Learning Human Ergonomic Preferences for Handovers

Aaron Bestick, Ravi Pandya, Ruzena Bajcsy, and Anca D. Dragan

Abstract— **Our goal is for people to be physically comfortable when taking objects from robots. This puts a burden on the robot to hand over the object in such a way that a person can easily reach it, without needing to strain or twist their arm – a way that is conducive to *ergonomic* human grasping configurations. To achieve this, the robot needs to understand what makes a configuration more or less ergonomic to the person, i.e. their ergonomic cost function. In this work, we formulate learning a person's ergonomic cost as an online estimation problem. The robot can implicitly make queries to the person by handing them objects in different configurations, and gets observations in response about the way they choose to take the object. We compare the performance of both passive and active approaches for solving this problem in simulation, as well as in an in-person user study.**

I. Introduction

When a robot hands over an object to a person, it has a choice to make – it chooses which specific grasping configuration to use. When the person then takes that object, they too have same choice to make. But depending on what the robot chose, their options might be limited, and they might be forced to twist their arm in an uncomfortable way just to be able to reach the object.

Our goal is to enable robots to choose handover configurations that result in *comfortable* options for the person who is taking the object. But to do that, the robot needs to know what "comfortable" actually means.

In this work, we capture comfort level via an *ergonomic cost*, which maps each human grasping configuration to a scalar value represents how comfortable or uncomfortable it is. If the robot had access to this cost function, it could use it to plan its handovers to explicitly make low cost human configurations for taking the object feasible. More interestingly, it could use it as a *predictive model* for how the person will take the object (assuming lower cost configurations are more likely). Having such a model empowers the robot to anticipate human action, and even influence people towards grasps that better suit their ultimate goal for the object [1].

In our previous work, we have simply written down what seemed like a reasonable ergonomic cost function and handed it to the robot [1], [2]. Other works have done the same [3]–[5]. But there was always something worrisome about this: how do we know this cost is any good? Even for the average person, we might have gotten it wrong. And further, not everyone is the average person. We expect to see individual variation in ergonomic preferences, e.g. based on which muscles in a person's arm happen to be stronger. These differences could be even more pronounced for people whose motion is restricted by age, injury, or disability.

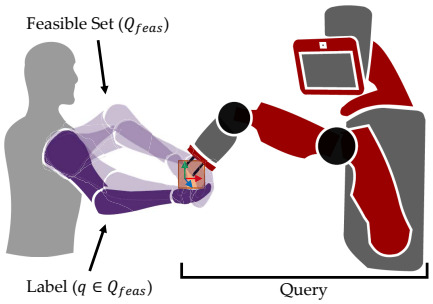

Fig. 1: The robot learns the human ergonomic preferences by selecting a handover configuration for an object (query), and using the way the person chooses to take the object (label) as an observation about their hidden ergonomic parameters.

In this work, we turn to *learning* a cost function, rather than assuming one. The parameters of the cost are a hidden part of the state. Every choice the person makes for a grasp configuration is an *observation* that depends on these parameters, so we can update the robot's belief via Bayesian inference.

A key aspect of this learning problem is that observations do not happen in isolation from the robot. The robot gets to influence them by selecting its own grasping configuration at which it offers the object – an *implicit query* that the robot makes to the person (Fig. 1). This query induces a *feasible set* of human grasp configurations for taking the object, which results in a *label*, i.e. the person's choice.

Therefore, which queries the robot makes affect its performance. Some queries elicit more information and help the robot learn faster, but sacrifice some human comfort during learning in order to do so.

We thus explore and contrast two natural approaches for selecting queries. In the *passive* approach, we use the principle of separating estimation and control, and the robot always selects the query that leads to the most comfortable configuration for the person according to its current belief about their ergonomic cost. In the *active* approach, the robot selects queries that lead to the highest expected information gain.[1]

What we contribute is a formulation of the ergonomic cost learning problem as online estimation via implicit, physical robot queries, and an in-depth analysis of the advantages of disadvantages of the two methods, in-

The authors are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA.

This work was supported by NSF NRI award #1427260 and a Siemens Fellowship.

[1]Note that solving the problem as a POMDP is computationally prohibitive still, but would lead to optimally trading off between exploration and exploitation, i.e. a hybrid between the active and passive approaches.

cluding how the menu of objects and queries available to the robot impacts their performance.

We do experiments in both simulation (with an easy to visualize 2DOF arm and a more difficult, but realistic 7DOF arm), as well as a user study. We find that online learning of ergonomic cost is feasible, that active learning can be faster, but that this depends on the kinds of objects it has available, and that it comes at small, but non-trivial comfort cost during learning.

II. Related Work

Learning from Implicit Queries in Robotics. Many robotic learning systems solicit explicit feedback from humans to optimize their own actions. This feedback can include binary preferences for one trajectory over another [6], [7], rankings of grasp quality on a continuous scale [8], or class labels applied to images the robot encounters [9].

Intuitively, some queries are more informative than others. The idea of *actively* selecting the most informative queries to present to the human expert has been widely studied and applied [10], [11]. These explicit, actively selected queries include binary classification [12]–[17], ranking [18], [19], and labeling [20]. Queries in robot learning can be particularly expensive in both time and effort, often requiring physical motion of the robot. This makes active learning approaches, which minimize the required number of queries to learn a model of a given accuracy, attractive here as well [7]–[9], [21], [22].

Sometimes an explicit human response to a query isn't required at all. In these *implicit* learning tasks, the robot simply takes an action, observes the human's response, and uses this response to infer their label, under the assumption that the response represents an optimal execution of a policy based on their true preferences. Recent work has used this framework to learn the characteristics of human drivers in order to predict their future trajectories [23]. Implicit learning can be more intuitive in situations where the human doesn't have a conscious, explicit ordering over possible actions, but nonetheless exhibits a preference for some over others, from which this ordering can be reconstructed [10]. We use implicit human feedback. Our approach is conceptually similar to Inverse Reinforcement Learning (IRL) [24], and is essentially Bayesian IRL [25], with the adjustment that, since our belief space is simply the configuration space, maintaining a particle representation of the full belief is tractable.

