Adapting Robotic Systems to User Control

by

Upasana Biswas

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved June 2023 by the
Graduate Supervisory Committee:

Yu Zhang, Chair
Subbarao Kambhampati
Spring Berman
Lantao Liu

ARIZONA STATE UNIVERSITY

August 2023

ABSTRACT

In this work, I propose to bridge the gap between human users and adaptive control of robotic systems. The goal is to enable robots to consider user feedback and adjust their behaviors. A critical challenge with designing such systems is that users are often non-experts, with limited knowledge about the robot's hardware and dynamics. In the domain of human-robot interaction, there exist different modalities of conveying information regarding the desired behavior of the robot, most commonly used are demonstrations, and preferences. While it is challenging for non-experts to provide demonstrations of robot behavior, works that consider preferences expressed as trajectory rankings lead to users providing noisy and possibly conflicting information, leading to slow adaptation or system failures. The end user can be expected to be familiar with the dynamics and how they relate to their desired objectives through repeated interactions with the system. However, due to inadequate knowledge about the system dynamics, it is expected that the user would find it challenging to provide feedback on all dimension's of the system's behavior at all times. Thus, the key innovation of this work is to enable users to provide partial instead of completely specified preferences as with traditional methods that learn from user preferences. In particular, I consider partial preferences in the form of preferences over plant dynamic parameters, for which I propose Adaptive User Control (AUC) of robotic systems. I leverage the correlations between the observed and hidden parameter preferences to deal with incompleteness. I use a sparse Gaussian Process Latent Variable Model formulation to learn hidden variables that represent the relationships between the observed and hidden preferences over the system parameters. This model is trained using Stochastic Variational Inference with a distributed loss formulation. I evaluate AUC in a custom drone-swarm environment and several domains from DeepMind control suite. I compare AUC with the state-of-the-art preference-based reinforcement

learning methods that are utilized with user preferences. Results show that AUC outperforms the baselines substantially in terms of sample and feedback complexity.

ACKNOWLEDGMENTS

I would like to take this opportunity to express my deepest gratitude to the individuals who have played a significant role in the completion of my master's thesis.

First and foremost, I am immensely grateful to my advisor, Tony, for their unwavering support and invaluable guidance throughout this research. Their dedication and commitment to my success have been truly exceptional and instrumental in shaping the direction of my study, investing countless hours of their own day to ensure that we produce the best possible output. Their relentless encouragement and constructive feedback have pushed me to surpass my own expectations and do better. Moreover, they have served as an inspiring role model, demonstrating the qualities of a dedicated researcher— always striving for improvement and fostering an unwavering curiosity. It is through their mentorship that I have discovered the immense joy and fulfillment that the world of research offers. I will forever cherish the lively and intellectually stimulating discussions we had, as they were the highlights of my journey. Their guidance has not only shaped the outcome of my master's thesis but has also contributed to my personal and intellectual growth.

I would also like to extend my appreciation to the members of my thesis committee, Dr. Kambhampati, Dr. Berman and Dr. Liu, for their valuable feedback and constructive criticism, which have greatly enriched the quality of my work. Their expertise and insights have contributed immensely to the refinement of my research, and I am grateful for their time and efforts in reviewing my thesis.

In addition, I would like to express my heartfelt gratitude to my parents and my sister, who have always believed in me and supported me unconditionally. They have consistently encouraged me to reach higher and provided me with the strength and confidence to pursue my ambitions. Their trust in my capabilities has been a driving

force behind my accomplishments, and I am forever grateful for their unconditional love and support. I am also grateful to my friends, who have stood by me as pillars of support, helping me regain focus and pick myself up during challenging times. Their presence in my life has been a source of strength, and I am sincerely thankful for their friendship.

Lastly, I extend my appreciation to Arizona State University for providing me with the necessary resources, facilities, and academic environment that have contributed to the successful completion of this thesis. The university's commitment to excellence has played a significant role in shaping my academic journey and fostering an environment conducive to research and intellectual growth.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

The integration of robots into human workspace has drawn significant attention due to their potential of complementing human capabilities, by tackling tasks that are deemed dull, dirty, or dangerous (i.e., the 3Ds of industry). For successful integration, it is crucial to ensure that humans perceive the presence of robots as socially responsible . For example, research in Human-Robot Interaction (HRI) has shown that factors such as safety, trust, transparency, predictability, and ease of use can significantly influence the acceptance of robotic technologies de Graaf and Ben Allouch (2013); Heerink *et al.* (2010); Shahrdar *et al.* (2019); de Graaf *et al.* (2016). A critical defining capability of such systems is to receive feedback from humans and adapt their behaviors accordingly Spencer *et al.* (2020); Sheridan (2016); Mitsunaga *et al.* (2008). In the domain of human-robot interaction, there exist different modalities of conveying information regarding the desired behavior of the robot, most commonly used are demonstrations , corrections and preferences . While demonstrations provided by an expert user are a natural way to demonstrate desired behavior to robots without any programming, methods that use them have limited applicability to tasks that require a level of expertise or knowledge that typical end users will not posses. Therefore, in order to implement a user-adaptive control which is generalizable to a variety of users and task domains, we utilize preferences expressed by non-expert end users as the basis of our work on Adaptive User Control (AUC). However, the remaining challenges in using user preferences for learning robot behavior are: 1) costly human feedback, and 2) noisy and conflicting feedback due to non-expert users, which we address in our work.

Control systems are often employed to manipulate the robot's motions and responses according to desired objectives in real-world. Behavior generated by such controllers often provide guarantees of stability, robustness, safety etc. Spielberg *et al.* (2019); Wang *et al.* (2018). Adapting robot control to user feedback is a well-researched problem in HRI, particularly physical-HRI where this feedback is physically embodied as force, either through demonstrations or shared human robot control. Contrasted with such approaches, we work on a user-adaptive control framework which employs preferences that may not involve physical HRI, which makes approaches relying on direct force feedback inapplicable.

On the other hand, preference-based learning approaches for behavior generation have also been studied extensively before Christiano *et al.* (2023); Warnell *et al.* (2018). One particular inspiring line of work is human-aware and explainable behavior generation Chakraborti *et al.* (2017); Zhang *et al.* (2017); Dragan *et al.* (2013); Hanni *et al.* (2023); Zakershahrak *et al.* (2018), which depends on not only the robot's dynamics but also the human's preferences (which pertain to explainability) for the robot's behavior. A mechanism is required to reconcile between the two factors for better HRI. Preferences may generally arise from multiple sources, such as reward preferences Gong and Zhang (2022), beliefs of the domain dynamics Gong and Zhang (2020); Reddy *et al.* (2018), a limited cognitive ability Choudhury *et al.* (2019), etc. Prior research has also studied how these preferences can be biased Gong and Zhang (2020) and dynamically changing Hanni and Zhang (2021). Our work will focus on the first two sources, which have been illustrated to be equivalent for behavior generation. The integration of such a preference-based framework with control theory is lacking.

Model Reference Adaptive Control (MRAC) is a control methodology with the objective of adapting uncertain plants to dynamic situations, and tracking a reference plant model, while modelling the uncertainties and disturbances in the system dynamics.

A possible approach of applying this methodology to user adaptive control with user preferences, is by treating the user as an environmental disturbance and accordingly modifying the plant and control system parameters. However, the influence of user preferences on the system dynamics would be challenging to model as noise. Moreover, preferences expressed as rankings between trajectories provide an impoverished way for humans to communicate their desired behavior objectives to the robot. The end user can be expected to be familiar with the plant dynamics and how they relate to these objectives through repeated interactions with the system .

The effectiveness of MRAC depends on the predicted system response based on the reference model, which generally does not consider human preferences adapting robotic control to user feedback . Therefore, in our work, we derive the reference plant model from user preferences expressed on the plant dynamic parameters (PDP's). There are major limitations when it comes to implementing such an approach with real-world users. Preferences on robot behavior are hidden in the user's mind and are challenging to retrieve accurately. Due to limited knowledge about the system, it is expected that the user would find it challenging to provide feedback on all dimension's of the system's behavior at all times. Even when assuming that users can fully specify their own preferences, they lack an adequate understanding of the system's functioning and the complexities of the environment to correctly guide the system at all times. This will likely lead to slow adaptation and sometimes failures. Consider the domain of autonomous driving. A vehicle initially operates at its manufacturer setting but has the ability to adapt its behavior based on the user's feedback. The user might focus only on a high-speed preference for the vehicle without considering the limitations of it's sensing abilities leading to collisions or unsafe movement at high speeds. This is undesirable for the user as well, but they might not consider this consequence while expressing their preference. In a similar way, suppose the user wants the vehicle to

maintain a larger distance to other vehicles, but does not wish the speed to reduce, and these behaviors are negatively correlated. However, if the user focuses on only one attribute, this leads to conflicting information for the learning system when it considers the human's feedback as complete. When the users are non-expert users, the problem only aggravates. Instead, we enable users to specify partial preferences, such as preferences for only the distance to the front vehicles or the speed in the example above.

