Online Behavior Modification for Expressive User Control of RL-Trained Robots

Isaac Sheidlower
Tufts University
Medford, Massachusetts, USA
isaac.sheidlower@tufts.edu

Mavis Murdock
Tufts University
Medford, Massachusetts, USA
mavis.murdock@tufts.edu

Emma Bethel
Tufts University
Medford, Massachusetts, USA
emma.bethel@tufts.edu

Reuben M. Aronson
Tufts University
Medford, Massachusetts, USA
reuben.aronson@tufts.edu

Elaine Schaertl Short
Tufts University
Medford, Massachusetts, USA
elaine.short@tufts.edu

ABSTRACT

Reinforcement Learning (RL) is an effective method for robots to learn tasks. However, in typical RL, end-users have little to no control over *how* the robot does the task after the robot has been deployed. To address this, we introduce the idea of *online behavior modification*, a paradigm in which users have control over behavior features of a robot in real-time as it autonomously completes a task using an RL-trained policy. To show the value of this user-centered formulation for human-robot interaction, we present a behavior-diversity–based algorithm, Adjustable Control Of RL Dynamics (ACORD), and demonstrate its applicability to online behavior modification in simulation and a user study. In the study (**n**=23), users adjust the style of paintings as a robot traces a shape autonomously. We compare ACORD to RL and Shared Autonomy (SA), and show ACORD affords user-preferred levels of control and expression, comparable to SA, but with the potential for autonomous execution and robustness of RL. The code for this paper is available at https://github.com/AABL-Lab/HRI2024_ACORD

CCS CONCEPTS

• **Human-centered computing**; • **Computing methodologies** → **Reinforcement learning**; *Artificial intelligence*;

KEYWORDS

human-robot interaction, user-centered learning, shared autonomy, reinforcement learning

ACM Reference Format:
Isaac Sheidlower, Mavis Murdock, Emma Bethel, Reuben M. Aronson, and Elaine Schaertl Short. 2024. Online Behavior Modification for Expressive User Control of RL-Trained Robots. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction (HRI '24), March 11–14, 2024, Boulder, CO, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3610977.3634947

HRI '24, March 11–14, 2024, Boulder, CO, USA

ACM ISBN 979-8-4007-0322-5/24/03.
https://doi.org/10.1145/3610977.3634947

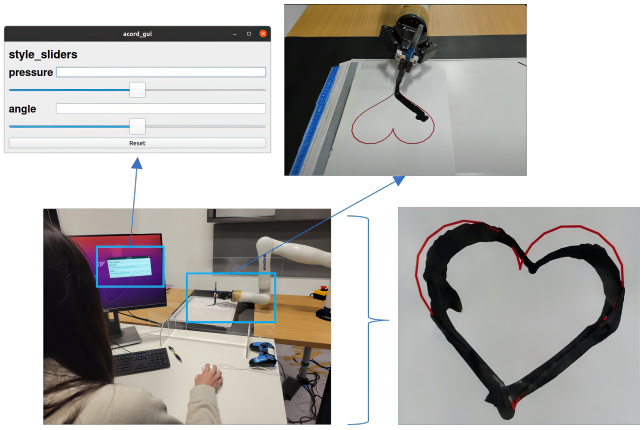

Figure 1: A participant using ACORD to adjust the style of a painting as the robot traces a heart autonomously.

1 INTRODUCTION

Real-world robots must complete tasks well and meet the needs of users. In many cases, a robot is optimized for only one of these. For instance, an industrial assembly line robot is programmed to perform a very specific task in a very specific way, typically in an isolated environment, and thus requires relatively little supervision from a person. Such a robot may have learned to complete the task optimally through Reinforcement Learning (RL). However, this "one policy fits all" approach is unlikely to work when robots are working closely with humans. There are many cases where users may wish to have a robot that can autonomously perform a task while allowing for control over some dimensions of the robot's behavior. For example, a user may want a dishwashing robot to move more slowly when cleaning their favorite mug or an assistive robot to use less force when helping with dressing. While an RL-based policy may be successful at completing the task, it may not suit the user's in-the-moment user needs for *how* that task should be completed.

In many situations, there is a need to facilitate interactions that give users this control over the style of task completion without burdening the user with potentially-lengthy human-in-the-loop teaching [13, 55]. Some existing methods augment a typical RL policy, but these methods have not been adapted for or validated with real users. Existing methods include goal-conditioned RL (GCRL)

for example, in which a robot's behavior is dictated by the parameterization of a goal state. Similarly, behavior diversity approaches often parameterize a robot's policy with a latent variable that encodes a skill or certain way of completing a task. These approaches are *robot-centered* approaches which do not explicitly allow user control over the resulting policies. We believe these approaches can be reformulated in a *user-centered* way to give users control over a robot's behavior as it completes a task. To enable close user-robot collaboration, we propose and study an approach that gives users direct control over these latent variables to adjust a robot's behavior to their liking.

In this paper, we present *online behavior modification*, a formulation that combines fully autonomous task completion with user-controlled behavior styles. This formulation is compatible with state-of-the-art methods for learning a task offline, such as RL, GCRL, and quality-diversity (QD), while making explicit a degree of online user control. We then present Adjustable Control Of RL Dynamics (ACORD), a user-centered, diversity-based algorithm which serves as a proof of concept for this formulation. We deploy ACORD in a user study to demonstrate its potential in a user-centered interaction that does not sacrifice task completion. Our contributions include:

(1) We propose the *online behavior modification* formulation, which describes an interaction where a robot autonomously completes a task while a user controls how it does so.
(2) We present ACORD, a diversity-based algorithm designed for online behavior modification. ACORD grants users continuous control over pre-specified behavioral features of the robot while ensuring autonomous task completion.
(3) We validate ACORD in an in-person study with non-expert users (n=23) in a collaborative painting task where users adjust the *style* of painting using ACORD, fully-autonomous RL, and a modified version of Shared Autonomy (SA). We find that ACORD is rated by users as affording the same high level of control as SA (82% agree with ACORD giving control, 73% with SA, versus 30% agreeing with RL giving control), while maintaining better overall task performance (**BF**=11.67 in our measure of consistency). Furthermore, we find users strongly prefer interacting with ACORD over the RL baseline (e.g., 83% preferred ACORD, **BF**=17.16).

2 RELATED WORK

Users like to have control over robots. Users desire this control via teleoperation [14, 24, 25, 41, 54], via dictating which actions the robot should *not* take [2, 9, 52], or via having a degree of direct control over a collaborative algorithm [20, 38, 56, 57]. A theme of these works is that allowing users to influence robot behavior allows for more expressivity and robustness than a single policy may represent. In fact, RL is also moving towards more expressive and capable policies. Goal-conditioned RL (GCRL), for example, enables a policy to perform different tasks depending on how a goal state is selected [12, 17, 29, 32]. Skill learning and diversity-based approaches [16, 31, 34], such as Quality-Diversity (QD) [15, 18, 43, 48, 49], allow a robot to autonomously learn meaningfully variations in its behavior. Algorithmically similar to our work, Kumar et al. [28] and Osa et al. [40] use diversity-based approaches to increase an agent's robustness to its environment, while we propose using similar techniques to give users more control over a robot's behavior. We also highlight that most previous robot-centered approaches have not been validated with users, a critical step to ensure that these methods serve the needs of users.

