Perceived robot capability.** There have been works studying different aspects of how humans perceive a robot’s capability. [7] investigated the effects of robot speed and speech on perceived capability. [8] introduced a Bayesian network as an incomplete model to encode the agent’s all possible capabilities in a given domain. [9] used a game-theoretic approach to model human’s expectations of the robot capability over time. As human’s estimation of the robot’s capability plays an important part in their trust in the autonomous robot [10], [11], previous works have also studied how capability inference affects human’s trust and reliance on the robot [12], [13], [14]. More recently, [15] formulated the intent and capability calibration problem as a Bayes Adaptive POMDP, and assumed that the agents model each other’s ability based on experience counts of action success or failure. Unlike the previous work which focused on capability models on discrete action space, this work, to the best of our knowledge, is the first to integrate perceived capability models into motion planning.

**Robot expressive motions.** As robots are increasingly being deployed beyond isolated factory environments in places

<sup>1</sup> Center for Vision, Cognition, Learning, and Autonomy, UCLA. Emails: {xfgao, yuanluyao}@ucla.edu, sczhu@stat.ucla.edu.

<sup>2</sup> Massachusetts Institute of Technology. Email: tshu@mit.edu.

<sup>3</sup> Department of Psychology, UCLA. Email: hongjing@ucla.edu.



(a) Inaccurate capability estimation can lead to failure in collaboration.

(b) Illustration of our framework for improving the quality of human-robot collaboration via capability calibration.

Fig. 1: (a) Consider a collaborative table clearing task, where the robot has a limited capability and cannot reach the yellow and white objects. Inaccurate estimation of the robot’s reachable workspace would harm collaboration: users who incorrectly estimate that the robot can reach the yellow object would assign it to the robot, resulting in a worse teaming performance. (b) To address this issue, we propose the capability calibration, where the robot uses its motion to demonstrate its capability to the human, who, by observing the demonstrations, forms a belief over the robot’s capability. Then the two proceed to a collaboration task, where the human is supposed to take responsibility for objects unreachable to the robot. In this example, the human is supposed to pick up the white and the yellow cubes and let the robot collect the red and the green ones.

where they may interact directly with humans, purely functional motions only aiming to accomplish tasks are inadequate for the human users to correctly understand the robots and establish effective collaborations [16], [17]. [18] defined three types of robot motions: functional, predictable and legible. This line of research aims to not only finish the task but also convey the rationality and the intent of the robot to the user [19], [20]. The generation of legible trajectories was tackled as an optimization problem in [21], [22], [23], solved with functional gradient descent and policy improvement respectively. Similar ideas of optimization were also adopted to study the robot expression of emotion [24], deception [25], style [26] and robot incapability [27]. In this paper, we also model the expressive trajectory generation as an optimization problem. However, unlike emotion and intent, capability can be a task-agnostic and time-invariant attribute of the robot. Thus, we can have a separate calibration without any specific task involved. Moreover, we integrate human belief in our cost function and accommodate human belief update to maximize the expressiveness of our trajectories. For a more comprehensive review of robot expressive motions, we refer readers to this survey [17].

### III. CAPABILITY CALIBRATION

We propose a capability calibration framework (as shown in Figure 1b) to align a human user’s understanding of a robot’s capability with the ground-truth, where the user has a chance to watch a small number of demonstrations from her robot partner before they start working together. We assume

that the user has no prior knowledge about a robot’s capability, but can estimate it based on observed robot motion. In this section, we introduce a novel approach for pragmatically and procedurally generating such demonstrations that can optimally reveal the true robot capacity. We show how this calibration could be applied to collaboration in Section IV.

#### A. Calibrating Reachable Workspace

The capability we focus on in this work is the reachable workspace of a robot. During the calibration, the robot generates multiple trajectories as demonstrations to show what it is capable of. We start from notations in this subsection. The proposed reachability-expressive motion planning, described in Section III-B, enables the robot to generate one trajectory showing its reachable workspace based on a simulated human belief that models what the human has already known about the robot. In Section III-C, we describe how human belief would be updated before a new trajectory can be generated. In Section III-E, we combine REMP and human belief update to generate multiple trajectories that calibrate the reachable workspace.

**Notation.** The robot’s ground-truth reachability is defined as  $f : \mathcal{X}_{ws} \rightarrow \{0, 1\}$ , i.e. whether a target position  $x$  in the workspace  $\mathcal{X}_{ws}$  is reachable by the end-effector according to the robot’s kinematic constraints. Meanwhile, we assume the human is maintaining a belief  $b_h^t : \mathcal{X}_{ws} \rightarrow [0, 1]$ , modeling how likely a target is reachable after observing robot trajectory  $\xi^t$  at time  $t \in \{1 \dots T\}$ . In addition, we define  $\mathcal{X}_{rs} \subseteq \mathcal{X}_{ws}$  as the robot’s reachable workspace.  $\phi_{ee} : \mathcal{Q} \rightarrow \mathcal{X}_{rs}$  is the