In this work, we propose to bridge the gap between human users and adaptive control by enabling robots to learn from user preferences to adapt their behaviors. Applying an adaptive control framework (such as MRAC) will enable us to easily integrate critical system constraints (e.g., safety) while respecting the human preferences. We propose to integrate learning from human preferences and adaptive control while focusing on tackling the aforementioned challenges to ensure the practicality of such an integration. Our key innovation is to enable users to provide partial instead of complete preferences that are used in the traditional learning methods based on preferences Christiano *et al.* (2023); Warnell *et al.* (2018); Zhang *et al.* (2019); Lee *et al.* (2021a). Providing complete preferences on trajectories (i.e., deciding whether one trajectory is better than the other overall) is error-prone and introduces possibility of misguidance Wirth and Fürnkranz (2013); Chen *et al.* (2022); Pini *et al.* (2011) while providing partial preferences (i.e., feedback on certain PDP's only) tends to be much more intuitive Ajzen *et al.* (2004); Dubra *et al.* (2004); Cailloux and Destercke (2018). In our work, we consider partial preferences in the form of attributed preferences over a set of plant dynamics parameters (PDPs). The updated PDPs can then be considered in a MRAC framework to produce the final command vectors. There are major challenges which arise from attempting an integration of learning from preferences and adaptive control mainly time-delay in the system's transient response

and error convergence. However, these are existing challenges in the domain of MRAC and we present a discussion of how works which try to tackle these issues can be applied with the proposed AUC.

We represent the evolution of PDPs throughout a task environment using Gaussian Processes (GPs). We apply a sparse Gaussian process latent variable model (GPLVM) Lawrence (2003) formulation to learn hidden variables that represent the relationships between the observed and hidden attribute preferences over the PDPs. Stochastic variational inference (SVI) Hoffman *et al.* (2013) is used to train the GPLVM with a loss function distributed across feedback samples and attributes for expressing preferences. By allowing human users to provide partial preferences on the attributes, the system acknowledges that preferences may not always be fully specified. This flexibility enables users to express their preferences to the extent they are able, and the system can utilize this information to make appropriate modifications to the plant parameters. This approach enhances user experience and system adaptability, allowing for personalized adjustments based on individual preferences. The proposed approach, referred to as adaptive user control (AUC), is evaluated on multiple domains - a custom Drone Swarm environment and domains from DeepMind Control Suite Tassa *et al.* (2018). The performance of AUC is compared to the state-of-the-art preference-based RL methods. Results showed that AUC substantially outperforms the baselines in terms of sample efficiency. Given that the proposed methodology entails direct modification of control parameters, we explore how the balance between user preferences and system safety can be established through the integration of safety constraints aimed at preventing the system from entering failure states.

Chapter 2

RELATED WORKS

Adaptive Control

Adaptive control is a control methodology that allows a system to adjust its control parameters in response to changes in the environment or in the system dynamics itself, while maintaining stability Åström (1983). Applying this approach to multiple input multiple output (MIMO) systems faces the challenge of the dynamic coupling among the input and output signals Tao (2014), which are commonly dealt with by utilizing decoupling compensators to eliminate the interactions Bayoumi and Mo (1988); Liu *et al.* (2019) and then treating the original system as a collection of single input and single output (SISO) systems. This approach introduces further complexity due to the addition of the decoupler matrix. We face a similar but more challenging issue in our problem because of the existence of correlated PDP's. However, we address it by introducing hidden variables which represent the underlying correlations between the parameters and learn the mapping from the hidden variables to the parameters. Common applications of adaptive control in human-robot interaction include assistive robots, where adaptation is commonly implemented through iterative methods. Bae and Tomizuka (2012)proposes an iterative learning algorithm to adaptively learn from the user's performance and continuously update the joint impedance parameters to achieve the desired rehabilitation goal. In Force-controlled rehabilitation Calanca *et al.* (2014); Lee *et al.* (2019), forces are computed by assistive algorithms to achieve low-level control of human. However, adaptive control of these systems requires complete error specification and completeness of the reference model specification,

which is challenging to acquire from end users who are often non-experts in the domain with limited knowledge of system dynamics and relevant performance objectives.

Reinforcement learning has also been applied to process control to achieve robust and adaptive control Khan *et al.* (2012). Spielberg *et al.* (2019) and Wang *et al.* (2018) adapt deep reinforcement learning algorithms to achieve online learning of the control policy with direct interaction with the process while transfer learning is used in Petsagkourakis *et al.* (2019) to adapt a policy trained on a simulation model to novel environments. Sedighizadeh and Rezazadeh (2008) and Carlucho *et al.* (2017) train a controller in a simulation environment using a q-learning strategy to improve the adaptive performance of a PID controller. While Spielberg *et al.* (2019), Wang *et al.* (2018) and Petsagkourakis *et al.* (2019) achieve adaptive control in a model-free fashion, they require a large amount of interaction with the system or previously collected data to learn an effective control policy, which is infeasible for complex systems. Therefore, applying reinforcement learning to adaptive process control introduces the problem of sample inefficiency and the "curse of dimensionality". This problem is tackled by AUC as it greatly improves on feedback efficiency by utilizing the correlations between the PDP preferences in the learning process.

2.1 Preference-based Reinforcement Learning

Reinforcement learning (RL) represents a flexible approach for behavior learning and adaptation. However, the success of such algorithms depends on the accuracy of reward specification Kober *et al.* (2013); Mnih *et al.* (2016). Reward specification is challenging for domains with tacit-knowledge where the objectives are complex and difficult to specify Singh *et al.* (2009). In such cases, inverse reinforcement learning (IRL) can be used when demonstrations of the desired domain are available to extract a reward function for training through reinforcement learning Ng and Russell (2000);

Arora and Doshi (2021); Schaal (1996); Abbeel and Ng (2004). However, this approach requires the user to possess the necessary knowledge and hardware interface to control the system effectively to provide demonstrations. A similar issue prevents imitation learning Ross and Bagnell (2010) to be applied in many domains, even through it can be very sample efficient. More recently, the difficulty in reward specification is addressed via preference based learning Christiano *et al.* (2023); Warnell *et al.* (2018); Zhang *et al.* (2019); Lee *et al.* (2021a) where a parameterized reward function is retrieved from preferences over trajectory pairs. However, providing such preferences is error-prone, which leads to inconsistent or conflicting feedback information. The result is a learning challenge that is not unlike the temporal credit assignment problem in RL Wirth *et al.* (2017). Consequently, these methods often require extensive training and a large amount of data Wirth *et al.* (2017); Christiano *et al.* (2023).

2.2 Human-Robot Interaction

Adaptive control has often been applied in physical HRI Haddadin and Croft (2016), where the robot works in close contact with a human operator leading to physically embodied interactions commonly realized as forces. Robots leverage these physical interactions with the users and learn to improve their behavior De Santis *et al.* (2008); Musić and Hirche (2017); Argall and Billard (2010). Admittance control is commonly applied to ensure that the robot response must be compliant to the forces exerted by the human Okunev *et al.* (2012a); Newman (1992); Wang and Kosuge (2012). However, explicit force control introduces instability in the system Hogan (1985), which led to the development of impedance control Hogan (1984, 1985); Gonzalez and Widmann (1995). This method requires a strictly defined prior i.e. robot dynamics model and interaction model Buerger and Hogan (2007); Kazerooni *et al.* (1986), which restricts it's applications in human-robot interaction. The interaction model needs to consider

human dynamics as well as their interactions with the robot, which is challenging to model with a wide range of human users.