Ergonomic Handovers. Ideally, a robot would allow you the most comfortable configuration options possible. Existing work focuses on selecting object handover positions [2]–[4], [26] or poses [1], [27]–[32] which accomplish this.

Previous work uses a wide variety of metrics to compare the feasible human grasp configurations allowed by a robot's chosen handover. Examples include predefined ergonomic costs [1], [2], [4], proximity of the object to the human's body [3], [4], visibility of the object [3], [4], manipulability at a given grasp configuration [30], [31] or simply the total number of grasp configurations available to the human given a particular handover pose [9], [32].

III. Learning Ergonomic Cost

We pose the ergonomic cost learning problem as one of learning from implicit queries, which yield human demonstrations. Suppose a robot hands an object to a person using a grasp g_R and at a pose T_{hand}. We'll call the tuple (g_R, T_{hand}) the robot's *action*, i.e the *query*. This action, in combination with the kinematic structure of the human's arm and the object's feasible grasp regions, induces a set of feasible human grasp configurations Q_{feas}, which we'll sometimes write as a function of the robot's action $Q_{feas}(g_R, T_{hand})$. When the human is presented with this robot action, they choose a single grasp configuration $q_H \in Q_{feas}$, which is our *observation* or *label*. Given multiple action-observation pairs, we train a model of $P(q_H|Q_{feas})$, which gives the probability that the human will choose a particular $q_H \in Q_{feas}$ when the robot takes an action (g_R, T_{hand}) that results in the feasible set Q_{feas}.

A. Feasible Set Computation

We represent the set of all feasible human grasps G_H on the object being handed off as a Task Space Region (TSR) [33], where $G_H \subset SE(3)$. This TSR is discretized to produce a finite set of all feasible grasps $G_H \triangleq \{g_{H1}, \ldots, g_{HN}\}$.

For each grasp $g_{Hi} \in G_H$, we can compute a set of inverse kinematics (IK) solutions $Q_{feas,i}$ which allow the human to reach the specified grasp. We collect all the IK solutions for all object grasps into a set $Q_{feas} = \bigcup_{i=0}^{N} Q_{feas,i}$. As mentioned above, we'll abstract away this process by simply writing the feasible set as a function of the robot's action $Q_{feas}(g_R, T_{hand})$.

B. Probabilistic Model

Given a set Q_{feas} of feasible human arm configurations, we seek a model $P(q_H|Q_{feas})$, which gives the probability that a given person will select any of the individual grasps $q_H \in Q_{feas}$.

We structure this model with an assumption: we assume that the human is approximately rational, and that the likelihood of them selecting a configuration decreases exponentially as the *ergonomic cost* of that configuration increases:

$$P(q_H) \propto e^{-\alpha C_{ergo}(q_H)}. \tag{1}$$

In general, C_{ergo} can be any function which maps configurations to scalar costs. The methods we present could estimate any parametrization of such a function. Nonetheless, to experiment with the methods, we must commit to a parametrization. We choose an intuitive one: we parametrize cost as squared distance from some *neutral* arm configuration q_H^* that captures the most comfortable configuration for the person, but measure distance with respect to an inner product W which is not necessarily Euclidean:

$$C_{ergo}(q_H, \lambda) \triangleq (q_H - q_H^*)^\intercal W (q_H - q_H^*). \tag{2}$$

In our experiments, we assume for simplicity (in order to lower the number of parameters we need to estimate) a diagonal *weight* matrix $W = diag(w)$. This captures preferences in moving certain joints away from the neutral configurations more than others. We collect these parameters into a single parameter vector $\lambda \triangleq [q_H^*, w]$ which the robot needs to estimate by interacting with the human.

Given a known λ and a set of feasible arm configurations Q_{feas}, the resulting probability of the human choosing a given configuration q_H takes the form of a Boltzmann distribution:

$$P(q_H|Q_{feas}, \lambda) = \frac{e^{-C_{ergo}(q_H;\lambda)}}{\sum_{\widehat{q_H} \in Q_{feas}} e^{-C_{ergo}(\widehat{q_H};\lambda)}} \tag{3}$$

Importantly, this distribution is normalized over all other configurations that the person *could have chosen*.

C. Bayesian Belief Updates

We start with a generic, uncertain belief $P(\lambda)_0$ over the human's cost function parameters. Our goal is to iteratively refine this belief as we collect more training data. Our technique could be used across a population to learn an average ergonomic cost, or for an individual to personalize the ergonomic cost to their preferences.

Because the beliefs over possible cost function parameters produced by our training examples are potentially quite complex, we use a particle filter to perform belief updates. This enables us to represent arbitrarily complex beliefs without being constrained by the form of a parameterized distribution.

To perform a single belief update, the robot chooses an action (g_R, T_{hand}), which induces a set of feasible human configurations $Q_{feas}(g_R, T_{hand})$. We then observe the human's choice q_H. Our complete training example is then the tuple (q_H, Q_{feas}). We update our prior belief $P(\lambda)$ with the training example to give a posterior $P(\lambda|q_H, Q_{feas})$ by applying Bayes' Rule:

$$P(\lambda|q_H, Q_{feas}) \propto P(q_H|Q_{feas}, \lambda)P(\lambda) \tag{4}$$

Note that the likelihood function $P(q_H|Q_{feas}, \lambda)$ is equal to the Boltzmann likelihood in (3).