While pure teleoperation maximizes user control, our work is applicable in tasks where direct user control is impractical or impossible. A more analogous method, used as a baseline in this work, is Shared Autonomy (SA), which starts with user direct control and adds an automated assistance behavior to direct the robot to an inferred user goal or skill based on some input [20, 23, 35, 42, 46]. Although RL has also been used to enhance SA [18, 45], it has not been used to adjust *how* the robot completes its task given the target task is known. There is a need for approaches such as ours that do just that.

Human-in-the-loop learning has offered approaches to guiding robot behavior via reward shaping [7, 11, 37, 44], ensuring various safety constraints are met [2, 30, 52], or, most closely related to how a robot does a task, via queries about behavior features [6–8]. Approaches such as Interactive RL emphasize teaching a robot in real-time as it adapts to the teacher's feedback [5, 26, 47]. While these approaches are effective at allowing users to alter robot behavior, they often require both lengthy teaching times and retraining when a user changes their preferences, so the adjustment occurs over several executions of the task. To complement these more time-consuming methods, there is a need for approaches such as ours that allow users to quickly change a robot's behavior in the moment, within the same task execution.

3 LEARNING POLICIES FOR ONLINE BEHAVIOR MODIFICATION IN RL SETTINGS

To enable human-centered control over how a robot complete its task, we propose three key properties for *online behavior modification*. First, the robot must always **autonomously make "task progress"** and ensure the task does not fail. In this context, "progress" may mean "expected completion in finite time" or "always getting closer to a goal"; formalization depends on the task. Second, there must be a non-empty set of **behavior features**, each of which has an associated *behavior oversight parameter*, k, that control the robot along the behavior feature axis. In other words, the policy must be explicitly parameterized with one or more observable variables that dictate an aspect of the robot's behavior. Finally, for each behavior feature that has a certain k associated with it, the adjustment of that k must be **interpretable to a user** and there must be an **accessible interface** that facilitates a user to freely adjust each k as the robot completes its task. These properties describe an interaction that ensures the user can have a robot that both meets their needs and can be personalized without having to teach the robot the task or their preferences.

In this section, we present Adjustable Control Of RL Dynamics (ACORD), a proof-of-concept algorithm for learning a policy for online behavior modification in continuous state and action space robotics tasks. ACORD is a behavior-diversity–inspired algorithm which explicitly gives users control over a robot's behavior. We describe how to adapt a standard RL setting to facilitate ACORD and demonstrate it in a simulation environment.

3.1 ACORD for Continuous Control RL-tasks

We assume a task modeled as a Markov decision process (MDP) with states S, actions A, transition function $T(s,a) \rightarrow s'$, and discount factor γ. To define task failure, we assume some environmental reward function R_{env}. To this system, we introduce *behavior oversight parameters*. Assume that $S = \mathbb{R}^n$ and define the space of behavior oversight parameters as $K = [0,1]^m, 1 \leq m \leq n$. Consider the coordinate representation of $s = \langle s_1, \cdots, s_i, \cdots, s_n \rangle$ and $k = \langle k_1, \cdots, k_j, \cdots, k_m \rangle$. Each coordinate of k, k_j, controls a coordinate of s, noted s_i. The set of all s_i that have a k_j mapping to them define a set of *behavior goals* for the robot, and the corresponding i-axes are *behavior feature* axes. Any s_i with no corresponding k_j is a free variable whose value is not explicitly constrained by a setting of k. For generality, we assume the range of behavior goals is unknown prior to learning (e.g., the maximum and minimum speeds the robot can move while completing its task are unknown). After learning, a user can directly adjust the values of k, thus changing the robot's behavior goal on the axis s_i, and consequently changing its behavior along that axis within a range that is learned by the algorithm, subject to "non-failure" condition above. This representation could be trivially extended to having k_j control multiple coordinates.

Learning a policy for ACORD entails finding a policy parameterized by k, π_k, which both makes progress in the task and enforces the behavior goals. To ensure that the learned mapping from each k_j to each s_i is interpretable by a user, we propose the soft constraint that the robot should learn a monotonic mapping from k_j to s_i and that the mapping range is as large possible without preventing the robot from completing its task.

3.2 ACORD Algorithm

ACORD makes use of three components: a discriminator that learns a continuous mapping from $s_i \rightarrow k_j$ to generate a diversity-inspired reward; an environment reward to define failure states and a task progress heuristic $h(s,a)$ to ensure task performance; and a domain randomization component that ensures that the agent learns and is robust to various different settings of k such that k may be adjusted in real time.

ACORD Discriminator We train a set of discriminators W_j to predict k_j given s_i, denoted: $W_j(s_i) \in [0,1]$. We parameterize the discriminator as a neural network and train it via the novel loss function:

$$L(W_j(s_i), k_j) = MSE(W_j(s_i), k_j) + \frac{1}{|\max(W_{j,s_i \sim D}(s_i)) - \min(W_{j,s_i \sim D}(s_i))| + \varepsilon} \quad (1)$$

where $W_{j,s_i \sim D}$ refers to the discriminator output of a batch sampled from a replay buffer D_W, and ε is a small number to avoid division by zero. This loss function enforces high prediction accuracy (via MSE) and that the predictions cover as wide a range as possible. The latter property is explicitly enforced by the denominator, leading to a faster convergence to the range covered by each k_j, resulting in more stable task behavior (see supplementary material for ablation study).

RL Task Description and Agent We define the state space of the RL agent to be $S \bigcup K$. This makes k observable to the agent. We will still denote any given state with s. We design a reward function such that the agent avoids failures, makes progress, and learns to enforce behavior goals:

$$R(s,a) = \begin{cases} R_{\text{env}}(s) & \text{if } s \in F^* \\ -c & \text{if } h(s,a) \leq 0 \\ \frac{1}{m} \sum_{i=1}^{m} (-\log |W_i(s_i) - k_i|) & \text{else} \end{cases} \quad (2)$$

where R_{Env} denotes the reward from the environment, F^* is the set of failure states which lead to a large negative reward, $h(s,a)$ denotes a heuristic for measuring task progress, and c is a positive constant that punishes the agent if it fails to make task progress. Last is the reward generated by the discriminator which ensures that, for a given k_i, the agent is acting in the part of the state space where the discriminator can easily predict the k_i value. Since $|W_i(s_i) - k_i| \in [0,1]$, this reward is always positive and the other conditions are always negative. This allows the reward function to be adapted and scaled to different environments with relative ease. Each of these terms may be scaled by a constant. We maximize this reward via the off-policy RL algorithm SAC [21].

Domain Randomization Over K We employ domain randomization [36, 50] for the setting of k during training. Every n time steps, we sample $k_i \sim \text{Uniform}(0,1) \forall k_i \in k$. The choice of n can be difficult as when a given k_i changes, it may take several steps for the robot to adjust its behavior accordingly. If n is too small, the algorithm cannot learn to enforce the value of k over time, and if n is too large, it cannot learn to react efficiently to a user changing k real time. Empirically, we find in the tasks in this paper that a reasonable choice for n is about half the length of an episode; we expect that this would be the case for many tasks.