Applying admittance/impedance control to cooperative tasks involving humans is challenging due to constant parameters Okunev *et al.* (2012b). However, existing literature in adaptive admittance/impedance control mainly focuses on task-dependent parameter adaption Tsumugiwa *et al.* (2002); Tsetserukou *et al.* (2007); Duchaine and Gosselin (2007). Significant work has been undertaken to achieve adaptation of robots to human users by designing a cascaded control system . Following results from human-factor studies Wolpert *et al.* (1998); Kleinman *et al.* (1970); Suzuki and Furuta (2012), the inner loop handles the robot specific control by utilizing adaptable admittance/impedance control and the outer loop task-specific controller incorporates the human dynamics by estimating the human–robot transfer characteristics Ranatunga *et al.* (2017), reference trajectory adaptation Li *et al.* (2018) or by adding a system identifier for human dynamics Alqaudi *et al.* (2016). However, the major shortcoming of these approaches is that they adapt to the force/torque applied by the human user, which introduces the assumption that the user is familiar with the hardware and the task. Therefore, they are not applicable to tasks which are difficult for humans to demonstrate. To apply adaptive control to such applications, we introduce partial user preferences as a modality for human-robot interaction. Since it is not feasible to learn an analytical model structure for the human from such preferences, we cannot use adaptive impedance approaches to implement user-adaptive control.

Chapter 3

METHODOLOGY

To better explain our approach, we provide a running example in a drone swarm domain, which is implemented via a potential-field based scheme with two levels of plant behavior. Consider a navigation scenario where the plant dynamics model can be specified by the parameters corresponding to the following potential fields: centroid attractive field, centroid repulsive field, and obstacle repulsive field, along with a separate parameter that determines the number of neighbours in a local cluster (for determining the centroid). The problem is for us to query the user for their partial preferences in terms of these or a subset of these plant dynamics parameters (PDPs) at different parts of the environment. The goal is to learn to dynamically predict the PDP trajectories throughout the environment from user preferences expressed in terms of these parameters. In our work, we assume the existence of a plant behavior model where these plant dynamic parameters are fed back, and the plant output is modified.

Note that such a process resembles that in adaptive control.

3.1 Preliminaries

Preference-based Reinforcement Learning (PbRL)

PbRL aims to optimize an agent's behavior by learning from user feedback in the form of preference rankings between trajectories. Let us consider a Markov Decision Process (MDP) defined by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P}$ represents the transition dynamics, and $\mathcal{R}$ denotes the reward function. Given user preferences expressed as pairwise rankings $\tau_1 \succ \tau_2$, where

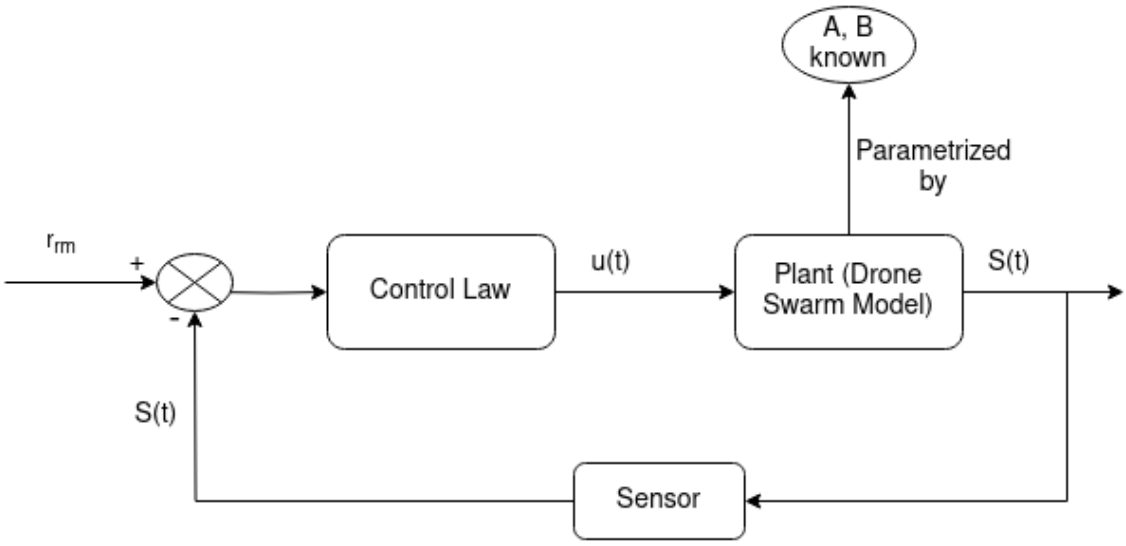


Figure 3.1: Initial System Dynamics Model

$\tau_i = (s_0, a_0), (s_1, a_1), ...(s_i, a_i)$ where i is the length of the trajectory, PbRL aim to learn a preference model $P(\tau_1 \succ \tau_2)$ that estimates the probability of the preference ranking based on the observed data. Typically, the Bradley-Terry model is used which captures the idea that the probability of preference is determined by the reward of the trajectory pair such that;

$$P(\tau_1 \succ \tau_2) = \frac{e^{r_{\tau_1}}}{e^{r_{\tau_1}} + e^{r_{\tau_2}}} \tag{3.1}$$

where $r_{\tau_i} = \sum_t R(s_t, a_t)$ where $[s_t, a_t] \in \tau_i$

Dynamic Model for Drone Swarm Domain

We consider a swarm of N drones, and write the plant dynamics model for each drone. For each drone, the plant behavior is the linear combination of the following behaviors: Goal Attractive Force(GAF), Obstacle Repulsive Force(ORF), Centroid Attractive Force(CAF), and Centroid Repulsive Force(CRF). Each behavior is given by a potential field, which is further parametrized by the PDP's: T_{OR}, T_{CA} and T_{CR} respectively. Each behavior outputs a two dimensional velocity vector along the x

and y axes. Each drone maintains a minimum altitude $\mathbf{z}$ and constant yaw alignment, according to a predefined setpoint $\theta_{\mathbf{z_{desired}}}$. The state of the plant $\mathcal{S}$ can be defined as :

$$\mathbf{S} = \begin{bmatrix} \mathbf{x} & \mathbf{y} \end{bmatrix}$$

where $\mathbf{x}, \mathbf{y}$ refers to the translation along the x and y axes, and $\theta_{\mathbf{z}}$ refers to the angular rotation along the z axes. Therefore, from **?**, the plant behavior can be denoted as:

$$\dot{\mathbf{S}} = \dot{\mathbf{S}}_{GAF} + \dot{\mathbf{S}}_{ORF} + \dot{\mathbf{S}}_{CAF} + \dot{\mathbf{S}}_{CRF} \tag{3.2}$$

$$\begin{aligned} \dot{\mathbf{x}} &= (D_x - \mathbf{x}) \cos\theta_{\mathbf{z}} + \exp\left(-\|\sigma\|\right) \cos\left(\theta_{obs}\right) + \mathbf{c}_x + \exp\left(-\|\mathbf{c}\|\right) \cos\left(\theta_{centroid}\right), \\ \dot{\mathbf{y}} &= (D_y - \mathbf{y}) \sin\theta_{\mathbf{z}} + \exp\left(-\|\sigma\|\right) \sin\left(\theta_{obs}\right) + \mathbf{c}_y + \exp\left(-\|\mathbf{c}\|\right) \sin\left(\theta_{centroid}\right) \end{aligned} \tag{3.3}$$

where $D = \sqrt{{D_x}^2 + {D_y}^2}$ denotes the total distance from the goal, $\sigma = \sqrt{{\sigma_x}^2 + {\sigma_y}^2}$, $\mathbf{c} = \sqrt{{\mathbf{c}_x}^2 + {\mathbf{c}_y}^2}$, $\theta_{obs} = \text{atan2}(\sigma_y, \sigma_x)$ and $\theta_{centroid} = \text{atan2}(\mathbf{c}_y, \mathbf{c}_x)$. From **?**, we can write:

$$\begin{aligned} \sigma_x = \mathbf{f}\left(T_{OR}, \mathbf{x}\right), \sigma_y = \mathbf{f}\left(T_{OR}, \mathbf{y}\right), \\ \mathbf{c}_x = \mathbf{f}\left(T_{CA}, T_{CR}, \mathbf{x}\right), \mathbf{c}_y = \mathbf{f}\left(T_{CA}, T_{CR}, \mathbf{y}\right) \end{aligned} \tag{3.4}$$

Using linearization techniques, we can approximate the plant dynamics model for the drone swarm domain as the model shown below, where $\mathbf{r}$ refers to the control output sent from the controller to the plant. Note that we assume the plant has no uncertainty. While an important component of MRAC is adapting the control parameters such that the uncertain plant tracks the reference model, in AUC, we focus on designing an adaptive control framework with a time-varying reference model and therefore assume a simpler plant model.