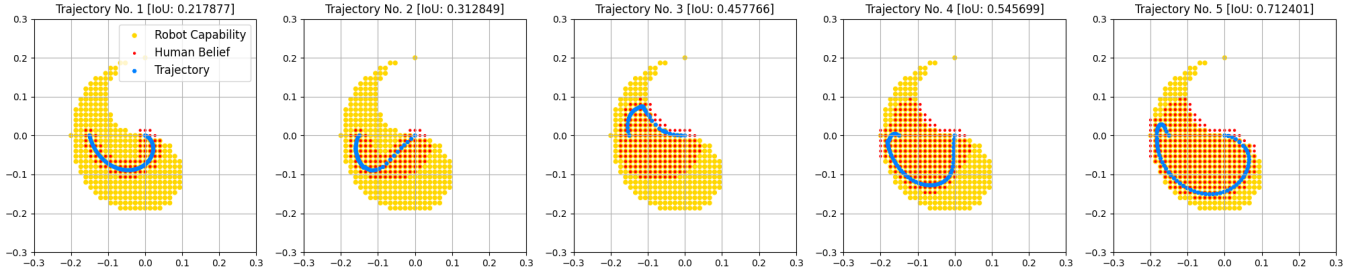


Fig. 2: Simulated human estimation of robot A’s reachability map, after observing each demonstration generated by trajectory optimization using  $c_b$ . Robot A is a 2-link arm with link lengths 0.1.

forward kinematic function of the end-effector, generating its position given a configuration.

### B. REMP: Reachability-Expressive Motion Planning

Expressing robot reachability is more than randomly moving the end-effector to somewhere in its reachable workspace. Our insight is that it is essential to understand what the human already knows or does not know about the robot, so that every demonstration can communicate as much information to the human as possible. We believe this can be formulated as an optimization problem: finding a new trajectory that would minimize the misalignment between the ground-truth reachability and human’s updated estimation. We capture the misalignment using a cost function  $c(\xi, b_h^t, f)$  and formulate the optimization problem as the following:

$$\begin{aligned} \xi^t = \arg \min_{\xi} \quad & c(\xi, b_h^t, f) + \frac{1}{\lambda} \sum_{i=1}^N \|\xi_{i+1} - \xi_i\|^2, \\ \text{subject to} \quad & \phi_{ee}(\xi_n) = x_r, \text{collision-free}(\xi). \end{aligned} \quad (1)$$

The first term is an expressiveness cost and the second term is a smoothness cost commonly seen in trajectory optimization. The trajectory at the  $t$ -th step is generated by minimizing the sum of the two costs, subjecting to a constraint that requires the end effector to reach a target position  $x_r$  at the end of the trajectory.

Assuming each point in the trajectory contributes to the cost independently, we can design the cost function based on a value  $v_i(b_h^t, f)$ , which represents the degree of alignment between human’s estimation and the robot’s ground-truth reachable workspace:

$$\begin{aligned} c_b(\xi, b_h^t, f) &= \alpha \sum_{i=1}^N v_i(\xi_i, b_h^t, f) \\ &= \alpha \sum_{i=1}^N e^{\beta(b_h^t(\phi_{ee}(\xi_i)) - f(\phi_{ee}(\xi_i)))} \end{aligned} \quad (2)$$

A small value  $v_i$  suggests that the human observer is underestimating the robot’s capability at  $\xi_i$ . In that case, we want to facilitate calibration by encouraging the robot to move to  $\xi_i$ . On the other hand, we would see a large  $v_i$  if the human is over-estimating the capability. In that case, it is beneficial for the robot to avoid reaching points near  $\xi_i$ . The hyperparameter  $\alpha$  and  $\beta$  control how aggressive the trajectory would be in expressing the capability. We call this cost function  $c_b$ , which captures human updated belief. Note

---

### Algorithm 1: Reachability-Expressive Motion Planning (REMP)

---

- 1 Given a target position  $x_r$  and a starting configuration  $\xi_1^t$ , human belief  $b_t$ ;
  - 2 Generate trajectory  $\xi^t$  based on  $b_h^t, \xi_1^t$ , Equation (1) ;
  - 3 Update human belief  $b_h^{t+1}$  using  $\xi^t$ , Equation (3) ;
  - 4 **return**  $b_h^{t+1}, \xi^t$
- 

that the intuition is if the observer previously underestimates the reachability of a point  $x$ ,  $b_h^t(x) - f(x)$  will be negative and give low cost for trajectories covering  $x$ . Hence, trajectories passing through underestimated points are more likely to be chosen. Trajectories including overestimated points, on the contrary, will have larger costs and are less likely to be selected.

### C. Human Belief Model

Our objective is to make people without any knowledge about robotics easily understand the true capacities of a robot. Thus, our human belief model attempts to capture what a novice user may think about a robot’s reachability after watching its trajectories.