We represent the prior belief at each step as a set of N particles $\Lambda = \{\widetilde{\lambda_1}, \ldots, \widetilde{\lambda_N}\}$ and corresponding weights $\Omega^\lambda = \{\omega_1^\lambda, \ldots, \omega_I^\lambda\}$. Given a new training sample (q_h, Q_{feas}), we compute the new particle weights $\Omega^{\lambda'}$ as:

$$\omega_i^{\lambda'} = \omega_i^\lambda \left(\frac{e^{-C_{ergo}(q_H;\widetilde{\lambda_i})}}{\sum_{\widehat{q_H} \in Q_{feas}} e^{-C_{ergo}(\widehat{q_H};\widetilde{\lambda_i})}} \right) \tag{5}$$

Particles are then resampled to produce final Λ' and $\Omega^{\lambda'}$ sets in which all the particle weights are identical. Note that the normalization constant in (5) is different for each value of i, in contrast to a more typical particle filter implementation where the normalization constant is the same for every particle.

D. Active Query Selection

Suppose we have a menu of N possible robot actions, which induce feasible sets $\{Q_{feas,1}, \ldots, Q_{feas,N}\}$, and our current belief about the cost function parameters is $P(\lambda)$. When we present our human with any of the feasible sets, we'll observe a training datapoint (q_H, Q_{feas}). Intuitively though, some queries elicit human responses which are more informative than others. The active method actively seeks out these queries to present to the human.

If we assume the belief state $P(\lambda)$ is represented using a set of particles p_i with corresponding weights k_i, we can compute the Shannon entropy of the belief state $H(P(\lambda))$ by discretizing the belief space into M discrete beliefs using a grid, then applying the standard definition of entropy:

$$H(P(\lambda)) \triangleq \sum_{i=1}^{M} P(\lambda_i) \log P(\lambda_i) \tag{6}$$

We can then compute the expected change in entropy $E[\Delta H(Q_{feas})]$ of our belief (i.e. the information gain) given that we chose a particular query Q_{feas}:

$$E[\Delta H(Q_{feas})] = H(P(\lambda)) - E_{\widehat{\lambda} \sim P(\lambda)} \left[E_{\widehat{q_H} \sim P(q_H|\widehat{\lambda})} \left[H\left(P(\lambda \mid \widehat{q_H}, Q_{feas}) \right) \right] \right] \tag{7}$$

To make the calculation more efficient, we approximate by substituting the maximum likelihood value of q_H for the inner expectation:

$$E[\Delta H(Q_{feas})] = H(P(\lambda)) - E_{\widehat{\lambda} \sim P(\lambda)} \left[H\left(P\Big(\lambda \mid \arg\max_{q_H} P(q_H \mid \widehat{\lambda}), Q_{feas}\Big) \right) \right] \tag{8}$$

After computing the expected information gain for each feasible set in the menu of possibilities, we select the one which produces the greatest expected information gain:

$$Q_{feas}^{active} = \arg\max_{Q_{feas}} E[\Delta H(Q_{feas})] \tag{9}$$

E. Passive Query Selection

The active query selection algorithm in Sec. III-D selects robot actions to maximize information gain from each query. In contrast, a passive approach selects the action that that minimizes expected ergonomic cost to the person at each step, i.e. the expected cost of the grasp configuration the human is most likely to pick, given the robot's current belief about their ergonomic cost parameters. The passive approach still gains information at ever step, but merely as a side effect. This corresponds to separating estimation from control or hindsight optimization [34], [35], i.e. always planning with the current belief as if the ground truth will be revealed at the next step, and updating the belief at every step based on the new observation.

For a given feasible set Q_{feas}, the expected human ergonomic is:

$$E[C_{ergo}(Q_{feas})] = E_{\widehat{\lambda}\sim P(\lambda)}\left[\min_{\widehat{q_H}\in Q_{feas}} C_{ergo}(\widehat{q_H};\widehat{\lambda})\right] \quad (10)$$

We then select the feasible set which minimizes the expected ergonomic cost incurred by the human:

$$Q_{feas}^{passive} = \arg\min_{Q_{feas}} E[C_{ergo}(Q_{feas})] \quad (11)$$

IV. Experimental Design

We evaluate our active learning algorithm in three separate scenarios: 1) a simulated 2 DoF planar arm with planar objects, 2) a simulated 7 DoF human arm with real, 3D objects, and 3) a human user study on a real life object handover task. In the first two simulated scenarios, we test the algorithm's ability to recover the known ground truth parameters of our simulated human's ergonomic cost function. In the user study, we evaluate the accuracy with which our learned model can predict the human's grasp in future handoffs.

We test the learning of both the neutral configuration q_H^* and the joint weights w.

Manipulated Variables: In each scenario, we manipulated the query selection algorithm. In the simulation scenarios, we compare *active*, *passive*, and *random* algorithms (i.e. choose an object and a configuration at random). In the human user study, we compare *active* with *passive*. We also manipulated the number of queries allowed, from 1 to 5 for the 2 DoF simulations and the user study, and from 1 to 10 for the 7 DoF simulations.

Objective Measures: We measure *accuracy* and *training cost* – accuracy is most important if we think of this as a training period with the robot, to be the followed by a lifelong interaction; cost is very important if we think of this as a continuous interaction, in which the robot needs to learn without making the person uncomfortable.

We measure the accuracy of the each algorithm's belief $P(\lambda)$ over the model parameters, at every time step (i.e. # of iterations) using three objective measures:

- $P(\lambda^*)$: Probability density of the belief at the known ground truth parameter value
- $||\arg\max P(\lambda) - \lambda^*||$: Euclidean distance between the ground truth parameter vector and the mode of our belief
- $\log \mathcal{L}(\arg\max P(\lambda) \mid q_H^{test}, Q_{feas}^{test})$: The log likelihood of the mode of our belief, with respect to a separate test dataset (i.e. test set log likelihood)

The first two measures require us to know the ground truth ergonomic cost function parameters λ^*, so we evaluate them only for our two simulated experiments.