Algorithm 1: ACORD

```
Initialize off-policy RL Learner Ψ
Initialize Discriminator(s) W
for environment step t do
    if nth step then
        k ~ Uniform(0,1)^m
    s_t ~ s_{t,env} concatenate k
    a_t ~ π_Ψ(a_t|s_t)
    s_{t+1,env} ~ p(s_{t+1}|s_t, a_t)
    s_{t+1} = s_{t+1,env} concatenate k
    r_t ~ R(s,a) [see Eq. 2]
    D_Ψ ← D_Ψ ⋃ (s_t, a_t, s_{t+1}, r_t)
    D_W ← D_W ⋃ (s_t, a_t, s_{t+1}, r_t)
    if zth step then
        Update Ψ via gradient descent
    if vth step then
        Update all W via loss in Eq. 1
```

3.2.1 On Using a Heuristic Progress Function. Online behavior modification as an interaction emphasizes that the robot can autonomously complete the task by constantly making progress in that task. There are several ways to formalize this constraint, and online behavior modification does not necessarily require a particular one. For example, in this work we define a task progress measure $h(s,a)$ and require that π_k prioritize trajectories that make

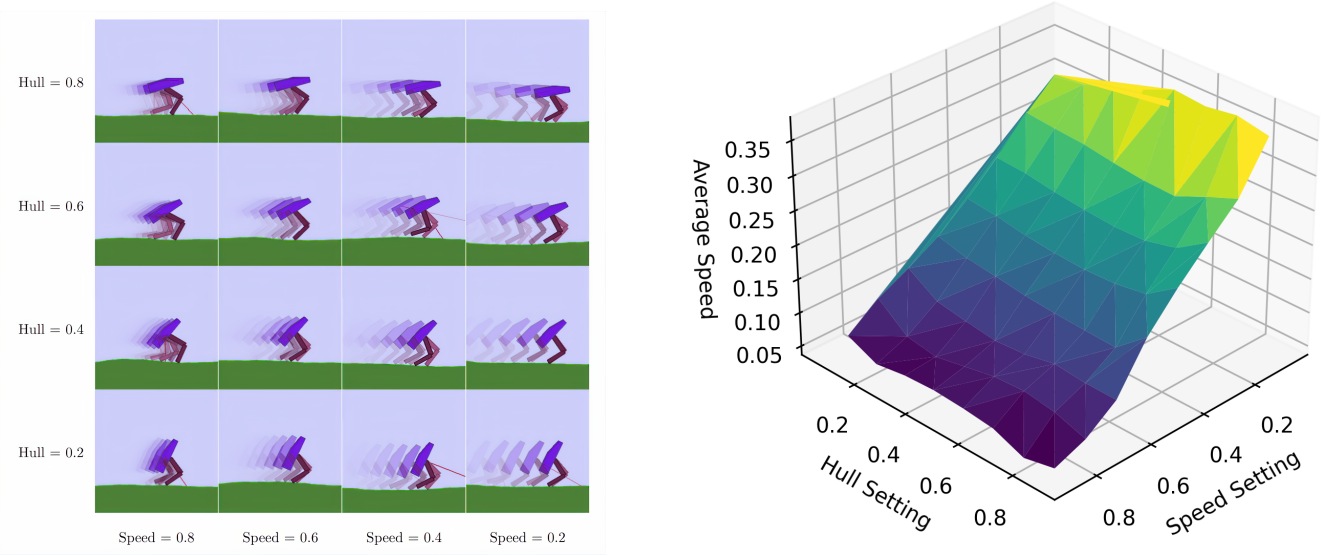

Figure 2: Left: The walking agent varies its behavior in a predictable and interpretable way given changes of k. The ghost traces from the previous six video frames show the agent's change in speed. Right: The resulting manifold learned by ACORD in the walker environment. The speed is robust to different hull angles.

$h(s, a)$ non-negative; this approach is appropriate for many robotics problems where there is a physical destination for the robot's motion (e.g., [32]). Another natural approach might be to use the environmental reward function $R_{\text{env}}(s, a)$ to measure task progress or require that the trajectories following π_k eventually reach a terminal success state. The exact specification will depend on the task and the formulation of the learning problem.

A heuristic progress function h can ensure the robot always completes the task despite a user changing how it does so. This aligns with our goal of giving users the most control possible over a robot's behavior while still accomplishing the task. This is in contrast to prior approaches that optimally solve for a trade-off between environmental reward and diversity, as in Quality-Diversity-based approaches [24, 25], or use a hyperparameter to dictate how each of the two objectives are weighted [26].

3.3 ACORD in Simulation

We train ACORD in simulation to show that the learned policy has the desired properties: it aligns pre-specified behavior features to the values specified by k; it has an interpretable behavior range over ks; and it completes the task and avoids failures robustly in variations in k. In a bipedal walker task [10], we specify two behavior oversight parameters: k_1 to control the speed of the robot along its x-axis and k_2 to control the angle of its hull. Failure cases are specified as crashing (-100 reward from the environment). We measure task progress by setting $h(s, a) = v_x$, the velocity of the robot along the x axis. Then, Eqn. 2 penalizes the system for moving backwards in x. We trained the agent to convergence prior to evaluation (~2 million steps; for a discussion of algorithm efficiency see Section 6). Figure 2, left, shows the resulting behavior by varying both ks. By changing k_j, there is a predictable change in behavior along the specified feature axis. Figure 2, right, shows the range over the robot's speed for various settings of k_1 given across different values of k_2. This demonstrates that ACORD can be robust to multiple settings of k_1 given k_2: varying the hull angle does not fully constrain the agent's ability to vary its speed. Of course, if two features are directly in conflict with each other, such as a k_i mapped to going backwards and a k_j mapped to going forwards, the behavior of the robot may not be as expected. Lastly, over multiple runs, the agent avoids crashing ~ 94% of the time with variations in many settings of K.

4 USER STUDY

To study ACORD and online behavior modification with real users, we designed a robot painting environment wherein users can adjust a robot's painting style as or before it traces a drawing. This domain is an inherently creative activity in which a person has styles and preferences that they wish to express. Online behavior modification captures the idea that task completion itself is not always the only desirable metric of a human-robot interaction: having control over *how* the task is completed can also be an important factor, as is the case with painting and other artistic tasks.

Robot Painting Task The painting task involved the robot tracing a previously generated shape. We specify each shape as an ordered list of waypoints in the x-y plane, $(p_0, \cdots, p_r)$. We formulate the task as an MDP where the state s is a vector containing the robot's end-effector position, orientation, and velocity; the position and orientation of a brush the robot is gripping; and the next waypoint that the robot should reach. Actions are relative Cartesian x-y velocities. Reward is given as $R(s, a) = -|p_{\text{brush}} - p_i|$, the negative distance between the current pose of the brush p_{brush} and the next waypoint p_i. Episodes terminate when the robot has reached every waypoint that makes up the shape or with failure when the arm leaves the workspace or is in collision.