$$\dot{\mathbf{S}}(t) = -A\mathbf{S}(t) + B\mathbf{u}(t) \tag{3.5}$$

where A can be expressed as $\mathbf{f}\left(T_{OR}, T_{CA}, T_{CR}\right)$. Therefore, if the PDP's are changed, the plant dynamics model also changes. Here, the reference signal $\mathbf{r}_{rm}(t) = \theta_{\mathbf{z_{desired}}}$ is

assumed to be provided to utilize for feedback control. Using linear approximation on $\dot{\mathbf{S}}(t)$ as shown above, we write the control law as:

$$\mathbf{u}(t) \approx \phi_1 \mathbf{r}_{rm} - \phi_2 \mathbf{S}(t) \tag{3.6}$$

Model Reference Adaptive Control (MRAC) with User Preferences

The objective of MRAC is to make the controlled system behave as closely as possible to the desired reference model using an adaptive controller that adjusts its parameters in real-time. The adaptation process uses the error between the states of the controlled system and the states of the reference model. The objective of the controller is to track a reference model described as:

$$\dot{\mathbf{S}}_m(t) = -A_{rm}\mathbf{S}_{rm}(t) + B_{rm}\mathbf{r}_{rm} \tag{3.7}$$

$$error(e) = \mathbf{S}(t) - \mathbf{S}_{rm}(t) \tag{3.8}$$

In our work, we aim to combine MRAC with PbRL, to create an Adaptive User Control system that can update the plant parameters of the reference model according to partial preferences expressed by users. In a similar fashion to PbRL, we model the drone swarm environment as an MDP. However, contrasted to PbRL, the proposed system considers a user preference P as: $\tau_{1_{\mathcal{P}_1}} \succ \tau_{2_{\mathcal{P}_1}}, \tau_{1_{\mathcal{P}_2}} \succ \tau_{2_{\mathcal{P}_2}}, \ldots, \tau_{1_{\mathcal{P}_D}} \succ \tau_{2_{\mathcal{P}_D}}$ where D is the maximum number of controllable plant parameters : $\{T_{OR}, T_{CA}, T_{CR}\}$ for the drone swarm domain. AUC aims to learn the parameters for a reference plant model from preferences expressed, by utilizing the correlations between the PDP's and modelling them as GP's. In particular, A_{rm} is generated using the set of preferences P. In our formulation of adaptive control, we assume B to remain unchanged. Therefore, for AUC, the reference model derived from P can be expressed as :

$$\dot{\mathbf{S}}_m(t) = -A_{rm}\mathbf{S}_{rm}(t) + B\mathbf{r}_{rm} \tag{3.9}$$

The plant should track the reference plant model derived above, and therefore the adaptive control mechanism should update ϕ_I to reduce the tracker error e i.e. achieve stabilization of controller parameters.

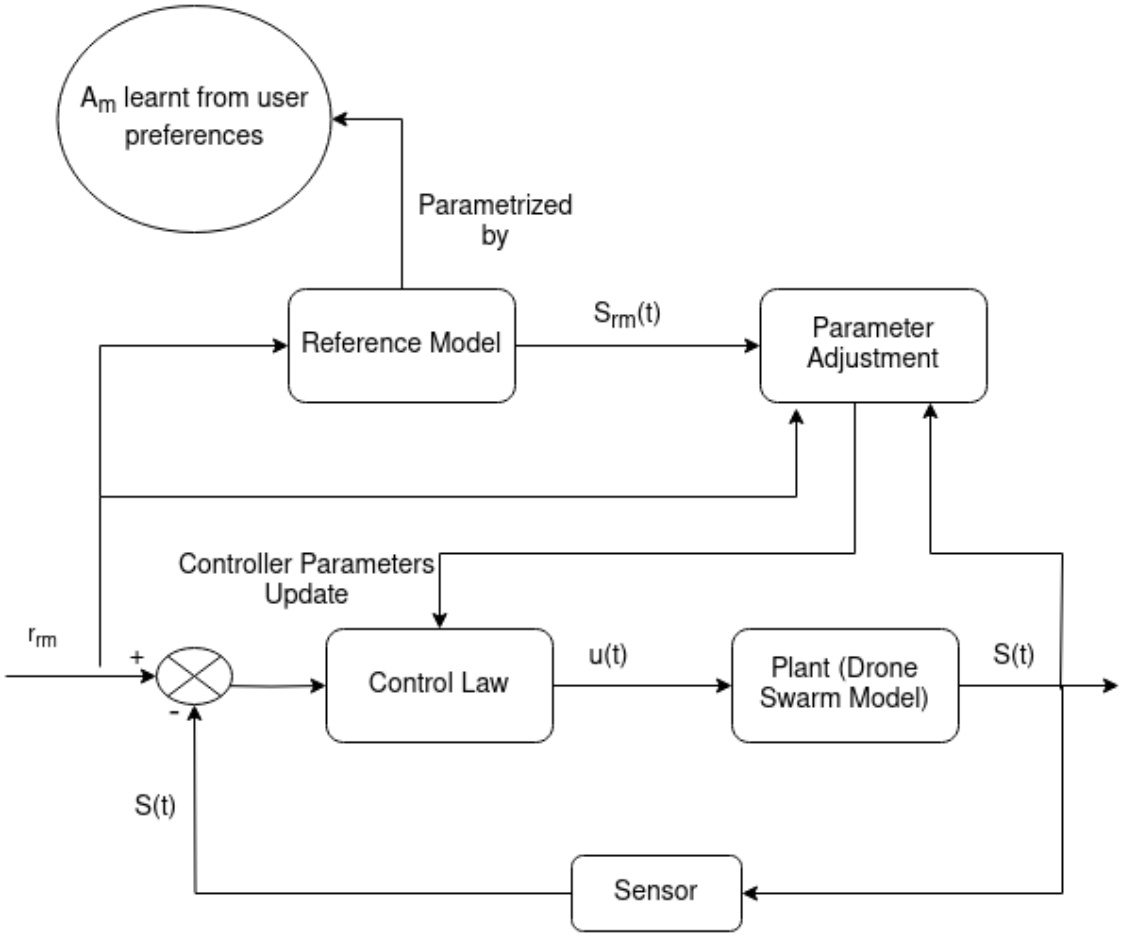


Figure 3.2: AUC with Adaptive System Dynamics Model

Derivation of Adaptation Law

From 3.5 and 3.6,

$$\begin{aligned} \dot{\mathbf{S}}(t) &= -A\mathbf{S}(t) + B(\phi_1 \mathbf{r}_{rm} - \phi_2 \mathbf{S}(t)), \\ \dot{\mathbf{S}}(t) &= (-A - B\phi_2)\mathbf{S}(t) + B\phi_1 \mathbf{r}_{rm} \end{aligned} \tag{3.10}$$

To reduce the error e, we derive the expression for the derivative of the error using 3.6, 3.9 and 3.10:

$$\begin{aligned}\dot{e} &= \dot{\mathbf{S}} - \dot{\mathbf{S}_{rm}} \\ &= (-A - B\phi_2)\mathbf{S}(t) + B\phi_1\mathbf{r}_{rm} + A_{rm}\mathbf{S}_{rm}(t) - B\mathbf{r}_{rm}\end{aligned} \tag{3.11}$$

Adding and subtracting $A_{rm}\mathbf{S}(t)$ to the above to get $\dot{e}$ in terms of e, we get:

$$\begin{aligned}\dot{e} &= A_{rm}(\mathbf{S}_{rm}(t) - \mathbf{S}(t)) + (A_{rm} - A - B\phi_2)\mathbf{S}(t) + (B\phi_1 - B)\mathbf{r}_{rm} \\ &= -A_{rm}e + (A_{rm} - A - B\phi_2)\mathbf{S}(t) + (B\phi_1 - B)\mathbf{r}_{rm}\end{aligned} \tag{3.12}$$

To derive the adaptive parameter update law using Lyapunov's Stability Theorem, we introduce a continuously differentiable positive definite function which can qualify as a Lyapunov candidate function and sensitivity parameter γ.

$$V(e, \phi_1, \phi_2) = \frac{e^2}{2} + \frac{(A_{rm} - A - B\phi_2)^2}{2B\gamma} + \frac{(B\phi_1 - B)^2}{2B\gamma} \tag{3.13}$$

The derivative $V(e, \dot{\phi_1}, \phi_2)$ should be negative semi-definite for the equilibrium point to be stable.