We also measure training cost:

- $E[C_{ergo}(q_H)]$: The expected ergonomic cost of a query with respect to the ground truth cost

Subjective Measures: In the user study, we also care about what experience the users prefer, especially in light of the fact that the difference between passive and active is in optimizing information gain vs. (greedily) optimizing user comfort. We ask 4 Likert scale questions and a forced choice (Table I).

Hypothesis: Because the active algorithm is designed to quickly reduce the entropy of the belief, we hypothesize that it will learn the parameters faster (i.e. have higher accuracy for the same number of queries, especially when this number is low). However, we also expect that the amount of improvement will depend on the set of available queries, and that the passive algorithm will incur lower true ergonomic cost during training.

V. Analysis for a Planar Two DoF Arm

Overall Analysis. We used a simulated planar 2 DoF arm. To create our training set, we randomly selected eight task space object shapes (as shown in Figures 5 and 6 and enumerated all the IK solutions for a discretized version of the object to yield a menu of eight feasible sets $\{Q_{feas,1}, \ldots, Q_{feas,8}\}$. The three algorithms – active, passive, and random – selected queries from this menu.

We repeated the training process a total of 50 times. For each trial, we selected a random menu of eight queries to make available to the robot. We selected a random ground truth value q_H^* or w to attempt to learn for each trial.

Each of the objective measures was evaluated after every iteration. The test set likelihood was evaluated on a set of 300 randomly generated objects. Fig. 2 shows the results.

The active learning algorithm produced faster learning on all three objective measures, for both the neutral configuration q^* and the weights w. This difference was particularly large between the passive and active algorithms when learning the weights, where the passive algorithm failed to converge toward the correct belief even after many iterations.

On the other hand, active does suffer a loss in true cost, especially after several queries when passive has converged to a decent estimate. How important this is depends on the use case: it is perhaps alright to suffer an initial loss in order to converge to a better cost that the robot will use for many additional interactions; however, users might not tolerate this well in certain tasks.

Artificial Feasible Sets. These initial results suggest that both methods are better than random query selection, with the active method learning faster and the passive method incurring less regret.

Next, we investigate what exactly causes the active learning algorithm to consistently outperform in accuracy. To explore this question, let's examine the active algorithm's decisions on the two simple feasible sets from Fig. 3. As the figure makes obvious, the feasible set Q_{feas} of configurations available to the human has a huge impact on the resulting likelihood function $P(q_H^* \mid q_H, Q_{feas})$, even if the human's chosen configuration q_H is similar or identical. When the active query

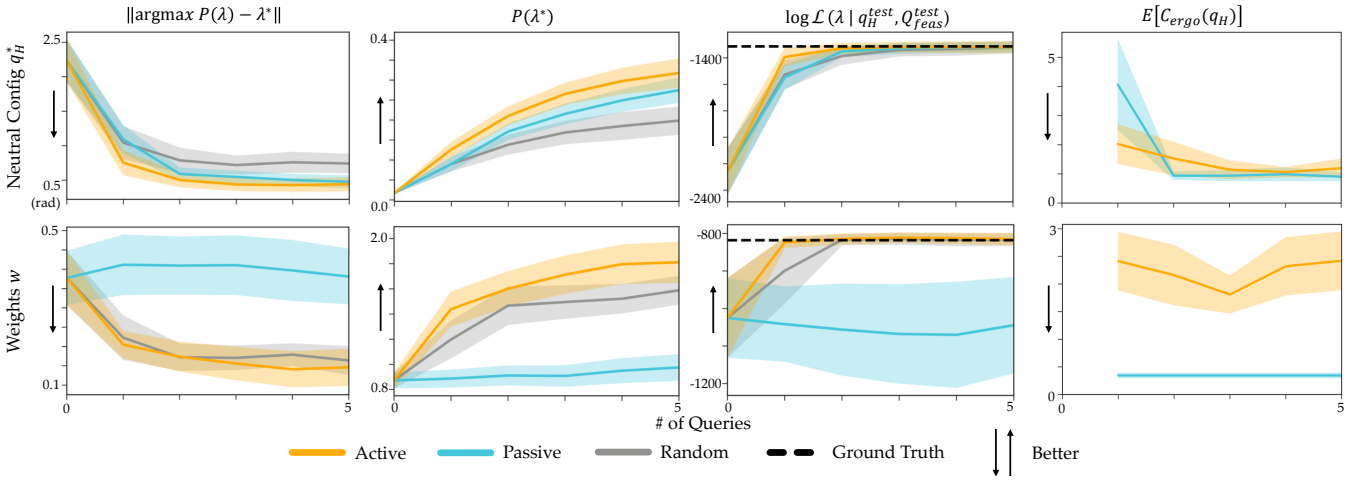

Fig. 2: The values of each objective measure vs. the number of training handovers, shown for both neutral configuration (q^*) and joint weights (w) learning. The active query selection algorithm yields faster learning than the passive and random algorithms for all three of the objective measures. This difference is particularly marked for learning of the joint weights w (bottom), where the passive algorithm fails to converge towards an accurate belief even after many iterations.

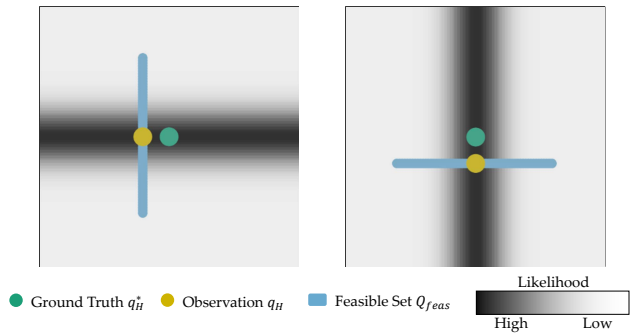

Fig. 3: Synthetic feasible sets shown with the ground truth optimal configuration (green), the simulated human's choice (yellow), and the resulting likelihood function (grey). Note how the shape of the feasible set of configurations completely changes the resulting likelihood function, even as the human's chosen configuration is relatively similar in both examples.