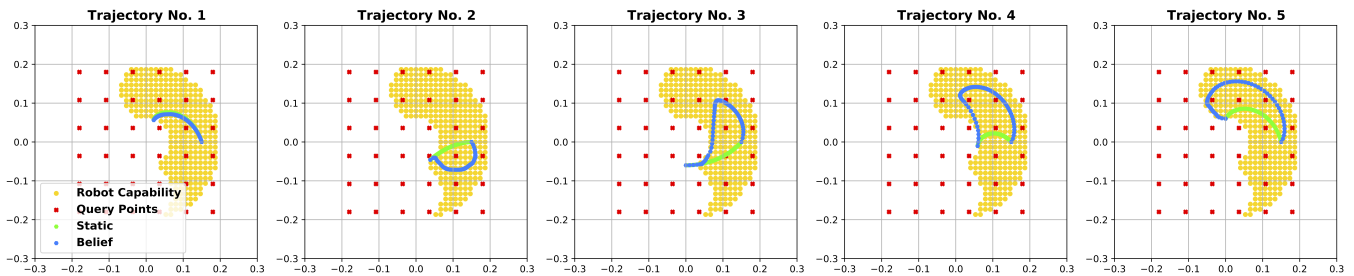
**Human belief update model.** We assume human updates its belief on an interested point  $x$  in the workspace after observing a new robot trajectory  $\xi$ . Intuitively, if a point is close to the visited positions in an observed trajectory, the human observer would consider it more likely to be reachable. We model the belief update process as an iterative Bayesian inference beginning from a uniform prior:

$$b_h^{t+1}(x) \propto b_h^t(x) p(\xi^t | x) \quad (3)$$

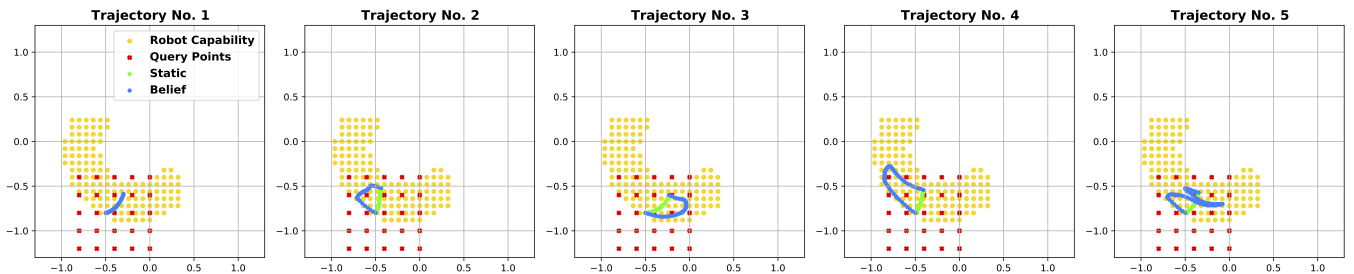
where  $d(\phi(\xi), x)$  captures the distance between the trajectory  $\xi$  and the interested position  $x$ , and  $p(\xi^t | x) = e^{-\gamma d(\phi(\xi^t), x)}$ . The hyperparameter  $\gamma$  defines how much the human extrapolates the observed trajectory to the points nearby: a large  $\gamma$  means that such extrapolation mainly happens to the point which is very close to the trajectory. In particular, we use the end-effector position  $\phi_{ee}$  as the feature, and compute the squared euclidean distance between the interested position and the closest end-effector position in the trajectory:

$$d(\phi(\xi), x) = \min_i \|\phi_{ee}(\xi_i) - x\|^2. \quad (4)$$

The design of our distance function is motivated by the fact that, given a trajectory, it is straightforward for users to



(a) Robot B is a 2-link arm with link lengths 0.13 and 0.07.



(b) Robot C is a PR2 robot. In this work, we consider the reachable workspace of its right arm.

Fig. 3: Visualization of the robot reachable workspace and the trajectories generated by using cost function  $c_b$  (belief) and  $c_s$  (static). (a) and (b) show the results for Robot B and Robot C respectively. It can be seen that the belief trajectories cover broader regions of the reachable workspace and new trajectories tend to visit areas that haven't been covered by their predecessors. The red dots, corresponding to Figure 5, represent the points we use to query the users in our experiments.

focus on the robot's end-effector which is central to the task, while trying to estimate its reachable workspace.

Due to the optimization tractability, we assume independence among voxels for belief update integrated in the motion planning. In the actual human belief simulation described in Section IV-C, we add local constraints to accommodate voxel proximity into human belief and use Monte-Carlo Markov Chain (MCMC) to compute the posterior given observations. Details of the human model can be found in Section S.III of the supplementary material.

**Static human model.** Our key intuition of the human model is that the human would update its belief of the robot's reachability after observing each trajectory. To test our intuition, we also design a baseline method that generates trajectories based on a fixed cost function, assuming an underlying uniform belief model  $\forall x, b_{static}(x) = b_0$ . The corresponding cost function under the assumption of a static human model is  $c_s$ , from which we can generate baseline trajectories based on Eq. (1). Note that this baseline would generate functional motions that solely aim to finish the physical task of reaching the target. We envision that in reality, users may also learn from these physical motions the robot's capability by interacting with the robot on some previous tasks, but such learning is not as efficient as the learning in a dedicated calibration phase.