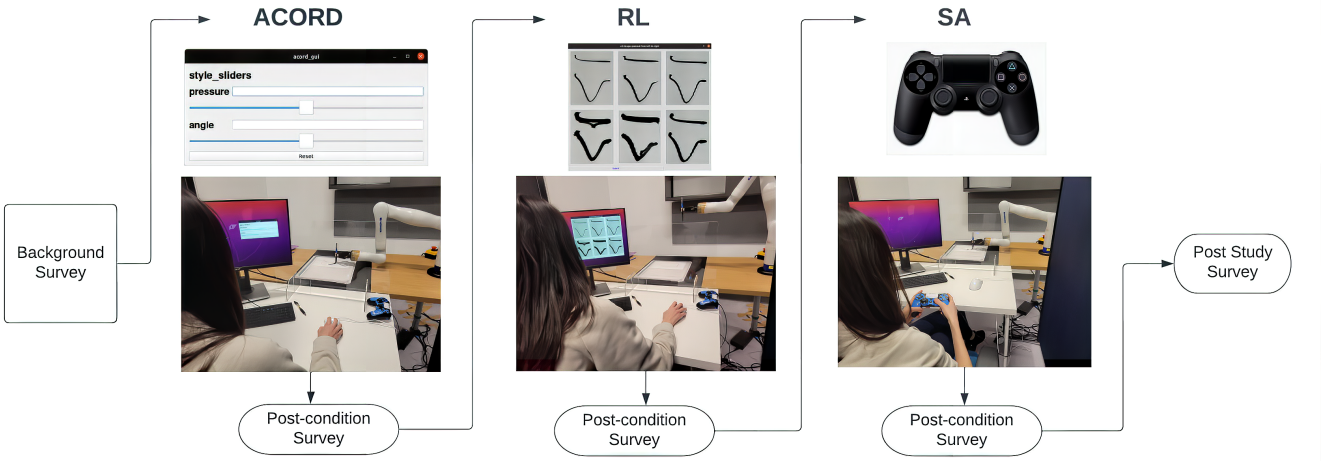

Figure 3: Overview of the study procedure. Participants interacted with each of the three conditions (order was counterbalanced), completing a survey after each condition.

Experimental Setup The setup (Figure 3) consisted of a Kinova Gen3 robot arm on a table with the participant sitting next to it. Depending on the condition, users had access to a different interface to interact with the robot. On the table was paper with a shape outlined in red on which the robot would paint. The participants were told which shape they would paint: heart or house (Figure 3). These shapes contain various motions and strokes and provide scope for participants to paint in their own style.

Painting Styles We define two different axes for the robot to vary its painting style. One is by adjusting the height of the brush or end-effector, thus affecting the pressure that the brush applies to the canvas. This can result in thinner or wider strokes. The other way is by rotating the robot's wrist or brush. This adjusts the angle of the brush, resulting in more varied strokes.

4.1 Conditions

We assume for all conditions that the robot knows how to perform the task optimally according to the MDP formulation. We fix the painting policy across each baseline to ensure the same amount of time is spent on each painting and that the style adjustment was the primary difference between conditions. We compare ACORD to two alternatives to vary the style of robot behavior: RL and SA.

Choosing Among a Discrete Set of Style-Varying RL Policies This condition gave the robot the most autonomy. Participants selected one of six styles based on an example image before the robot drew the shape. Each style represented a fixed value for the pitch and height of the end-effector. The robot then painted the shape autonomously according to that selected style. This type of control, in which a user chooses between a set of RL policies, is appropriate for tasks where RL control is necessary and/or available and "styles" are well defined, such as choosing a "risky" or "risk-averse" obstacle avoidance strategy. In other cases, these predefined policies may have been learned via human feedback, but their execution during this single task is fixed.

Shared Autonomy (SA) This condition gave the participants the most direct control. Users were given assisted velocity control over the height and pitch axes of the robot end-effector through a controller. The input was augmented with a SA assistance strategy following [22, 23], with $\alpha = 0.5$ to allow the user's commands to directly influence the robot position [39]. The SA assistance infers online which of the six styles defined in the previous condition the user is intending to achieve.

While similar to the standard goal-based SA paradigm, we note two key differences. First, the system continuously moved along the x-y plane via the optimal policy while the user controlled the style axes. Second, rather than considering goal states to be terminal, the user continued to control the style axes for the whole trajectory and could move from one goal then to another. This approach allows for the closest comparison between ACORD and SA, but this multi-goal formulation of SA is a direction for future research in itself.

Adjustable Control Of RL Dynamics We trained and deployed an ACORD agent using sim-to-real via the Gazebo simulation environment [27]. Failure was defined as leaving a set workspace. We defined $h(s, a) = a \cdot (p_i - p_{\text{brush}})$, the component of the action in the direction towards the current waypoint p_i. Penalizing $h(s, a) \leq 0$, as in Eqn. 2, penalizes actions that move away from p_i.

Two ks were learned to allow for *continuous control* over the painting style: one for the height, k_1, and angle, k_2, measured at the *brush tip* rather than at the robot's end-effector. This means when a user moves the slider to adjust the brush's rotation, through k_1, ACORD maintains contact with the paper since k_2 stays the same. The users had access to a GUI with two sliders to control both ks. Users adjusted the sliders, affecting the robot's behavior and painting style in real time.

4.2 Experimental Procedure

Recruitment We recruited a total of 24 participants from the university and the surrounding area with a variety of different backgrounds. All participants were 18 years or older. Of those participants 15 were female and 9 were male. 13 participants were in the age range of 18-24, 9 in the range of 25-35, 1 in the range of 35-44 and 1 in the range of 55-64. Participants reported their level of programming expertise from 0 (none) to 10 (expert). The mean level of programming experience was 2.9 with a standard deviation of 2.3. Furthermore, 11 participants reported having experience interacting with robots, and 3 of those 11 had significant expertise

Figure 4: Participant paintings. Users were able to produce a wide range of different styles for the pre-specified shapes, including the emergent "polka dot" style in SA (4th column from left) and widening or narrowing "strokes" using ACORD (rightmost column, top and center).

(attending robotics conferences and events regularly). The study lasted approximately 45 minutes and participants were compensated $15. Of the 24 participants, the data from one participant was excluded due to non-participation (ignoring the robot's behavior and providing only uniform feedback on all surveys). This left data from $n = 23$ participants for analysis. The study procedure was approved by the Tufts University IRB.

Procedure Participants provided informed consent then took a background survey. The experimenter then explained the task and control in the conditions, including allowing participants to practice with SA and ACORD. In each condition, participants painted the house shape and then the heart shape, then filled out a survey about that condition. Conditions were fully counterbalanced within subjects. Finally, participants completed a post-study survey, were thanked, and given compensation.

Outcome Measures The post-condition survey included NASA TLX [1] and UTAUT [53] surveys. We adjusted the scale of all questions to a 5-item Likert-scale. We also asked two other Likert-scale questions: *I had control over the robot's behavior* and *I could express myself through the robot*, and an open response question: *How much do you feel the robot's ability to complete the task depended on your input?* The post-study survey had participants rank each condition based on their preference, the ability to express themselves, the perceived reliability of how well the robot traced the shape, and which mechanism (e.g. controller or sliders) they preferred. In addition, it asked two open response questions: a request for general comments and the question *how could the interactions be improved?*

We evaluated two quantitative metrics for how reliably the shape was traced. For each painting, we calculated the *coverage*, or percentage of the red line that remained visible in the image after the task was complete. We also calculated the *consistency*, or the coverage of the red line after applying translations and rotations of the painting to best align with the shape of the red line.

Hypotheses We expect that ACORD will give users control over the robot's behavior while still effectively completing the task, as users have more direct control than RL but less than that of SA. Thus, we expect that ACORD will be the most preferred approach and that it will give users feelings of slightly less control as SA while having similar performance to RL. This results in three hypotheses:

H1: Users will prefer to interact with ACORD over SA and RL.