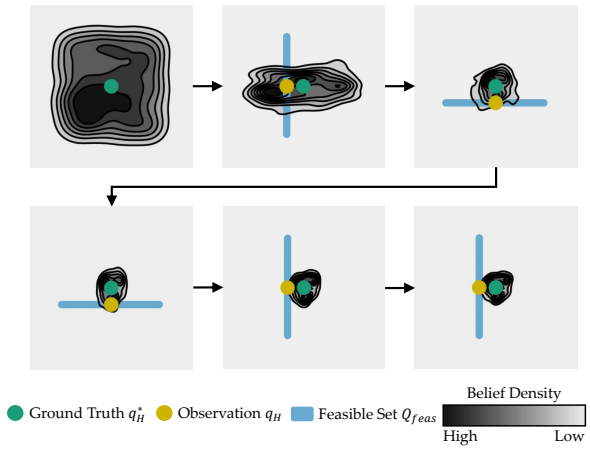

Fig. 4: Sequence of belief updates when training sets are selected using the active learning algorithm from the set of two possibilities above. Note how the active learning algorithm alternates between the two queries so as to remove the maximum amount of probability mass from the belief at each iteration. Because our model assumes humans are noisy, repeated iterations with the same query will continue to refine our belief past it's initial value after the first iteration.

selection algorithm is applied to a menu of queries containing just these two feasible sets, it makes the decisions shown in Fig. 4. The probability density of the resulting belief after each iteration is shown in grey. Notice how the active algorithm's desire to reduce the belief's entropy causes it to alternate between the two queries, selecting whichever one will remove the greatest amount of probability mass from the current belief at each iteration.

Object-Derived Feasible Sets. With this insight about the impact of feasible set shape and size on the resulting likelihood used to update our belief, let's return to our original two DoF planar arm scenario. Figures 5 and 6 show examples of the randomly generated task space objects included in the training sets used to generate the experimental data shown in Fig. 2 (bottom row), and the resulting configuration space feasible sets, ground truth q_H^*'s, simulated human's chosen q_H's, and the likelihood functions resulting from each query and observation.

The highly nonlinear nature of the inverse kinematics map means that seemingly similar task space objects can create completely different configuration space feasible sets. In addition, both the shape of the task space objects (Fig. 5), and their pose T_{hand} (Fig. 6) affect the resulting likelihood function. We can see that, in general, configuration space feasible sets composed of multiple, widely separated disjoint regions produce likelihood functions with sharp gradients which quickly eliminate large pieces of the belief.

What actually happens when we present the active and passive query selection algorithms with a menu of feasible sets like the one shown in Fig. 5? Fig. 7 compares active and passive. Notice how the active learning algorithm consistently selects training examples with widely separated, disjoint feasible regions in the configuration space. In contrast, the passive algorithm prefers feasible sets where at least one of the feasible configurations is near the mode of the current belief $P(q_H^*)$. This is reasonable, as the passive algorithm attempts to greedily reduce the human's ergonomic cost, at the expense of slower learning. After

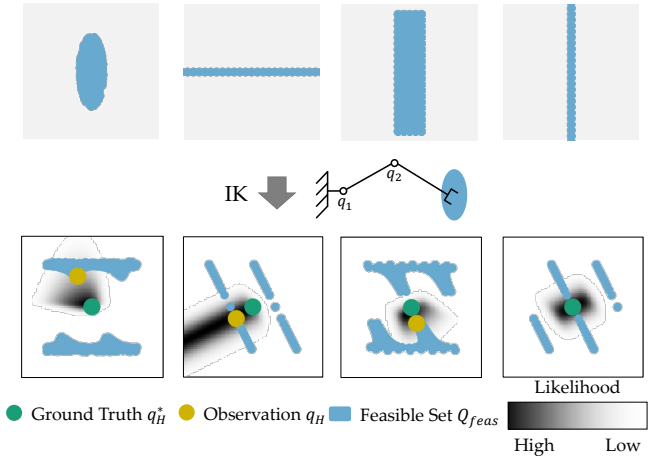

Fig. 5: A collection of example 2D task space objects and their corresponding configuration space representations. The objects are all centered at the same position, and only their shape varies. Even with this constraint, the resulting configuration space feasible sets and likelihoods vary widely.

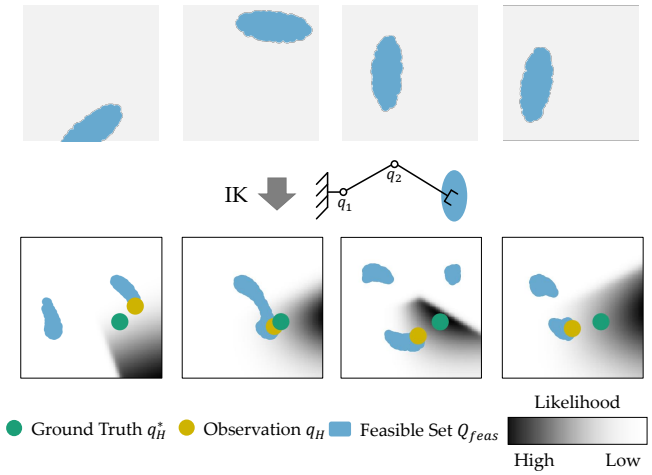

Fig. 6: A set of example 2D task space objects and their corresponding configuration space representations. Every object has the same shape, but they are each positioned at a different random pose. Just this change in pose creates significant variation between the configuration space feasible sets and likelihoods generated by each object.

the third iteration, the active learning algorithm's belief has converged to the correct ground truth value. In contrast, the passive algorithm's belief is still somewhat uncertain, and the algorithm continues to select a query whose resulting likelihood function will not remove the uncertainty.