#### D. Generating Reachability-Expressive Trajectories

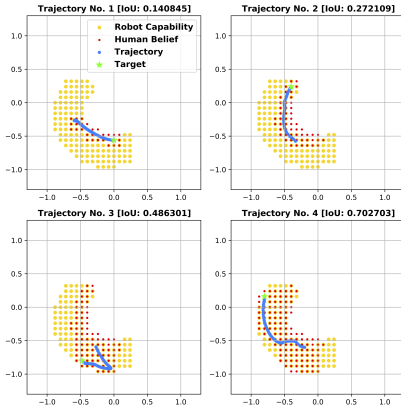
**Implementation.** We implemented our framework using TrajOpt [28] on two kinds of simulated robots in OpenRAVE [29], including a manipulator with 2 links and a PR2 robot. For the 2-link arm, we manipulated its joint limits and link lengths to allow it to have a variety of two-dimensional

reachable workspaces. These serve as testing cases for our framework, as we want to study how well the framework copes with reachable workspaces of different sizes and shapes. For the PR2 robot, we didn't do such manipulations since the goal here is to see how practical it is to apply the framework to real robot manipulators. Without loss of generality, we focus on the right arm of the PR2 robot. In practice, we can use grid search to find hyperparameters that generate trajectories to maximize the accuracy of reachability estimation in simulation, as described in Section IV-C.

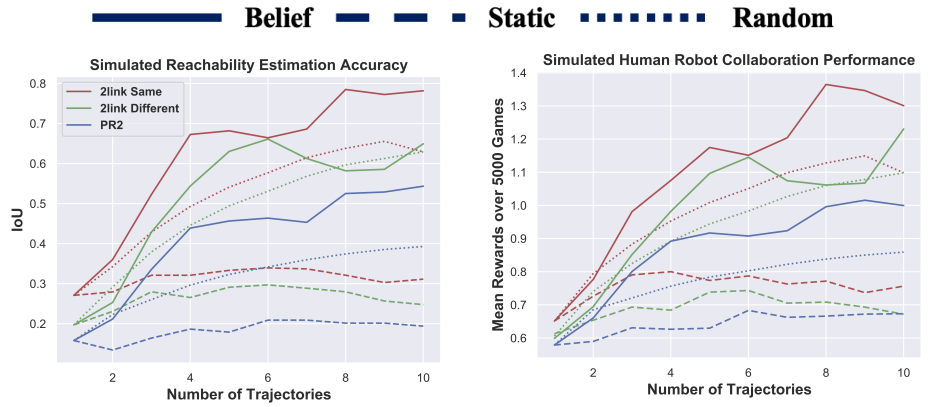
**Qualitative behaviors.** Figure 2, Figure 3a and Figure 3b show the trajectories generated by the cost functions  $c_b$  and  $c_s$  for the PR2 robot and a 2-link arm by iteratively running Algorithm 1 using the updated belief. As both  $c_b$  and  $c_s$  assume a uniform belief on the robot's reachable workspace at the beginning, the first trajectories generated by these cost functions are almost identical. Starting from the second trajectories, we find that the trajectories generated using  $c_b$  can cover a large part of the robot's reachable workspace. On the contrary, trajectories generated by  $c_s$  are very cost-sensitive and not able to traverse the workspace. Overall, It is clear that our method accommodates human belief at each time step and tries to traverse uncovered regions to better express the robot's reachability.

#### E. Planning for Start and Target Pairs via TAMP

We have shown how REMB can generate an expressive trajectory given a starting configuration and a target position.. For a better capability calibration, we also want to find an optimal sequence of starting configurations and target positions. As outlined in Algorithm 2, this could be achieved by Task and Motion Planning (TAMP) [30], [31], where the



(a) Trajectories generated by task and motion planning and the simulated reachability estimation given observed trajectories.



(b1) Simulated reachability estimation accuracy, measured by Intersection of Union between the human reachability estimation and the ground-truth. Higher value indicates better estimation. (b2) Simulated human-robot collaboration performance measured by averaged rewards acquired by the group. Every point on the curves for *random* is the mean of 100 trajectories.

Fig. 4: (a) Combining REMP with task planning, we can optimize the starting and target positions for better calibration. (b) Simulation results of reachability estimation and collaboration performance.