H2: Users of ACORD will feel more in control of the robot than in RL but less than in SA.

H3: RL will be objectively and subjectively the most reliable, ACORD the second most and SA the least.

5 RESULTS

To analyze the data, we use Bayesian statistics following the interpretation scheme presented in [51]: a Bayes Factor (BF) between 3 and 10 we interpret as "moderate evidence" for the alternative hypothesis, between 10 and 30 as "strong evidence," and 30 or above as "very strong evidence." To evaluate the post-study survey data, we encoded responses as pairwise comparisons between two of the three conditions. For each comparison, the rank was encoded as 1 if the "left" condition was preferred, -1 if the "right" condition was preferred, and 0 if the participant ranked the two conditions equally. To analyze this data, we used a Bayesian Wilcoxon Signed Ranked test with a Cauchy prior distribution with $r = 1/\sqrt{2}$. To analyze the Likert scale data , we used a Bayesian Repeated Measures ANOVA. We used a Bayesian Paired Samples T-Test to analyze the coverage and consistency metrics.

User preferences We find strong evidence that ACORD is preferred over RL (**BF**=17.16) and anecdotal evidence that people prefer SA over RL (**BF**=2.11). There is strong evidence that people found ACORD more fun than RL (**BF**=79.87) and moderate evidence people found SA more fun than RL (**BF**=5.03). These results provide support for ACORD being preferred over RL while being no less preferred than SA. We also find a trend towards ACORD being

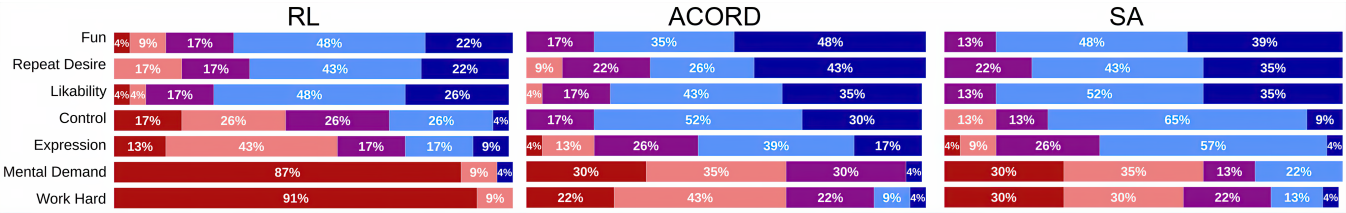

Figure 5: Responses to post-condition 5-point Likert scale questions. The darkest blue represents "strongly agree" or, in the case of Mental Demand, "very high." The darkest red represents "strongly disagree" or, in the case of Mental Demand "very low."

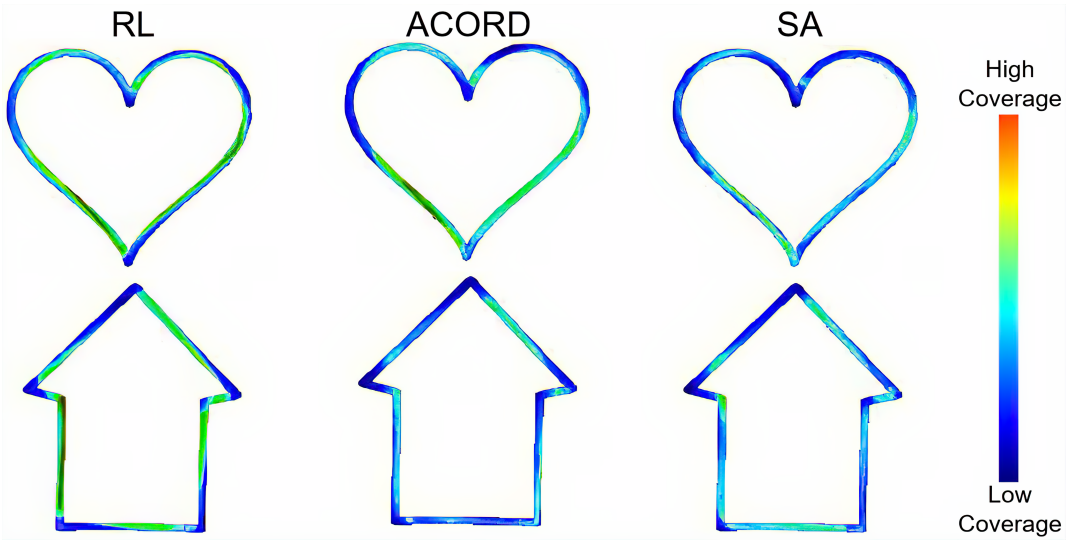

Figure 6: Heatmaps depicting the *consistency* of each approach sorted left to right from most consistent overall to least consistent. The heatmap consists of the participant's paintings layered on top each other after being shifted for maximal coverage. Areas of high coverage depict areas where many participants painted over, and vice versa for areas of low coverage.

preferred to a greater extent over RL than SA. Finally, we found that users rated RL as much less mentally demanding than SA and ACORD (**BF**=112.87 and **BF**=45.92 respectively), and much less hard work (**BF**>10000 and **BF**>10000), although the previous results suggest this was not a significant factor in user preferences. These findings partially support **H1** and directly support that ACORD provides at least as much benefit to user experience as SA.

User Control and Expression In the post study-survey we find strong evidence that people find ACORD and SA more expressive than RL (**BF**=18.40 and **BF**=13.65) and similarly for the post-condition survey measure of expressiveness (**BF**=23.38 and **BF**=40.31). Users also found a greater sense of control with ACORD and SA (**BF**=6318.61 and **BF**=40.31). There is anecdotal evidence that users reported more control in ACORD than SA (**BF**=2) and differences between the two were often commented on in open-ended responses. These results support the first part of **H2**, that users felt more in control in ACORD than in RL, however our results suggest that some users may have felt an even *greater* sense of control in ACORD than in SA.

Quantitative Painting Analysis We find on average, across both shapes, ACORD and SA had better coverage than RL (**BF**>10000 and **BF**=1095.2), likely due to the persistent offset in the RL condition caused by *bristle drag* of the brush. We account for misalignment by computing the maximum coverage found over small translations and rotations of the template, which we refer to as consistency. As expected, RL has better consistency than SA and ACORD in both shapes and, in general, the normalized sum across both shapes (**BF**>10000). While SA has higher consistency in the house shape (**BF**=1884.64), ACORD has much higher consistency in the heart shape and a higher consistency overall (**BF**>10000 and **BF**=11.67). A visualization of the consistency results can be found in Figure 6. According to our two reliability metrics, **H3** is supported by the consistency metric and not by the coverage metric. The coverage findings, however, showcased how a human in the loop can use the flexibility of added control to compensate for execution-time limitations in pre-trained RL models.

Qualitative Results Figure 4 shows paintings from each condition that are representative of the different painting styles found and the *emergent behaviors* that users demonstrated. With ACORD, we see the emergent behavior of brush strokes, where users moved both sliders quickly to make a specific stroke. In SA, some users made polka dots by bringing the brush up as much as they could, releasing the joystick, then letting the assistance bring the brush back to the paper. This was a surprising use of SA and goes against the task description of tracing the shape, yet gave users who figured this out a new way of expressing themselves and highlights that users had a desire for control and creativity in the task. While both ACORD and SA enabled this control, many users emphasized "consistency" and "ease of use" when describing ACORD; in contrast, users described SA as "mentally demanding" or "too sensitive."