VI. Analysis for a Seven DoF Human Arm

Our second test scenario used a simulated 7 DoF human arm and real objects. Inspired by the advantage of objects that induce separate feasible regions, we use a set of bicycle handlebars, but also a bicycle U-lock which does not have this property.

As in Sec. V, we conducted a total of 50 simulation trials each for the weight and neutral configuration learning portions of the test. For each trial, we supplied the robot with a randomly selected menu of eight object handoff poses and grasps. The ground truth parameters were set to a randomly chosen value, which we attempted to recover. The results are in Fig. 8.

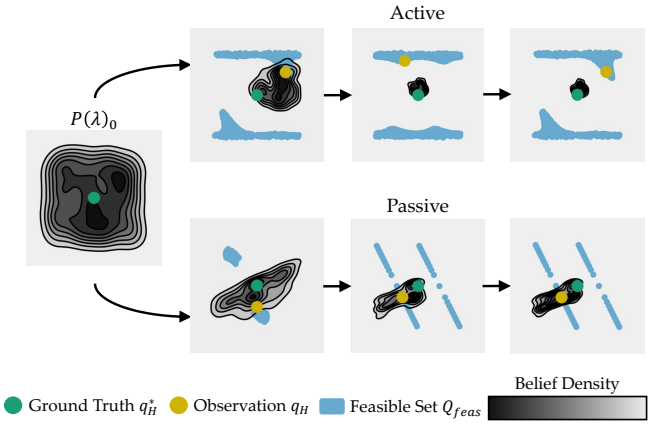

Fig. 7: Active and passive learning algorithms applied to the same scenario. The active algorithm consistently selects queries with widely separated, disjoint feasible regions in the configuration space, while the passive algorithm tends to select queries with at least one feasible configuration close to the ground truth optimal value. The active algorithm's belief converges quickly to the ground truth value, while the passive algorithm allows significant uncertainty to remain after the first three iterations.

As in Sec. V, the active learning algorithm produced consistently faster learning. Particularly notable was the learning of the joint weights, where the passive algorithm produced a test set likelihood which was worse than that of the initial belief, but the active algorithm performed acceptably.

To help explain this, examine Fig. 9. It shows both the task space feasible sets and corresponding configuration space feasible sets for two selected poses of the bike handlebars and lock. As our earlier trials suggested, the bicycle handlebars, with their two widely separated grip zones produced a much larger, more dispersed configuration space feasible set than the bike lock. Active learning exploits this.

VII. User Study

Simulation enabled us to tease out interesting aspects of the passive and active approaches, but we still need to study how well these methods perform with real people. We thus conducted a user study with a real handover task. Our study had five participants, two female and one male, ranging in age from 20 to 28.

The handover task used the same bike handlebar and lock objects from the previous section. We selected a menu of robot queries containing four poses of the handlebars and four poses of the lock. We attached rubber grips to both objects to limit the feasible grasps on each to two distinct regions (as shown on the left side of Figure 9).

Our study consisted of three phases: two training and one test. In the first phase, the robot conducted five iterations of training using the active learning algorithm. In the second phase, five iterations of the passive algorithm were performed. In the third phase, each of the two objects was presented to the participant at eight different poses, for a total of 16 testing examples. The manipulated variables were unchanged from our earlier simulation studies, but because we no longer had access to the humans' ground truth cost

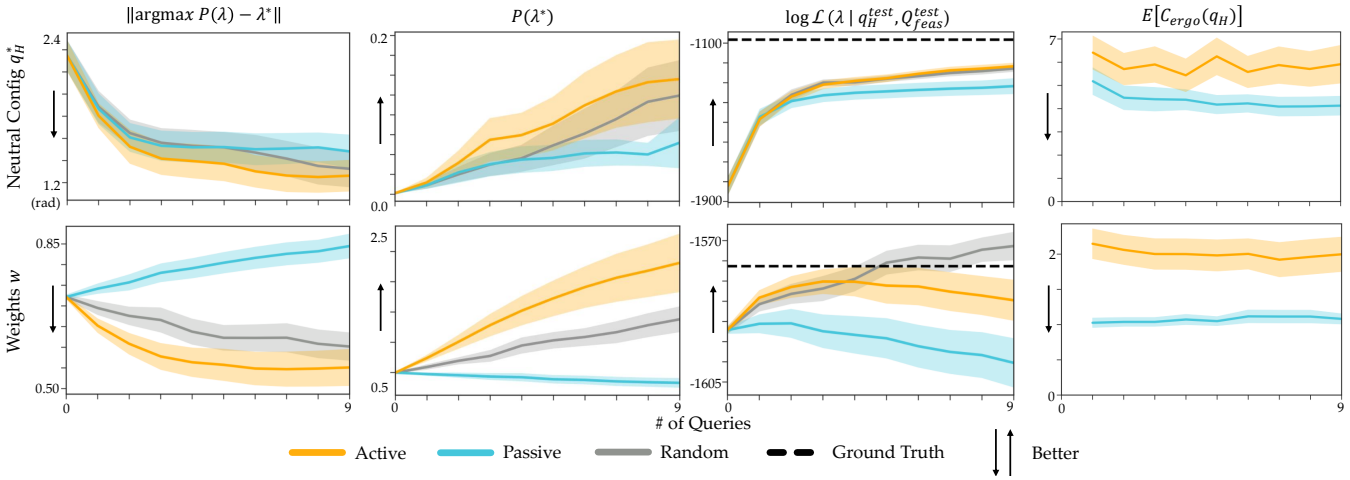

Fig. 8: The values of each objective measure vs. the number of training handovers, shown for neutral configuration (q^*) learning on a simulated handoff task with a 7 DoF human arm model.