---

### Algorithm 2: TAMP for Calibration on Reachable Workspace

---

- 1 Given a set of target positions  $G = \{x_1, \dots, x_N\}$ , number of trajectories  $K$ , starting configuration set  $\Xi$ , initial human belief  $b_0$ ;
  - 2  $\forall x, b_h^0(x) \leftarrow b_0, \delta \leftarrow \infty$ ;
  - 3 **for**  $\kappa \in k - \text{combination}(\{1, \dots, N\})$  **do**
  - 4      $b_h \leftarrow b_h^0$ ;
  - 5     **for**  $t \leftarrow 1$  **to**  $K$  **do**
  - 6         // Greedily choose  $\xi_1^*$  from  $\Xi$
  - 6          $b_h, \zeta^t \leftarrow \text{REMP}(x_{\kappa(t)}, \xi_1^*, b_h)$ ;
  - 7          $D = \sum_{x \in \mathcal{X}_{ws}} |b_h(x) - f(x)|$ ;
  - 8         **if**  $\delta > D$  **then**
  - 9              $\delta \leftarrow D$ ;
  - 10             $\xi^{1:K} \leftarrow \zeta^{1:K}$ ;
  - 11 **return**  $\xi^{1:K}$
- 

plans of start and target pairs come from task planning (each pair is a task) and the trajectories for a given pair comes from motion planning (i.e., REMP). Figure 4a depicts an example to optimize 4 trajectories that start from different configurations and reach different targets. We assume a uniform prior for the human belief, and update the belief w.r.t. Eq. (3). By doing so, the planner captures what the human user has already known about the robot’s reachable workspace, and generates new trajectories that are most informative for the time being. Note that Algorithm 2 plans by enumerating all possible combinations, but any stochastic planning approaches can be used to accommodate resource constraints and task scalability.

## IV. APPLYING REMP TO HUMAN-ROBOT COLLABORATION

In this section, we discuss how to apply the proposed robot capability calibration algorithm to human-robot collaboration *after* the calibration.

### A. Collaborative Table Clearing

We design a human-robot collaboration task in a table clearing scenario, where some objects are scattered on a table and a robot can assist the human with the object collection. The human and the robot will take turns picking up the objects. In each time step, the human collects first and the robot collects one of the remaining objects. The human can reach all of the objects, while the robot can only reach a subset of them. To finish the table clearing as quickly as possible, the human and the robot need to split the work wisely, so that, in each round, the robot has objects to pick up, instead of witnessing the human working and cannot help.

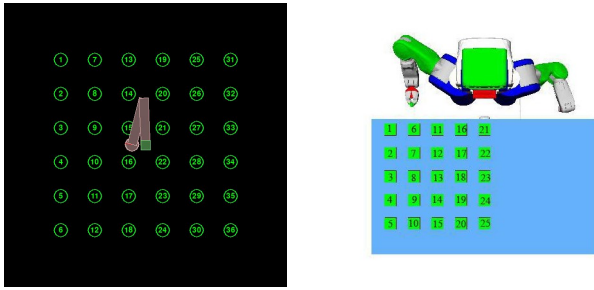
### B. Human and Robot Policy

After observing robot expressive demonstrations and updating the belief with Eq. (3), the human is assumed to act in an approximately optimal way with respect to the current estimation of the robot capability,  $b'_h$ . We use a Boltzmann noisily-rational human decision model [32], [33], assuming the human is more likely to help the robot with its unreachable objects based on the human’s current reachability estimation. Since we want to emphasize the effect of the calibration, we use a simple uniform robot policy in the simulation, i.e., it would randomly pick up objects it can reach, and do nothing if no objects are reachable. Details of the human and robot policy can be found in Section S.IV of the supplementary material.

### C. Simulation Results

Using the behavior model described in the previous sections, we simulated with 3 robots A, B and C with different configurations and capabilities: (i) Robot A is a 2-link arm where each link is of equal length, (ii) Robot B is a 2-link arm where the length of its first link (0.13) is larger than the length of the second (0.07), (iii) Robot C is a PR2 robot. The *belief* and *static* methods in the legend correspond to





(a) 36 points for 2-link arms.

(b) 25 points for PR2.

Fig. 5: To evaluate users’ estimation of the robot’s reachable workspace, we sample query points in the workspace and ask users to select points that they think the robot’s end effector can reach. These points correspond to the red dots in Figure 3a and Figure 3b.

the definition in Section III-C. In addition, we implemented a *random* baseline, where the robot continuously move its end-effector in its workspace randomly to demonstrate its reachability to the user.

Figure 4 shows the quantitative results of human-robot collaborative table clearing. In each table clearing task, we randomly sample 4 objects, exactly two of which are reachable by the robot. In each step, the human follows the policy described in the previous section and the reward is calculated with the number of objects picked up and the time penalty. Details of the game setup can be found in supplementary Section S.IV. As the results show, our method outperforms the baselines in terms of both reachability estimation and cooperation task rewards when using all three types of robots. There is a performance decrease when many trajectories are shown, which is attributed to the limited memory of our human model. The region covered by the previous trajectories became uncertain to the user after observing trajectories away from that region.

Figure 4 suggests that as the robot shows more demonstrations, the human has a better understanding of its capability and collaborates with it more effectively. However, without modeling human belief changes, the improvement is quite limited. In other words, it is hard for the human to get new information after the first few demonstrations. On the contrary, trajectories generated by our proposed REMP algorithm keep providing new information to the user. After 5 trajectories, the user’s estimation can cover most of the robot’s reachable workspace.

## V. USER STUDY

As we have demonstrated the effectiveness of REMP in simulation, we now turn to investigate how much it would help real humans work with robots in a user study.