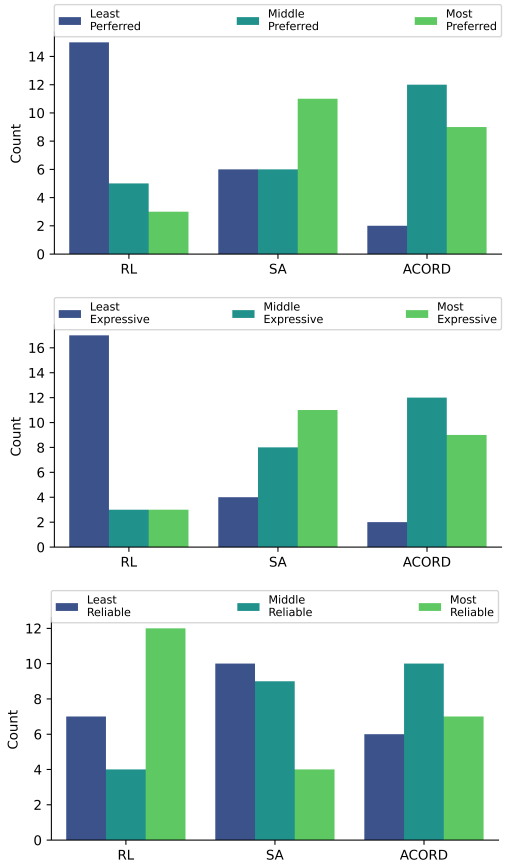

Figure 7: Results of the post study surveys. Users ranked each condition based one their preference (top), perceived expressive potential (mid), and perceived reliability (bottom).

Some users did not enjoy that ACORD required "shifting their eyes" from the screen to the robot, although of course this is an issue with the interface and not with ACORD itself. RL was criticized for not being able to adjust the style in real time; however, multiple users said it would be ideal for a "mass production" setting.

6 DISCUSSION

Online behavior modification describes an interaction in which a user has control over how an otherwise autonomous robot completes a task. While prior work has offered various algorithmic avenues to fulfill this type of user control, such as GCRL or Skill Learning, they have been formulated in robot-centered ways and lack validation in terms of usability and acceptance by actual users. In contrast, online behavior modification is a user-centric formulation that can leverage the benefits of these approaches to empower users in ways that can be systematically tested and compared.

Online behavior modification occupies a novel place within approaches to combine autonomous execution with human input. Our user study compared the ACORD algorithm to both a library of autonomous RL policies and a version of SA modified for a multi-goal setting where different styles represent different goals. We validate that ACORD can be used to adjust the style of a robots behavior and is perceived favorably by users. Our study shows that ACORD provides high levels of perceived control and expressiveness, as SA does, while being easier to use. There are also key technical and theoretical differences between online behavior modification and SA. In the context of SA, the task-level goal is unknown, and the robot, through an interpretation of the user's control signal, is attempting to infer the goal of the task. In contrast, in online behavior modification, the task-level goal is known, and the purpose is to maximize the user's control over how the robot autonomously completes that task. SA also requires the user to operate directly in the robot's action-space defined for the task, while algorithms such as ACORD build a separate new space for user input. In a larger system, online behavior modification algorithms like ACORD could work *with* SA, for example by using an SA system to infer *where* the user wants to go, and ACORD to give the user control over *how* the robot gets there. This opens up various directions for future research, both studying and comparing different algorithms for online behavior modification, as well as how online behavior modification may fit into or be combined with other paradigms.

Limitations An assumption in this work is that the designers of the system *know which axes of behavior people care about for the task*. This could be resolved by working with users to understand which behavior features they wish to adjust. Future work might also develop a general understanding of the types of features that users most want to adjust for a given task or types of tasks. Another limitation of the study is that we only considered m=2 behavior parameters to adjust. Osa et al. [40] have shown that the diversity-based methods ACORD is partially based on can learn effectively with up to 25 discrete latent variables. However, a large number of latent variables may impede the usability and interpretability of the system. Thus, more work is needed to understand how users interact with more numerous and abstract features. While ACORD was sufficiently efficient to be deployed on a real robot and be used by real users, the algorithm is relatively sample-inefficient (about 3 hours of fine-tuning after training in simulation). Future work could improve ACORD's efficiency by leveraging other techniques, such as hindsight and Constrained MDPs [3, 4]. Lastly, although online behavior modification entails the robot avoid task failures, that specification may not be sufficient for saftey-critical scenarios unless, potentially, combined with safe RL methods [2, 19, 33].

Conclusion This paper introduced the online behavior modification formulation, in which a user has control over *how* an otherwise-autonomous robot completes a task. Leveraging robot-centered algorithmic approaches for varying robot behavior, we proposed ACORD, a user-centered behavior diversity inspired algorithm that explicitly allows users continuous control over behavior features of a robot. We demonstrate ACORD's applicability to online behavior modification in simulation prior to deploying it in a user study. Interacting using ACORD was strongly preferred over selecting among RL policies, likely due to its creative potential and real-time control element, while its task accuracy and ease of use outperformed SA, in addition to being usable in tasks for which SA is not appropriate. This work highlights how human-centered formulations of robot learning can be used to enhance user experience with robots and opens directions for future research in this area.

ACKNOWLEDGMENTS

The work described here was supported in part by the US National Science Foundation (IIS-2132887).

REFERENCES

[1] [n. d.]. TLX @ NASA Ames - Home. https://humansystems.arc.nasa.gov/groups/tlx/

[2] Mohammed Alshiekh, Roderick Bloem, Ruediger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. 2017. Safe Reinforcement Learning via Shielding. https://doi.org/10.48550/arXiv.1708.08611 arXiv:1708.08611 [cs].

[3] Eitan Altman. 2021. *Constrained Markov Decision Processes: Stochastic Modeling* (1 ed.). Routledge, Boca Raton. https://doi.org/10.1201/9781315140223

[4] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. 2018. Hindsight Experience Replay. http://arxiv.org/abs/1707.01495 arXiv:1707.01495 [cs].

[5] Christian Arzate Cruz and Takeo Igarashi. 2020. A Survey on Interactive Reinforcement Learning: Design Principles and Open Challenges. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*. ACM, Eindhoven Netherlands, 1195–1209. https://doi.org/10.1145/3357236.3395525

[6] Chandrayee Basu, Mukesh Singhal, and Anca D. Dragan. 2018. Learning from Richer Human Guidance: Augmenting Comparison-Based Learning with Feature Queries. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. 132–140. https://doi.org/10.1145/3171221.3171284 arXiv:1802.01604 [cs].

[7] Erdem Biyik, Nicolas Huynh, Mykel Kochenderfer, and Dorsa Sadigh. 2020. Active Preference-Based Gaussian Process Regression for Reward Learning. In *Robotics: Science and Systems XVI*. Robotics: Science and Systems Foundation. https://doi.org/10.15607/RSS.2020.XVI.041

[8] Andreea Bobu, Marius Wiggert, Claire Tomlin, and Anca D. Dragan. 2021. Feature Expansive Reward Learning: Rethinking Human Input. *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction* (March 2021), 216–224. https://doi.org/10.1145/3434073.3444667 arXiv: 2006.13208.