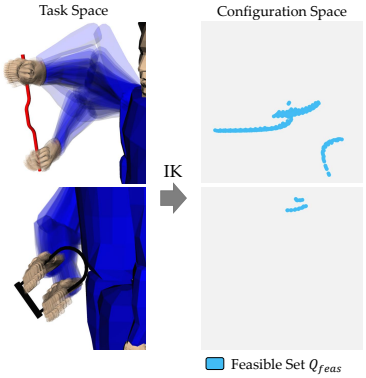

Fig. 9: Task space sets of human grasp configurations pictured alongside the corresponding configuration space feasible sets Q_{feas}. Notice how the bicycle handlebars produce a large configuration space feasible set with multiple disjoint portions (informative), while the lock produces a much more compact feasible set (uninformative)

functions, we used only the third objective measure: test set log likelihood. Intuitively, this number measures the accuracy with which the robot's learned models were able to predict the humans' actions in a new situation. [2]

The performance of the two algorithms is shown in Fig. 10. The active learning algorithm consistently yielded a higher likelihood on the test dataset than the passive algorithm for the first two training iterations. After this, the two algorithms were close to identical in performance. This suggests that the active algorithm successfully selected initial training queries which helped it to quickly identify the human's ergonomic model.

Although both training algorithms had access to both the bike handlebars and the bike lock, the active algorithm selected the handlebars exclusively, while the passive algorithm selected only the lock. This is intuitively reasonable: The handlebars induce a large, fragmented configuration space feasible set, which yields informative human responses. The lock produces a small, compact set, which is much less informative, but makes it easy for the robot to guarantee that whichever grasp the human chooses will be reachable comfortably.

[2]Since we weren't able to measure the human participants' arm configurations accurately in real time, we instead just provided the active and passive algorithms with the human's chosen grasp region (i.e. which handle they grabbed). The training example q_H was then taken to be the most likely configuration given that grasp, according to a previously specified model.

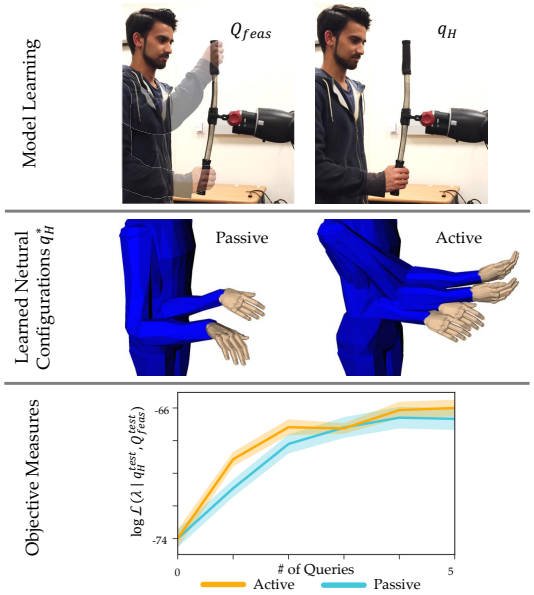

Fig. 10: The user study asked subjects to select one grasp configuration q_H from a set of two available grasps Q_{feas} in each training query. The passive learning algorithm induced many participants to choose identical labels q_H, and learned only two distinct neutral configurations q_H^* from the set of five subjects. The active algorithm's informative queries elicited a wider variety of behavior, yielding four distinct q_H^*'s. Measured by the test set log likelihood, active learning consistently produced a better model fit after the first one to two iterations, after which the passive algorithm's model had a comparable likelihood. This result is similar to the 7 DoF simulation results in Fig. 8. The simulation results suggest that the active vs. passive performance difference may be larger for weight (w) learning.

As shown in Table I, participants had a slight preference for the passive learning algorithm, but found both just as physically easy.

VIII. Discussion

We formulated ergonomic cost learning as an online estimation problem based on implicit physical queries

from the robot. We compared a passive and an active learning approach. In simulation, we found that the active method leads to faster learning and higher accuracy, but sacrifices user comfort during learning. We also discovered that active learning works best when the object that the robot is handing, combined with the grasping configuration it chooses, leads to feasible choices for the human that have different connected components. With real people, the differences were more subtle: active learning is significantly more accurate at first, but passive quickly catches up. Users preferred the comfort of passive, but did not rate it as physically easier to work with.

Above all, what is exciting is that the user study suggests that this online estimation works with real people, in that it improves how well the robot can predict what a real user would do in new situations. An active technique will make it more likely that the robot converges to a better model, but the extent to which that matters in practice remains an open question, with passive techniques also performing well overall.

TABLE I: Post-Study Survey Results

Statement	Active	Passive
"I prefer Program __"	3.4	4.4
"The robot was helpful when running Program __"	4.6	4.6
"It was physically easy to do the task when the robot was running Program __"	4.8	4.8
"The robot running Program __ handed me objects in a way that made the task easier"	4.8	4.6
"If you had to choose a program you prefer, which would it be?"	20%	80%

References

[1] A. Bestick, R. Bajcsy, and A. Dragan, "Implicitly assisting humans to choose good grasps in robot to human handovers," in *International Symposium on Experimental Robotics*, 2016.

[2] A. Bestick, S. A. Burden, G. Willits, N. Naikal, S. S. Sastry, and R. Bajcsy, "Personalized kinematics for human-robot collaborative manipulation," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 1037–1044, IEEE, 2015.

[3] J. Mainprice, E. A. Sisbot, T. Siméon, and R. Alami, "Planning safe and legible hand-over motions for human-robot interaction," in *IARP workshop on technical challenges for dependable robots in human environments*, vol. 2, p. 7, 2010.

[4] E. A. Sisbot and R. Alami, "A human-aware manipulation planner," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1045–1057, 2012.