### A. Experiment Design

**Manipulated variables.** Like simulations, we varied types of motion users observed in the user study, i.e., the *belief*, *static* and *random* methods defined in Section IV-C.

**Dependent variables.** Our dependent variables include how well participants can understand the robot’s capability and

TABLE I: Survey statements to evaluate reachability, predictability, reliability and trust toward robots.

- 
1. It is easy to tell where the robot’s hand can reach.
  2. The robot behaves in a predictable manner.
  3. I can rely on the robot to function properly despite its limited capability.
  4. I trust the robot.
- 

how such understanding can help them in the collaboration, as well as their self-reported perception of the robot:

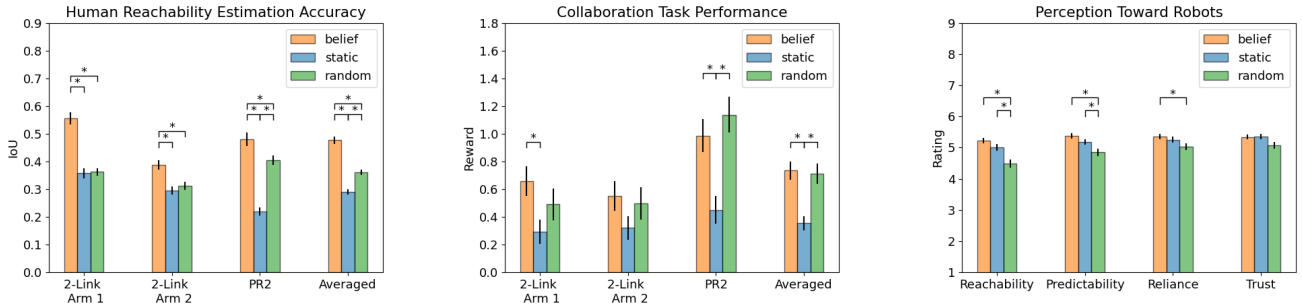
- To measure **capability understanding**, we ask users to choose positions that they think the robot can reach from a number of object queries, as shown in Figure 5. We record their selections and compare them with the ground-truth.
- For **collaboration task performance**, we use the accumulated reward of the team as a measure.
- To measure the **perception of the robot**, we ask participants to rate statements listed in Table I on a 7-point Likert scale labeled from “strongly agree” to “strongly disagree”, after they have finished interaction with a robot. Inspired by [34], the statements shown in Table I are designed to evaluate their subjective understanding of the robot in different aspects, including reachability, predictability, reliance and trust.

**Procedure.** After a brief introduction, each participant is asked to interact with three robots A, B and C in random order. The purpose is to see how robot’s physical configurations affect capability perception.

During the interaction with each robot, the participants would first go through a calibration phase before collaborating with the same robot on the table clearing task:

- In the calibration phase, the participant would be randomly assigned to an experiment group, and observe  $T=4$  demonstrations based on the condition. Each demonstration is a video showing a trajectory of a robot manipulator moving from a starting configuration to a target configuration. After seeing all the demonstrations, participants are asked to estimate the robot’s reachable workspace by choosing positions that they think the robot can reach from a number of object queries, as shown in Figure 5.
- In the collaboration phase, participants are asked to perform an online table clearing task together with the same robot they have just been calibrated. As discussed in Section IV-A, the task required the team to clear all four objects on the table. During each time step, the participant would pick up an object first before the robot makes its decision. The team would get rewarded based on how fast they take all the objects. Since two of the objects cannot be reached by the robot, to get the maximum accumulated reward (+2), the participant needs to pick up objects that cannot be reached by the robot. Failure to do so would result in the team getting a lower accumulated reward (0).

**Subject allocation.** We recruited 202 subjects (37% Female, median age 34) from Amazon Mechanical Turk. The study was approved by the UCLA Institutional Review Board. Demonstrations for one robot are between-subject: participants only saw demonstrations from one of the three conditions when interacting with a robot. The robot types are within-subject: participants interacted with all three robots.



(a) Intersection of Union between the human reachability estimation and the ground-truth. A higher value indicates better estimation.

(b) The human-robot team performance in the collaboration task. A higher value indicates a higher reward.

(c) Users' ratings toward the Likert statements in table I. A higher rating indicates higher confidence.

Fig. 6: User study results. Here we report means and standard errors.

**Hypotheses.** We hypothesized that our proposed capability calibration framework would enable participants to have a better understanding of the robot's capability, view the robot more positively and perform better in the collaboration task.

**H1:** Participants going through capability calibration in the *belief* condition would have a better understanding of the robot's capability, compared to those in the other conditions.

**H2:** Teams in the *belief* condition would perform better in the collaboration tasks than those in the other conditions.

**H3:** Participants in the *belief* condition would have a more positive perception of the robot, compared to those in the other conditions.