[9] Jake Brawer, Debasmita Ghose, Kate Candon, Meiying Qin, Alessandro Roncone, Marynel Vázquez, and Brian Scassellati. 2023. Interactive Policy Shaping for Human-Robot Collaboration with Transparent Matrix Overlays. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, Stockholm Sweden, 525–533. https://doi.org/10.1145/3568162.3576983

[10] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI Gym. http://arxiv.org/abs/1606.01540 arXiv:1606.01540 [cs].

[11] Erdem Bıyık, Dylan P. Losey, Malayandi Palan, Nicholas C. Landolfi, Gleb Shevchuk, and Dorsa Sadigh. 2022. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research* 41, 1 (Jan. 2022), 45–67. https://doi.org/10.1177/02783649211041652 Publisher: SAGE Publications Ltd STM.

[12] Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. 2021. Goal-Conditioned Reinforcement Learning with Imagined Subgoals. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR, 1430–1440. https://proceedings.mlr.press/v139/chane-sane21a.html ISSN: 2640-3498.

[13] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. http://arxiv.org/abs/1706.03741 Number: arXiv:1706.03741 arXiv:1706.03741 [cs, stat].

[14] Matei Ciocarlie, Kaijen Hsiao, Adam Leeper, and David Gossow. 2012. Mobile manipulation through an assistive home robot. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5313–5320. https://doi.org/10.1109/IROS.2012.6385907 ISSN: 2153-0866.

[15] Antoine Cully. 2019. Autonomous skill discovery with quality-diversity and unsupervised descriptors. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, Prague Czech Republic, 81–89. https://doi.org/10.1145/3321707.3321804

[16] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. 2018. Diversity is All You Need: Learning Skills without a Reward Function. http://arxiv.org/abs/1802.06070 Number: arXiv:1802.06070 arXiv:1802.06070 [cs].

[17] Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R. Salakhutdinov. 2022. Contrastive Learning as Goal-Conditioned Reinforcement Learning. *Advances in Neural Information Processing Systems* 35 (Dec. 2022), 35603–35620. https://proceedings.neurips.cc/paper_files/paper/2022/hash/e7663e974c4ee7a2b475a4775201ce1f-Abstract-Conference.html

[18] Matthew C. Fontaine and Stefanos Nikolaidis. 2021. Differentiable Quality Diversity. http://arxiv.org/abs/2106.03894 Number: arXiv:2106.03894 arXiv:2106.03894 [cs].

[19] Javier Garcıa and Fernando Fernandez. [n. d.]. A Comprehensive Survey on Safe Reinforcement Learning. ([n. d.]).

[20] Deepak Gopinath, Siddarth Jain, and Brenna D. Argall. 2017. Human-in-the-Loop Optimization of Shared Autonomy in Assistive Robotics. *IEEE Robotics and Automation Letters* 2, 1 (Jan. 2017), 247–254. https://doi.org/10.1109/LRA.2016.2593928 Conference Name: IEEE Robotics and Automation Letters.

[21] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. *arXiv:1801.01290 [cs, stat]* (Aug. 2018). http://arxiv.org/abs/1801.01290

[22] Shervin Javdani, Henny Admoni, Stefania Pellegrinelli, Siddhartha S. Srinivasa, and J. Andrew Bagnell. 2018. Shared autonomy via hindsight optimization for teleoperation and teaming. *The International Journal of Robotics Research* 37, 7 (June 2018), 717–742. https://doi.org/10.1177/0278364918776060 Publisher: SAGE Publications Ltd STM.

[23] Shervin Javdani, Siddhartha Srinivasa, and Andrew Bagnell. 2015. Shared Autonomy via Hindsight Optimization. In *Robotics: Science and Systems XI*. Robotics: Science and Systems Foundation. https://doi.org/10.15607/RSS.2015.XI.032

[24] Parham M. Kebria, Hamid Abdi, Mohsen Moradi Dalvand, Abbas Khosravi, and Saeid Nahavandi. 2019. Control Methods for Internet-Based Teleoperation Systems: A Review. *IEEE Transactions on Human-Machine Systems* 49, 1 (Feb. 2019), 32–46. https://doi.org/10.1109/THMS.2018.2878815 Conference Name: IEEE Transactions on Human-Machine Systems.

[25] David Kent, Carl Saldanha, and Sonia Chernova. 2017. A Comparison of Remote Robot Teleoperation Interfaces for General Object Manipulation. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction (HRI '17)*. Association for Computing Machinery, New York, NY, USA, 371–379. https://doi.org/10.1145/2909824.3020249

[26] W. Bradley Knox and Peter Stone. 2009. Interactively shaping agents via human reinforcement: the TAMER framework. In *Proceedings of the fifth international conference on Knowledge capture - K-CAP '09*. ACM Press, Redondo Beach, California, USA, 9. https://doi.org/10.1145/1597735.1597738

[27] N. Koenig and A. Howard. 2004. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, Vol. 3. 2149–2154 vol.3. https://doi.org/10.1109/IROS.2004.1389727

[28] Saurabh Kumar, Aviral Kumar, Sergey Levine, and Chelsea Finn. 2020. One Solution is Not All You Need: Few-Shot Extrapolation via Structured MaxEnt RL. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., 8198–8210. https://proceedings.neurips.cc/paper/2020/hash/5d151d1059a6281335a10732fc49620e-Abstract.html

[29] Minghuan Liu, Menghui Zhu, and Weinan Zhang. 2022. Goal-Conditioned Reinforcement Learning: Problems and Solutions. http://arxiv.org/abs/2201.08299 arXiv:2201.08299 [cs].

[30] Björn Lütjens, Michael Everett, and Jonathan P. How. 2019. Safe Reinforcement Learning With Model Uncertainty Estimates. In *2019 International Conference on Robotics and Automation (ICRA)*. 8662–8668. https://doi.org/10.1109/ICRA.2019.8793611 ISSN: 2577-087X.

[31] Ajay Mandlekar, Yuke Zhu, Animesh Garg, Jonathan Booher, Max Spero, Albert Tung, Julian Gao, John Emmons, Anchit Gupta, Emre Orbay, Silvio Savarese, and Li Fei-Fei. 2018. ROBOTURK: A Crowdsourcing Platform for Robotic Skill Learning through Imitation. In *Proceedings of The 2nd Conference on Robot Learning*. PMLR, 879–893. https://proceedings.mlr.press/v87/mandlekar18a.html ISSN: 2640-3498.

[32] Gabriel B. Margolis and Pulkit Agrawal. 2022. Walk These Ways: Tuning Robot Control for Generalization with Multiplicity of Behavior. http://arxiv.org/abs/2212.03238 arXiv:2212.03238 [cs, eess].

[33] Daniel Marta, Christian Pek, Gaspar I. Melsión, Jana Tumova, and Iolanda Leite. 2022. Human-Feedback Shield Synthesis for Perceived Safety in Deep Reinforcement Learning. *IEEE Robotics and Automation Letters* 7, 1 (Jan. 2022), 406–413. https://doi.org/10.1109/LRA.2021.3128237 Conference Name: IEEE Robotics and Automation Letters.

[34] Oier Mees, Markus Merklinger, Gabriel Kalweit, and Wolfram Burgard. 2020. Adversarial Skill Networks: Unsupervised Robot Skill Learning from Video. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 4188–4194. https://doi.org/10.1109/ICRA40945.2020.9196582 ISSN: 2577-087X.