[5] J. Mainprice, M. Gharbi, T. Siméon, and R. Alami, "Sharing effort in planning human-robot handover tasks," in *RO-MAN*, pp. 764–770, IEEE, 2012.

[6] A. Gritsenko and D. Berenson, "Learning Task-Specific Path-Quality Cost Functions From Expert Preferences," in *NIPS Workshop: Autonomously Learning Robots*, 2014.

[7] D. Sadigh, A. D. Dragan, S. Sastry, and S. A. Seshia, "Active Preference-Based Learning of Reward Functions," *Robotics science and systems (RSS)*, 2017.

[8] C. Daniel, M. Viering, J. Metz, O. Kroemer, and J. Peters, "Active Reward Learning," *Robotics: Science and Systems (RSS)*, vol. 10, 2014.

[9] M. Cakmak, C. Chao, and A. Thomaz, "Designing Interactions for Robot Active Learners," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 2, pp. 108–118, 2010.

[10] D. Braziunas, "Computational Approaches to Preference Elicitation," tech. rep., 2006.

[11] B. Settles, "Active Learning Literature Survey: Computer Sciences Technical Report 1648," tech. rep., University of Wisconsin, Madison, 2010.

[12] "Label ranking by learning pairwise preferences," *Artificial Intelligence*, vol. 172, no. 16-17, pp. 1897–1916, 2008.

[13] F. L. Wauthier, M. I. Jordan, and N. Jojic, "Efficient Ranking from Pairwise Comparisons," *International Conference on Machine Learning*, 2013.

[14] B. Carterette, P. N. Bennett, D. M. Chickering, and S. T. Dumais, "Here or There: Preference Judgments for Relevance," in *European Conference on Advances in Information Retrieval (ECIR)*, 2008.

[15] A. Karbasi, S. Ioannidis, and laurent Massoulie, "Comparison-Based Learning with Rank Nets," *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pp. 855–862, 2012.

[16] A. E. Abbas, "Entropy methods for adaptive utility elicitation," *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans.*, vol. 34, no. 2, pp. 169–178, 2004.

[17] C. Boutilier, "A POMDP Formulation of Preference Elicitation Problems," *Proceedings of AAAI Conference on AI*, pp. 239–246, 2002.

[18] J. Fürnkranz, E. Hüllermeier, W. Cheng, and S.-H. Park, "Preference-based reinforcement learning: a formal framework and a policy iteration algorithm," *Machine Learning*, vol. 89, no. 1-2, pp. 123–156, 2012.

[19] R. Holladay, S. Javdani, A. Dragan, and S. Srinivasa, "Active Comparison Based Learning Incorporating User Uncertainty and Noise," in *RSS Workshop on Model Learning for Human-Robot Communication*, June 2016.

[20] G. Druck, B. Settles, and A. McCallum, "Active learning by labeling features," *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 2009.

[21] J. Kulick, M. Toussaint, T. Lang, and M. Lopes, "Active learning for teaching a robot grounded relational symbols," *International Joint Conference on Artificial Intelligence*, 2013.

[22] C. Dima and M. Hebert, "Active Learning For Outdoor Obstacle Detection," in *Robotics: Science and Systems*, 2005.

[23] D. Sadigh, S. S. Sastry, S. A. Seshia, and A. Dragan, "Information gathering actions over human internal state," *IEEE International Conference on Intelligent Robots and Systems*, 2016.

[24] A. Ng and S. Russell, "Algorithms for inverse reinforcement learning," in *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000.

[25] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pp. 2586–2591, 2007.

[26] C.-m. Huang, M. Cakmak, and B. Mutlu, "Adaptive coordination strategies for human-robot handovers," in *Robotics: Science and Systems (RSS)*, 2011.

[27] M. Cakmak, S. S. Srinivasa, M. K. Lee, J. Forlizzi, and S. Kiesler, "Human preferences for robot-human hand-over configurations," in *IEEE International Conference on Intelligent Robots and Systems*, pp. 1986–1993, IEEE, 2011.

[28] J. Kim, J. Park, Y. Hwang, and M. Lee, "Advanced grasp planning for handover operation between human and robot: three handover methods in esteem etiquettes using dual arms and hands of home-service robot," *2nd International Conference on Autonomous Robots and Agents*, pp. 34–39, 2004.

[29] V. Micelli, K. Strabala, and S. Srinivasa, "Perception and control challenges for effective human-robot handoffs," *RSS 2011 RGB-D Workshop*, 2011.

[30] A. H. Quispe, H. B. Amor, and M. Stilman, "Handover planning for every occasion," *IEEE-RSJ International Conference on Humanoid Robots*, 2014.

[31] A. H. Quispe, H. Ben Amor, H. Christensen, and M. Stilman, "It takes two hands to clap: towards handovers in bimanual manipulation planning," in *Robotics: Science and Systems (RSS)*, 2014.

[32] K. Strabala, M. K. Lee, A. Dragan, J. Forlizzi, S. S. Srinavasa, M. Cakmak, and V. Micelli, "Towards seamless human-robot handovers," *Journal of Human-Robot Interaction*, vol. 1, no. 1, pp. 112–132, 2012.

[33] D. Berenson, S. Srinivasa, and J. Kuffner, "Task Space Regions: A framework for pose-constrained manipulation planning," *The International Journal of Robotics Research*, vol. 30, no. 12, pp. 1435–1460, 2011.

[34] S. W. Yoon, A. Fern, R. Givan, and S. Kambhampati, "Probabilistic planning via determinization in hindsight," in *AAAI*, pp. 1010–1016, 2008.

[35] E. K. Chong, R. L. Givan, and H. S. Chang, "A framework for simulation-based network control via hindsight optimization," in *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, vol. 2, pp. 1433–1438, IEEE, 2000.