### B. Result and Analysis

**Capability understanding.** We first analyzed the accuracy of the user's estimation of the robot's reachable workspace, by computing the IoUs between their responses and the ground-truth. We performed a two-way ANOVA of the IoUs using the type of motion and robots as independent variables. As a result, we found a significant effect for the motion ( $F(2, 603) = 86.28, p < .001$ ). A post-hoc analysis with Tukey HSD revealed that all three conditions are different from each other, with *belief* significantly better than *static* ( $p < .001$ ) and *random* ( $p < .001$ ). This confirms our hypothesis **H1**. Figure 6a shows the accuracy of participants' reachability estimation w.r.t different robots. On average, *belief* performs 65% better than *static* and 32% better than *random*. Compared to the simulation results in Figure 4, the user study result follows relatively the same order for different conditions.

**Task performance.** We also analyzed the collaboration task performance. A two-way ANOVA indicates that there is a statistically significant effect of the accumulated rewards between conditions ( $F(2, 603) = 12.21, p < .001$ ). The post-hoc showed a significant difference between *belief* and *static*. This partially supports our hypothesis **H2**. We didn't observe a significant difference between *belief* and *random*. Figure 6b shows the task performance for different robots in three conditions. The Pearson correlation coefficient between reachability estimation and collaboration performance is  $r = .203$  with p-value smaller than 0.001, indicating a positive correlation. This validates that calibrating perceived robot ca-

pability benefits the collaboration performance. Surprisingly, users in the *random* condition have a slightly higher reward when collaborating with the PR2 robot compared with those in *static*, even if their reachability estimation is less accurate, although the difference is not significant. This is probably due to the stochastic nature of the *random* baseline and specific object locations in our collaboration task.

**Perception of robots.** Finally, we analyzed participants' perception toward robots. Running a two-way ANOVA, we found significant effects for three measures, including reachability ( $F(2, 603) = 11.60, p < .001$ ), predictability ( $F(2, 603) = 7.84, p < .001$ ), reliance ( $F(2, 603) = 3.12, p = .045$ ). The post-hoc revealed significant difference between *belief* and *random* for reachability ( $p < .001$ ), predictability ( $p < .001$ ) and reliance ( $p = .038$ ), confirming **H3**. Overall, users tended to prefer *belief* over *static*, and *static* over *random*. This is not surprising for predictability, since the robot arms moved in an unpredictable manner in *random*. However, it is unexpected for reachability, considering the fact that users are actually better at predicting robots' reachability in *random* than in *static*. The Pearson correlation coefficient between reachability rating and prediction accuracy is  $r = .109$ , indicating a weak positive correlation. Similarly, we observe a very weak correlation  $r = .019$  between self-reported reliance and users' actual ability to rely on the robot during collaboration. This suggests that there may be a discrepancy between the users' self-reported capability understanding and what they actually know about the robot.

In summary, we found that users in the *belief* condition had the most accurate estimation of the robots' capability, and reported the robots in this condition as the most reliable, the most predictable, and the easiest to understand among all three conditions. Moreover, users working with the *belief* robots achieved a higher reward than those working with the *static* robots did. These objective and subjective results together suggest that our approach has an overall advantage for improving humans' understanding of robots as well as the quality of collaboration over the baselines.

## VI. CONCLUSION

In this paper, we propose an expressive robot motion planning algorithm, REMP, which can efficiently calibrate

the robot's reachability with the user. We integrate human belief into the cost function of our optimization problem and accommodate human belief update in our trajectory generation. We validated the advantage of our algorithm with simulation and an online user study using three robots.

Note that we mainly focused on the robot's spatial reachability in this work, as reaching is one of the most basic tasks in human-robot interaction. Understanding reachability would greatly help users understand more complex robot capacities. Nevertheless, REMP can be combined with other approaches to facilitate the understanding of a robot's different capabilities. We view our work as a successful first step towards a more general capability calibration setting. For future work, we intend to explore online capability calibration during the collaboration process. Another promising direction is to develop multi-modal demonstrations, including gestures, gazes and verbal communication [35], [36], [37].