[35] Christopher Mower, Joao Moura, and Sethu Vijayakumar. 2021. Skill-based Shared Control. In *Robotics: Science and Systems XVII*. Robotics: Science and Systems Foundation. https://doi.org/10.15607/RSS.2021.XVII.028

[36] Fabio Muratore, Felix Treede, Michael Gienger, and Jan Peters. 2018. Domain Randomization for Simulation-Based Policy Optimization with Transferability Assessment. In *Proceedings of The 2nd Conference on Robot Learning*. PMLR, 700–713. https://proceedings.mlr.press/v87/muratore18a.html ISSN: 2640-3498.

[37] Vivek Myers, Erdem Biyik, Nima Anari, and Dorsa Sadigh. 2022. Learning Multimodal Rewards from Rankings. In *Proceedings of the 5th Conference on Robot Learning*. PMLR, 342–352. https://proceedings.mlr.press/v164/myers22a.html ISSN: 2640-3498.

[38] Heramb Nemlekar, Neel Dhanaraj, Angelos Guan, Satyandra K. Gupta, and Stefanos Nikolaidis. 2023. Transfer Learning of Human Preferences for Proactive Robot Assistance in Assembly Tasks. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, Stockholm Sweden, 575–583. https://doi.org/10.1145/3568162.3576965

[39] Benjamin A. Newman, Reuben M. Aronson, Siddhartha S. Srinivasa, Kris Kitani, and Henny Admoni. 2022. HARMONIC: A multimodal dataset of assistive human–robot collaboration. *The International Journal of Robotics Research* 41, 1 (Jan. 2022), 3–11. https://doi.org/10.1177/02783649211050677 Publisher: SAGE

Publications Ltd STM.

[40] Takayuki Osa, Voot Tangkaratt, and Masashi Sugiyama. 2022. Discovering diverse solutions in deep reinforcement learning by maximizing state–action-based mutual information. *Neural Networks* 152 (Aug. 2022), 90–104. https://doi.org/10.1016/j.neunet.2022.04.009

[41] Carolina Passenberg, Angelika Peer, and Martin Buss. 2010. A survey of environment-, operator-, and task-adapted controllers for teleoperation systems. *Mechatronics* 20, 7 (Oct. 2010), 787–801. https://doi.org/10.1016/j.mechatronics.2010.04.005

[42] Benjamin Pitzer, Michael Styer, Christian Bersch, Charles DuHadway, and Jan Becker. 2011. Towards perceptual shared autonomy for robotic mobile manipulation. In *2011 IEEE International Conference on Robotics and Automation*. 6245–6251. https://doi.org/10.1109/ICRA.2011.5980259 ISSN: 1050-4729.

[43] Justin K. Pugh, Lisa B. Soros, and Kenneth O. Stanley. 2016. Quality Diversity: A New Frontier for Evolutionary Computation. *Frontiers in Robotics and AI* 3 (2016). https://www.frontiersin.org/article/10.3389/frobt.2016.00040

[44] Ellis Ratner, Dylan Hadfield-Menell, and Anca D. Dragan. 2018. Simplifying Reward Design through Divide-and-Conquer. http://arxiv.org/abs/1806.02501 Number: arXiv:1806.02501 arXiv:1806.02501 [cs].

[45] Siddharth Reddy, Anca D. Dragan, and Sergey Levine. 2018. Shared Autonomy via Deep Reinforcement Learning. http://arxiv.org/abs/1802.01744 Number: arXiv:1802.01744 arXiv:1802.01744 [cs].

[46] Mario Selvaggio, Marco Cognetti, Stefanos Nikolaidis, Serena Ivaldi, and Bruno Siciliano. 2021. Autonomy in Physical Human-Robot Interaction: A Brief Survey. *IEEE Robotics and Automation Letters* 6, 4 (Oct. 2021), 7989–7996. https://doi.org/10.1109/LRA.2021.3100603 Conference Name: IEEE Robotics and Automation Letters.

[47] Isaac Sheidlower, Allison Moore, and Elaine Short. 2022. Keeping Humans in the Loop: Teaching via Feedback in Continuous Action Space Environments. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 863–870. https://doi.org/10.1109/IROS47612.2022.9982282 ISSN: 2153-0866.

[48] Bryon Tjanaka, Matthew C. Fontaine, Julian Togelius, and Stefanos Nikolaidis. 2022. Approximating Gradients for Differentiable Quality Diversity in Reinforcement Learning. http://arxiv.org/abs/2202.03666 Number: arXiv:2202.03666 arXiv:2202.03666 [cs].

[49] Bryon Tjanaka, Matthew C. Fontaine, Yulun Zhang, Sam Sommerer, Nathan Dennler, and Stefanos Nikolaidis. 2021. pyribs: A bare-bones Python library for quality diversity optimization. https://github.com/icaros-usc/pyribs Publication Title: GitHub repository.

[50] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. 2017. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. https://doi.org/10.48550/arXiv.1703.06907 arXiv:1703.06907 [cs].

[51] Johnny van Doorn, Don van den Bergh, Udo Böhm, Fabian Dablander, Koen Derks, Tim Draws, Alexander Etz, Nathan J. Evans, Quentin F. Gronau, Julia M. Haaf, Max Hinne, Šimon Kucharský, Alexander Ly, Maarten Marsman, Dora Matzke, Akash R. Komarlu Narendra Gupta, Alexandra Sarafoglou, Angelika Stefan, Jan G. Voelkel, and Eric-Jan Wagenmakers. 2021. The JASP guidelines for conducting and reporting a Bayesian analysis. *Psychonomic Bulletin & Review* 28, 3 (June 2021), 813–826. https://doi.org/10.3758/s13423-020-01798-5

[52] Sanne van Waveren, Christian Pek, Jana Tumova, and Iolanda Leite. 2022. Correct Me If I'm Wrong: Using Non-Experts to Repair Reinforcement Learning Policies. In *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction (HRI '22)*. IEEE Press, Sapporo, Hokkaido, Japan, 493–501.

[53] Viswanath Venkatesh, Michael G. Morris, Gordon B. Davis, and Fred D. Davis. 2003. User Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly* 27, 3 (2003), 425–478. https://doi.org/10.2307/30036540 Publisher: Management Information Systems Research Center, University of Minnesota.

[54] Nick Walker, Kevin Weatherwax, Julian Allchin, Leila Takayama, and Maya Cakmak. 2020. Human Perceptions of a Curious Robot that Performs Off-Task Actions. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, Cambridge United Kingdom, 529–538. https://doi.org/10.1145/3319502.3374821

[55] Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. 2022. A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems* 135 (Oct. 2022), 364–381. https://doi.org/10.1016/j.future.2022.05.014

[56] Yang Xing, Chen Lv, Dongpu Cao, and Peng Hang. 2021. Toward human-vehicle collaboration: Review and perspectives on human-centered collaborative automated driving. *Transportation Research Part C: Emerging Technologies* 128 (July 2021), 103199. https://doi.org/10.1016/j.trc.2021.103199

[57] Matthew Zurek, Andreea Bobu, Daniel S. Brown, and Anca D. Dragan. 2021. Situational Confidence Assistance for Lifelong Shared Autonomy. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE Press, Xi'an, China, 2783–2789. https://doi.org/10.1109/ICRA48506.2021.9561839