$$\begin{aligned}\dot{V} &= e\dot{e} - \frac{(A_{rm} - A - B\phi_2)B\dot{\phi_2}}{B\gamma} + \frac{(B\phi_1 - B)B\dot{\phi_1}}{B\gamma} \\ &= e(-A_{rm}e + (A_{rm} - A - B\phi_2)\mathbf{S}(t) + (B\phi_1 - B)\mathbf{r}_{rm}) - \frac{(A_{rm} - A - B\phi_2)B\dot{\phi_2}}{B\gamma} \\ &+ \frac{(B\phi_1 - B)B\dot{\phi_1}}{B\gamma} \\ &= -A_{rm}e^2 - \frac{(A_{rm} - A - B\phi_2)(\dot{\phi_2} - \gamma S(t)e)}{\gamma} + \frac{(B\phi_1 - B)(e\mathbf{r}_{rm} + \dot{\phi_1})}{\gamma}\end{aligned} \tag{3.14}$$

For $\dot{V} \leq 0$ in 3.14, we get the parameter update laws as:

$$\begin{aligned}\dot{\phi_2} &= \gamma S(t)e \\ \dot{\phi_1} &= -e\mathbf{r}_{rm}\end{aligned} \tag{3.15}$$

Asymptotic Tracking with time-varying Reference Model

The parameter adaptation laws derived above introduce significant oscillations in the controller parameters, due to the transient response of the system while the error converges to zero. Issues such an overshoot or convergence rate during the transient period are existing challenges in the domain of adaptive control and severely limit the practical application of such control schemes. As discussed above, while traditional MRAC approaches use a static reference plant model, AUC utilizes a learned time-varying reference model derived from human preferences. Consequently, the aforementioned issues due to the transient response are further exacerbated in our methodology. Significant research has been conducted to improve the transient system response in terms of oscillatory behavior and convergence speed of the tracking error for MRAC systems, such as nonadaptive high gain feedback Datta and Ioannou (1994); Sun (1991), switching control law Morse (1996) or a parameter dependent persistent excitation condition Arteaga and Tang (2002). Therefore, utilizing improved adaptive control schemes with modified adaptive laws and control architecture can help us improve the transient response time of AUC, thus allowing the system to deal with a time-varying reference model.

Another viable approach to address the challenges is to discretize the learned continuous Gaussian Process. By dividing the GP into partitions over time, each with a duration larger than the transient response period, we can approximate the time-varying reference model as a sequence of constant reference models. This discretization allows us to use the adaptive control laws for the reference model over each time period, facilitating the plant to roughly track the reference model derived from user preferences while ensuring stability of the control system. Extensive research has been conducted on the modification of reference models in model-reference adaptive control Stepanyan

and Krishnakumar (2012); Na *et al.* (2020); Gibson (2014). However, the aspect of time-varying modification is commonly overlooked or not adequately addressed in these investigations. Nguyen (2022) proposes a modified MRAC scheme with real-time updates to the reference model, to optimize the performance metric of the plant. This work is the closest in terms of dealing with a time-varying reference model, which is another novelty of AUC. In their work, the multi-objective performance optimization problem generates time-varying Riccati and Sylvester equations, the solutions to which produce time-varying controller parameters. We could integrate a similar approach in AUC, where a critical performance metric would be safety of the system. Permitting users to modify the system's behavior can benefit many applications domains to achieve the desired flexibility and adaptability of systems that cohabit with human users in complex environments. However, unrestricted modifications by non-expert users may inadvertently introduce safety risks. Thus, the pursuit of striking the delicate balance between user preferences and system safety is crucial to mitigate risks associated with system misuse, and can be addressed using the method proposed above.

3.2 GPLVM

We use the GPLVM formulation to represent the human feedback data. In our formulation, $\mathbf{P}$ represents the observed user feedback as $\mathbf{P} = \{P_n\}_{n=1}^{N}$ where N represents the size of the trajectory in the environment, $\mathbf{P} \in \mathbb{R}^{N \times I_n}$ where $I_n \in [1, D]$. Therefore, for the drone swarm domain, $D = 4$ and the user can provide preferences over any of the four controllable plant parameters.

Since these are correlated, we posit the existence of latent variables $\mathcal{X} = \{x_n\}_{n=1}^{N} \in \mathbb{R}^{N \times Q}$ such that $Q < D$, that influence the observed plant parameters and represent the underlying structure in the observed parameter space. The sparse formulation

of the GPLVM represents the forward mapping ($\mathcal{X} \rightarrow \mathbf{P}$) as a Gaussian Processes defined independently across D dimensions. Since the latent variables represent the correlations between the parameters, the independence assumption holds when learning the forward mapping. The GP describing the dataset would be:

$$\mathbb{P}(\mathbf{P}|X) = \prod_{n=1}^{N} \prod_{i=1}^{I_n} \mathbb{P}(p_{n,i}|x_n)$$

where p_i represents the i^{th} column of the n^{th} sample. Therefore, if we learn to model the relationship between the latent variables and the plant parameters as GP's, we can assume that human feedback has added noise η. We can represent this as:

$$p_{n,i} = f_i(x_n) + \eta \tag{3.16}$$

where i represents the parameter we are trying to predict and x_n represents the latent variables for the n^{th} sample

3.3 Sparse GP Formulation

We utilise a variational inference approach to train the GPLVM. In order to design an analytically tractable framework, we apply the sparse GP formulation Titsias (2009). We introduce a set of M inducing variables per I_n^{th} dimension $u_i \in \mathbb{R}^M$ computed on inducing locations given by $Z \in \mathbb{R}^{M \times Q}$. The inducing locations live in the input space, where M represents a subset of the original input data X to reduce the computational complexity of the GPLVM. We express the feedback data using the Bayesian GPLVM as presented in Titsias and Lawrence (2010):

$$\begin{aligned} \mathbb{P}(X) &= \prod_{n=1}^{N} \mathcal{N}(x_n|0, \mathbb{I}_Q) \\ \mathbb{P}(f_i|u_i, X, Z, \theta) &= \mathcal{N}(f_i|\alpha_i u_i, Q_{nn}) \\ \mathbb{P}(\mathbf{P}|F, X) &= \prod_{n=1}^{N} \prod_{i=1} \mathcal{N}(p_{n,i}|f_i(x_n), \eta) \end{aligned} \tag{3.17}$$

where $\alpha_i = K_{nm}K_{mm}^{-1}u_i$, $Q_{nn} = K_{nn} - K_{nm}K_{mm}^{-1}K_{mn}$, K_{nn} is the covariance matrix generated by evaluating a kernel function $k_\theta(x_n, x_m)$ on latent points $\{x_n\}_{n=1}^N$. The kernel hyperparameters θ are common across all $\{I_n\}_{n=1}^N$ dimensions. The inducing variables for each dimension u_i following the GP prior distribution can be formulated as: $p(u_i|Z, \theta) = \mathcal{N}(0, K_{mm})$

3.4 Doubly Stochastic Variational Inference

In order to compute the distribution of the latent variables given the observations, we use stochastic variational inference (SVI) Hoffman *et al.* (2013) with a distributed loss function to handle partially specified preferences. SVI approximates the posterior distribution of the model's latent variables with a tractable variational distribution, by maximizing the evidence lower bound (ELBO) on the intractable log marginal likelihood of $p(x)$. Hensman *et al.* (2013) and Lalchand *et al.* (2022) showed how SVI can be applied to GP's, by utilizing a sparse GP formulation and introducing a set of global inducing variables, which can be treated as global latent variables. Doubly stochastic variational inference involves computing stochastic gradients of the ELBO by sampling from a noisy variational distribution of the latent variables and a mini-batch of data to obtain a "doubly stochastic" and unbiased estimate of the gradient. Following this, applying SVI to our formulation, we introduce variational distributions over the latent variables X and inducing variables U_I, and compute the gradient of the ELBO by sampling from the distribution $q(X)$ We write the expression for the doubly stochastic ELBO, deriving from basic definition of ELBO as shown in Lalchand *et al.* (2022),

$$q(X) = \prod_{i=1}^{N} \mathcal{N}(x_n|\mu_n, \boldsymbol{s}_n)$$

$$q(U_I) = \prod_{i=1}^{I_n} \mathcal{N}(u_i|\boldsymbol{m}_i, \boldsymbol{s}_i)$$

$$ELBO = E_q[\sum_{n,i} log\mathcal{N}(p_{n,i}; f_i(x_n), \eta)] - \sum_i KL(q(u_i)||p(u_i|Z, \theta)) \tag{3.18}$$

The expected log likelihood for each data point and dimension can be computed by Monte Carlo Estimation. For each data point, we generate j samples from $q(\mathbf{x}_n)$ using the reparametrization trick as introduced in Kingma and Welling (2022). We sample from the posterior x_n using $x_n^{(j)} = \mu_n + \boldsymbol{s}_n \odot \epsilon^{(j)}$ where $\epsilon^{(j)} \sim \mathcal{N}(0, \boldsymbol{I})$. Following the formulation of the loss function, we can observe that the loss is distributed along dimensions as well as samples, enabling us to handle partially specified preferences. For each sample P_n, the corresponding latent variables x_n are determined solely based on the observed PDPs for which the user provides feedback on. On the other hand, the M inducing variables per dimension u_i are influenced by all the feedback samples in the dataset that have the observed PDP. By incorporating collective information from multiple feedback samples, the inducing variables provide a representation of the PDPs, capturing the global patterns and dependencies that may exist between the PDPs. Therefore, this combined learning approach enhances the model's ability to uncover and model the intricate connections present among the PDPs as well as the mapping connecting the latent variables to the observed PDPs in the feedback.