## REFERENCES

- [1] T. A. Stoffregen, K. M. Gorday, Y.-Y. Sheng, and S. B. Flynn, "Perceiving affordances for another person's actions." *Journal of Experimental Psychology: Human Perception and Performance*, vol. 25, no. 1, p. 120, 1999.
- [2] A. Powers and S. Kiesler, "The advisor robot: tracing people's mental model from a robot's physical attributes," in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, 2006, pp. 218–225.
- [3] S. R. Fussell, S. Kiesler, L. D. Setlock, and V. Yew, "How people anthropomorphize robots," in *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2008, pp. 145–152.
- [4] B. Hayes and J. A. Shah, "Improving robot controller transparency through autonomous policy explanation," in *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2017, pp. 303–312.
- [5] P. Stone, G. A. Kaminka, S. Kraus, J. S. Rosenschein, *et al.*, "Ad hoc autonomous agent teams: Collaboration without pre-coordination." in *AAAI*, 2010, p. 6.
- [6] S. V. Albrecht and P. Stone, "Reasoning about hypothetical agent behaviours and their parameters," in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS '17. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2017, p. 547–555.
- [7] E. Cha, A. D. Dragan, and S. S. Srinivasa, "Perceived robot capability," in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2015, pp. 541–548.
- [8] Y. Zhang, S. Sreedharan, and S. Kambhampati, "Capability models and their applications in planning," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, 2015, pp. 1151–1159.
- [9] S. Nikolaidis, S. Nath, A. D. Procaccia, and S. Srinivasa, "Game-theoretic modeling of human adaptation in human-robot collaboration," in *Proceedings of the 2017 ACM/IEEE international conference on human-robot interaction*, 2017, pp. 323–331.
- [10] B. M. Muir, "Trust in automation: Part i. theoretical issues in the study of trust and human intervention in automated systems," *Ergonomics*, vol. 37, no. 11, pp. 1905–1922, 1994.
- [11] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," *Human factors*, vol. 46, no. 1, pp. 50–80, 2004.
- [12] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Planning with trust for human-robot collaboration," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 307–315.
- [13] S. H. Huang, K. Bhatia, P. Abbeel, and A. D. Dragan, "Establishing appropriate trust via critical states," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3929–3936.
- [14] Y. Xie, I. P. Bodala, D. C. Ong, D. Hsu, and H. Soh, "Robot capability and intention in trust-based decisions across tasks," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2019, pp. 39–47.
- [15] J. Lee, J. Fong, B. C. Kok, and H. Soh, "Getting to know one another: Calibrating intent, capabilities and trust for human-robot collaboration," *arXiv preprint arXiv:2008.00699*, 2020.
- [16] T. Hellström and S. Bensch, "Understandable robots-what, why, and how," *Paladyn, Journal of Behavioral Robotics*, vol. 9, no. 1, pp. 110–123, 2018.
- [17] G. Venture and D. Kulić, "Robot expressive motions: a survey of generation and evaluation methods," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 8, no. 4, pp. 1–17, 2019.
- [18] A. D. Dragan, S. Bauman, J. Forlizzi, and S. S. Srinivasa, "Effects of robot motion on human-robot collaboration," in *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2015, pp. 51–58.
- [19] D. Szafir, B. Mutlu, and T. Fong, "Communication of intent in assistive free flyers," in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, 2014, pp. 358–365.
- [20] E. Rosen, D. Whitney, E. Phillips, G. Chien, J. Tompkin, G. Konidaris, and S. Tellex, "Communicating robot arm motion intent through mixed reality head-mounted displays," in *Robotics Research*. Springer, 2020, pp. 301–316.
- [21] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006, available at <http://planning.cs.uiuc.edu/>.
- [22] A. Dragan and S. Srinivasa, "Generating legible motion," 2013.
- [23] F. Stulp, J. Grizou, B. Busch, and M. Lopes, "Facilitating intention prediction for humans by optimizing robot motions," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 1249–1255.
- [24] M. L. Felis, K. Mombaur, and A. Berthoz, "An optimal control approach to reconstruct human gait dynamics from kinematic data," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2015, pp. 1044–1051.
- [25] A. D. Dragan, R. M. Holladay, and S. S. Srinivasa, "An analysis of deceptive robot motion." in *Robotics: science and systems*. Citeseer, 2014, p. 10.
- [26] C. K. Liu, A. Hertzmann, and Z. Popović, "Learning physics-based motion style with nonlinear inverse optimization," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 1071–1081, 2005.
- [27] M. Kwon, S. H. Huang, and A. D. Dragan, "Expressing robot incapability," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 87–95.
- [28] J. Schulman, J. Ho, A. X. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization." in *Robotics: science and systems*, vol. 9, no. 1. Citeseer, 2013, pp. 1–10.
- [29] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, USA, 2010.
- [30] L. P. Kaelbling and T. Lozano-Pérez, "Hierarchical task and motion planning in the now," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1470–1477.
- [31] L. P. Kaelbling and T. Lozano-Perez, "Integrated task and motion planning in belief space," *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1194–1227, 2013.
- [32] O. Morgenstern and J. Von Neumann, *Theory of games and economic behavior*. Princeton university press, 1953.
- [33] C. L. Baker, J. B. Tenenbaum, and R. R. Saxe, "Goal inference as inverse planning," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 29, no. 29, 2007.
- [34] M. Madsen and S. Gregor, "Measuring human-computer trust," in *11th Australasian conference on information systems*, vol. 53. Citeseer, 2000, pp. 6–8.
- [35] C. Breazeal, C. D. Kidd, A. L. Thomaz, G. Hoffman, and M. Berlin, "Effects of nonverbal communication on efficiency and robustness in human-robot teamwork," in *2005 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2005, pp. 708–713.
- [36] A. Thomaz, G. Hoffman, and M. Cakmak, "Computational human-robot interaction," *Foundations and Trends in Robotics*, vol. 4, no. 2-3, pp. 105–223, 2016.
- [37] H. Romat, M.-A. Williams, X. Wang, B. Johnston, and H. Bard, "Natural human-robot interaction using social cues," in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2016, pp. 503–504.