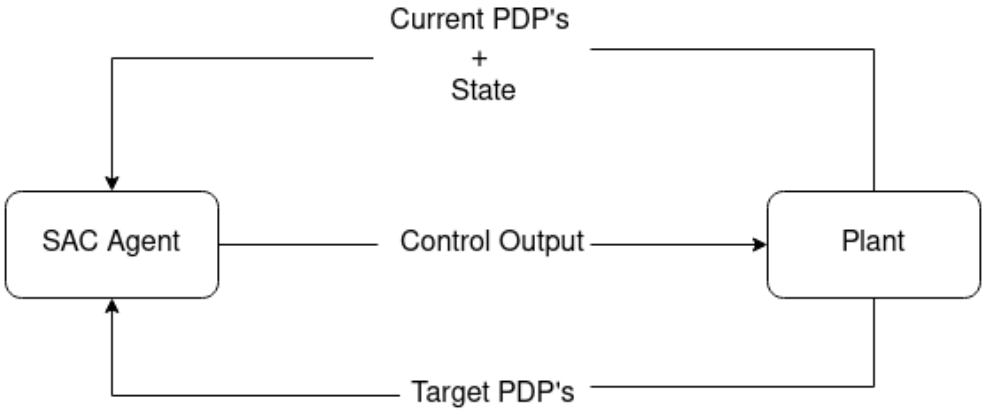


Figure 3.3: Network Architecture Implemented in AUC for Controller Adaptation

Since AUC constantly updates the reference model, we need to ensure stability of

the control system. In adaptive control literature, this is commonly implemented by updating the control parameters ($\mathbf{K}$) to ensure stability of the dynamic system, using the adaptation law generated from the Lyapunov stability theorem Åström (1983); Sastry (1999). After computing the error between the states of the controlled system and the states of the reference model, it is used to adapt the control parameters in real time. This can be implemented with AUC as shown in 3.14 and 3.15. In the current framework of AUC, we approximate the updates to the control parameters by training an SAC agent to predict the stable control output. The network inputs the system state, and extracts the state features to approximate the stability conditions. As shown in Fig. 3.3, we augment the input with the current PDPs to allow the system to estimate the error.

Algorithm 1 Sparse GPLVM with Doubly-stochastic Variational Inference

Inputs:

$\mathbf{P} = \{P_n\}_{n=1}^N$, gradient- based optimizer **optim**()

Initialize:

- Model Parameters: θ (covariance parameters for describing GP), η (function approximation noise)
- Variational Parameters:
  - Inducing locations : Z
  - Local Variational Parameters: $\phi = \{\mu_n, \boldsymbol{s}_n\}_{n=1}^N$
  - Global Variational Parameters: $\lambda = \{\boldsymbol{m}_i, \boldsymbol{s}_i\}_{i=1}^D$

while error converges **do**

- Sample random minibatch from user preferences data $\mathbf{P}_\# \subseteq \mathbf{P}$
- Sample J samples from noisy latent variable distribution $x_n^{(j)} = \mu_n + \boldsymbol{s}_n \odot \epsilon^{(j)}$, $\epsilon^{(j)} \sim \mathcal{N}(0, \boldsymbol{I})$
- Estimate ELBO Loss for $\mathbf{P}_\#$: $L_\# = E_q[\sum_\# \sum_i log\mathcal{N}(p_{\#,i}; f_i(x_\#), \eta)] - \sum_i KL(q(u_i)||p(u_i|Z, \theta))$
- Gradient Step: update $\theta, \phi, \lambda, Z, \eta \leftarrow$ **optim**()

end while

return $\theta^*, \phi^*, \lambda^*, Z^*, \eta^*$

Chapter 4

EXPERIMENTAL SETUP

In this section, we present a set of experiments to evaluate AUC. Our problem setting of dealing with unspecified or hard-to-define errors in control tasks is most similar to preference-based learning. Therefore, we compare our method to the following state-of-the-art PbRL baselines - Meta-Reward-Net Liu *et al.* (2022), SURF Park *et al.* (2022) and PEBBLE **?**.

- Meta-Reward-Net: This algorithm employs bi-level optimization techniques during the reward learning phase.
- SURF: This method proposes a combination of semi-supervised learning and data augmentation.
- PEBBLE: This method uses unsupervised pre-training and off-policy learning.

The participants are shown video clips of the robotic agent in the environment (in simulation) and asked to intervene and modify the controllable parameters (PDPs) at any moment they deemed necessary. This way of soliciting feedback contrasts with that in the traditional PbRL methods. However, each partial preference provided by the user for a subset of the PDPs can be viewed as specifying a complete preference for some virtual trajectory over the current one. We illustrate the proposed approach with the example of a Drone Swarm domain, which is implemented as a potential-fields. The plant model can be defined by the following plant dynamic parameters (PDP's) - Attractive Centroid Threshold, Repulsive Centroid Threshold, Repulsive Obstacle Threshold and number of neighbours in a local cluster. A behavior-based model is

defined relating the plant parameters and the control parameters. We also consider continuous tasks from DMControl Suite: Walker-run, Cheetah-run, and Quadruped-walk, with the plant dynamic parameter (PDP) of velocity of agent in environment. We query feedback from the user and ask them to judge the trajectories in terms of these PDP's.

All the methods are judged on the number of feedback samples required for making the system reach the target plant behavior from the initial plant behavior, where the behaviors are described by different plant dynamics parameters (PDPs). However, evaluating PbRL methods using real human feedback is challenging due to the high sample complexity and high cost of collecting feedback. B-ref Lee *et al.* (2021b) designed a very commonly used benchmark Liu *et al.* (2022); Park *et al.* (2022); Lee *et al.* (2021a) for PbRL by designing scripted/simulated human teachers that provide preferences with respect to a ground truth reward function, with added irrationalities Chan *et al.* (2021); Chipman (2017). To take advantage of these scripted teachers, we evaluate in two steps. In the first step, the user preferences with participants are fed to AUC, which learns the trajectories of the PDPs in the environment, which are treated as the target plant behavior. These trajectories are used to "bias" the scripted teacher to provide preferences closer to the preferences solicited from the participants, by modifying the ground truth return in the following way:

$$R_{user} = R_{task} - \sum_{i=1}^{I_n} \|p_{n,i} - p_{current}\|^2$$

In the second step, the baselines are then trained with the biased script teacher. We count the average number of pairwise preferences, which is required to train them to successfully produce the target plant behavior.

- Drone Swarm Domain: For this domain, the initial plant behavior for all the

methods is generated using a behavior-based control scheme. The "user reward" is obtained from the biased script teacher as shown above. The baselines optimize this "user reward" to predict the PDPs throughout the environment. The PDPs predicted by all the methods are then fed to the behavior schema to produce the low-level control output for the drones.

- DeepMind Control Suite: For the three domain which are part of this suite, the initial plant behavior for the baselines is generated by optimizing an initial reward function, which is learnt from feedback provided by an unbiased scripted teacher given the reward function, to predict the low-level control output. The target PDPs, which are predicted by AUC are used to train a biased script teacher to train the baselines to predict the target plant behavior. For AUC, SAC as presented in Fig. 3.3 is trained for continuous control Srinivas *et al.* (2020) to generate the initial plant behavior, where the input to the network considers the PDPs. This trained SAC acts as the behavior schema for this domain. The PDP trajectories generated by AUC is fed to the pre-trained SAC to produce low-level control for the target plant behavior.

Chapter 5

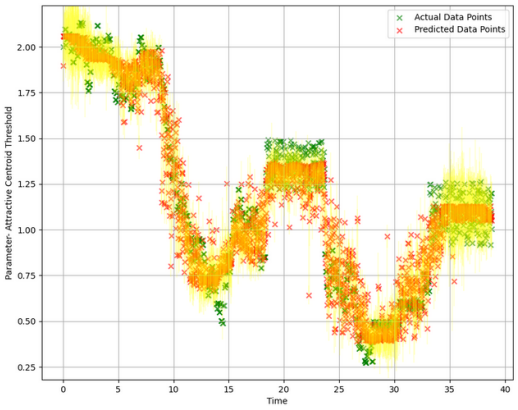
RESULTS

We compare the performance of the proposed method against the baseline PbRL algorithms in terms of sample efficiency. We apply AUC to generate the target PDP trajectories for each human user. Then, we train the baselines with the biased teacher. For AUC, we count the average number of feedback samples i.e. human interventions and modifications required to fit the GP's and predict the target PDP trajectories. For the baselines, we count the average number of pairwise preferences, which is required to train them to successfully produce the target plant behavior. This is determined by comparing how close the PDPs in the control generated by the different methods are to the target parameter values at each time-step.

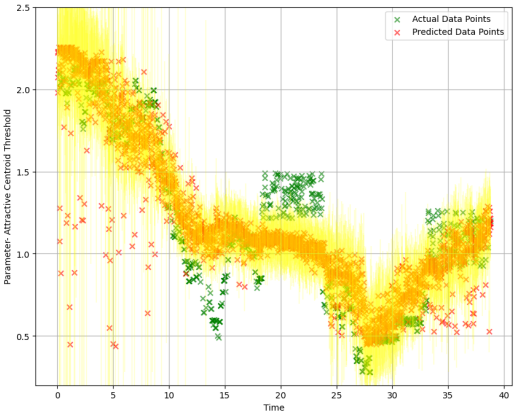
Table 5.1: Average Feedback Required for Achieving Target Parameter Trajectory

Method	Drone Swarm	Walker-walk	Cheetah-run	Quadruped-walk
Proposed Method	180.9	59.2	160.3	302.3
MRN	1003.3	182.1	242.5	692.2
SURF	1209.2	559.29	501.4	992.1
PEBBLE	1210.3	602.4	890.3	1030.10

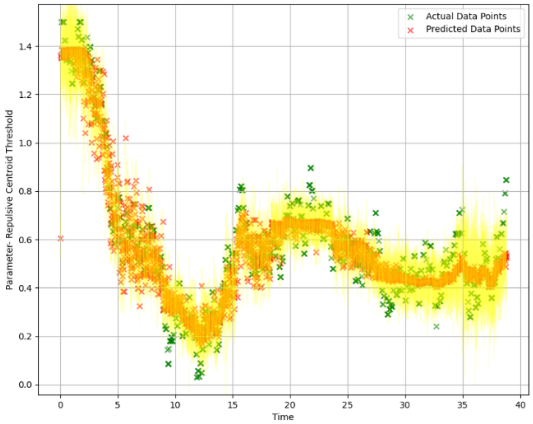
As we can see in Fig. **??**, the proposed methodology is able to successfully learn a hidden variable distribution and learn a Gaussian Process relating the hidden variables to the target parameter values, which are derived from the human feedback. Moreover, as seen in Table 5.1, our proposed method is able to achieve similar performance with much lesser human feedback.


Actual Data Points
Predicted Data Points
Parameter- Attractive Centroid Threshold
Time

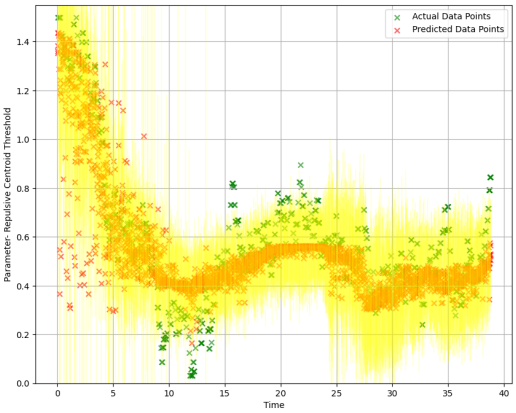
(a)


Actual Data Points
Predicted Data Points
Parameter- Attractive Centroid Threshold
Time

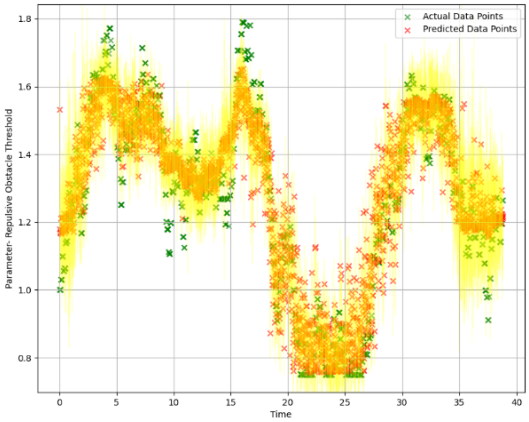
(b)


Actual Data Points
Predicted Data Points
Parameter- Repulsive Centroid Threshold
Time

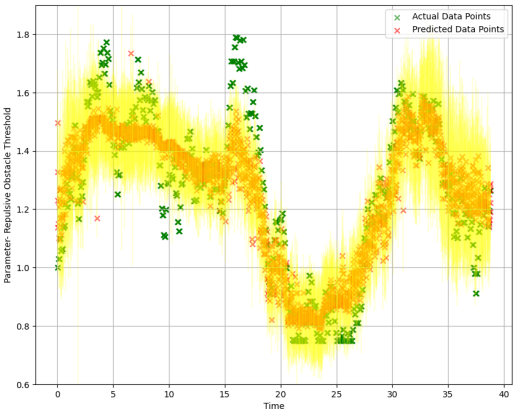
(c)


Actual Data Points
Predicted Data Points
Parameter- Repulsive Centroid Threshold
Time

(d)


Actual Data Points
Predicted Data Points
Parameter- Repulsive Obstacle Threshold
Time

(e)


Actual Data Points
Predicted Data Points
Parameter- Repulsive Obstacle Threshold
Time

(f)

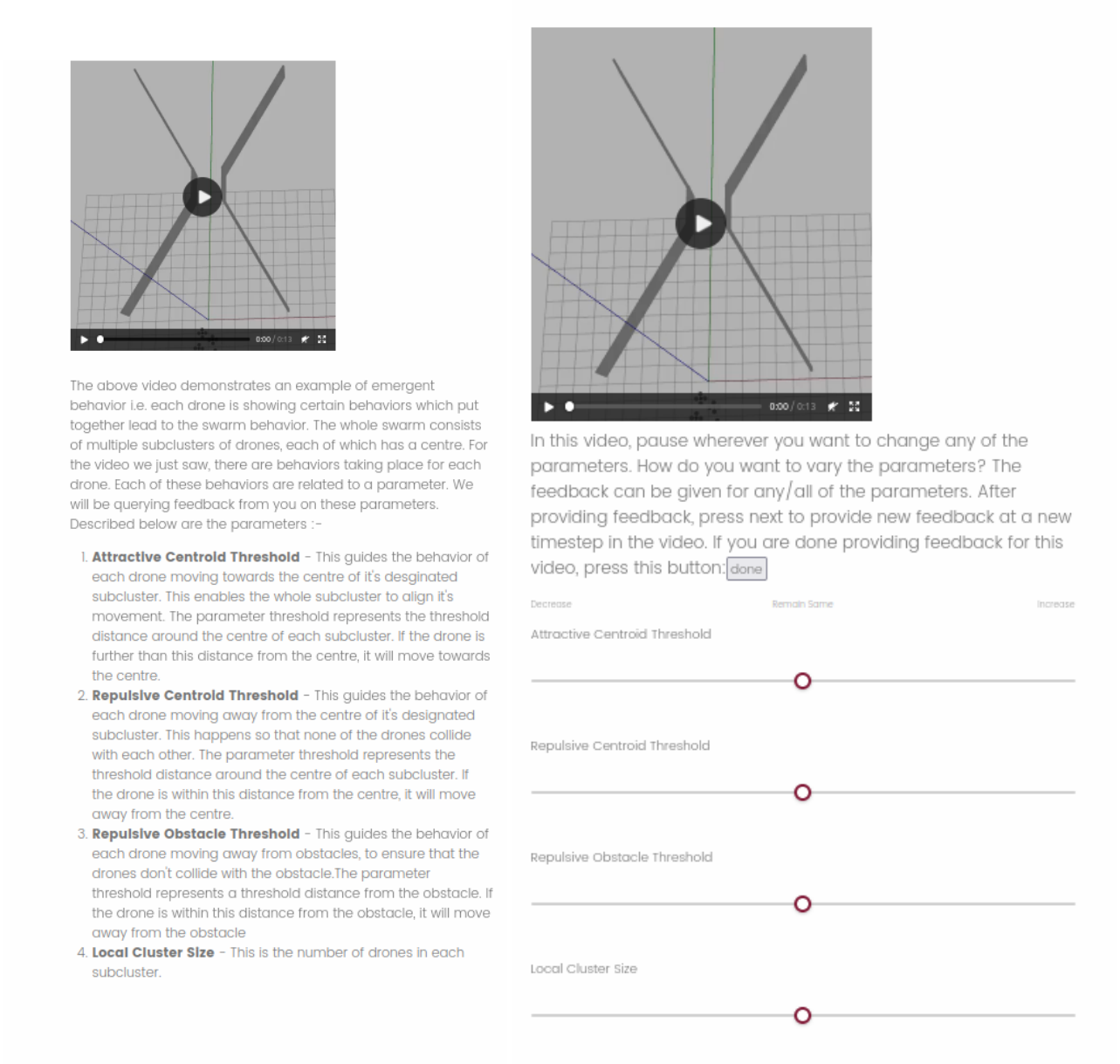
Figure 5.2: Applying AUC to Predict an Instance of Target PDP Trajectories for DroneSwarm Environment where (a), (d) Show Attractive Centroid Threshold, (b), (e) Show Repulsive Centroid Threshold, and (c), (f) Show Repulsive Obstacle Threshold. Second Row shows AUC Prediction with Safety Constraints.

5.1 Control with Safety Constraints

We test out the performance of the proposed approach with safety constraints in place on the drone swarm environment. To generate safe plant dynamics parameters for each state, we train the SAC network similar to Fig. 3.3. First, we identify safety considerations that the drones must adhere to during their operation to avoid critical system failures - minimum distance from each other and the obstacles, which are then incorporated into the reward function used by SAC. Therefore, the SAC approximates the system model from the MDP state using neural networks, and then learns to predict safe plant parameters for each state in the environment i.e. safe plant parameter trajectories. The ELBO loss in Eq. (3.18) is augmented with an additional term to penalize large deviations from the safe values.

$$E_q[\log p(\mathbf{P}|F,X)] = E_q[\sum_{n,i} log\mathcal{N}(p_{n,i}; f_i(x_n), \eta)] - \sum_n KL(q(X)||p(X) - \sum_i KL(q(U_I)||p(U_I) - \lambda g(p_{n,i}, p_{safe_{n,i}})$$

where $g(p_{n,i}, p_{safe_{n,i}}) = max(\epsilon, |p_{n,i} - p_{safe_{n,i}}|)$, and ϵ is the safety threshold parameter. As we can observe in Fig2., training the proposed approach with $\epsilon = 0.5$ leads to smoother PDP trajectories.



(a) Introduction to Drone Swarm environment (b) Query Process

Figure 5.3: Human Study for Drone Swarm Environment

5.2 Human Study Design

We set up a human study using Qualitrics. The study employed a participant-centered design aimed at collecting human feedback data. Each participant was

presented with a selection of the Drone swarm task and DeepMind Control tasks, and they were given the freedom to choose as many tasks as they desired to participate in. Prior to providing feedback, participants were provided with a detailed explanation of the environment. This included an overview of the agent's goals within that environment and an understanding of the PDPs involved. To illustrate the potential impact of these parameters on the agent's movement, participants were shown a video featuring the agent operating within the environment along with the trajectories of these parameters throughout. Then, the participants were shown video clips of the agent in the environment and were asked to intervene and modify the controllable parameters at any moment they deemed necessary. We provide an example of the human study conducted for the drone swarm environment below:

5.2.1 DeepMind PDP Trajectories

We present the PDP trajectories generated using user feedback for the DeepMind control tasks. The user is asked to provide feedback in terms of desired velocity of the walker, cheetah and quadruped in the environment. From this feedback data, AUC successfully learns a Gaussian Process to predict the desired trajectory of the PDPs over time.

5.3 Ablation Study- Providing Random Trajectory Segments vs Whole Trajectory

We modify critical design choices used in the proposed method and evaluate the effect on system performance.

Primarily, we explore the effect of using random trajectory segments as done in traditional PbRL works. We query the user for feedback on randomly selected trajectories from the intital plant behavior for each domain, and use AUC to generate the target behavior. As we can see from the Table 2, we see that using random

trajectories greatly increases the sample complexity of the approach.

Table 5.2: Average Feedback Required for Achieving Target Parameter Trajectory when Provided Random Trajectories vs AUC Method

Method	Drone Swarm	Walker-walk	Cheetah-run	Quadruped-walk
Proposed Method	180.9	59.2	160.3	302.3
Ablation Study	403.2	129.2	281.1	509.3

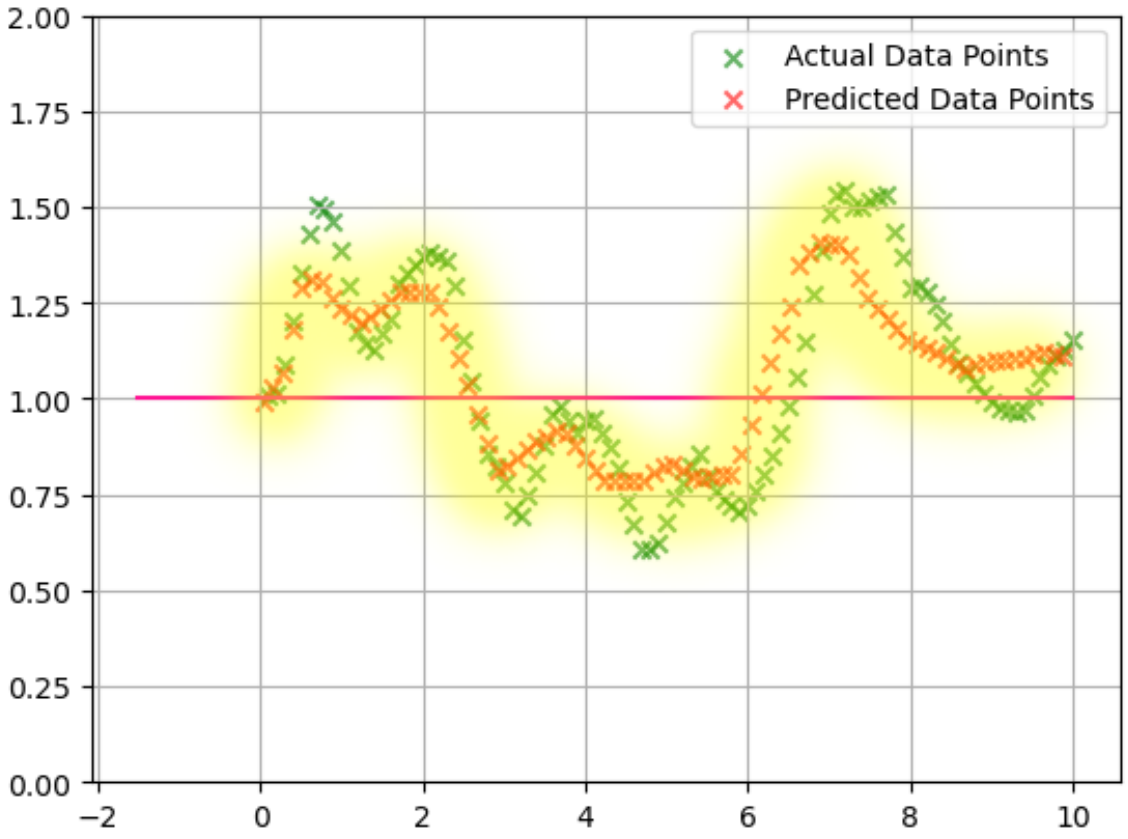


Figure 5.4: PDP Trajectories for DeepMind Walker-Walk Domain

Chapter 6

CONCLUSION

In this paper, we propose a novel method to achieve adaptive control from partially specified attribute preferences. It bridges preference-based learning and adaptive control to improve the adaptability of robotic systems that cohabit with human users. Our proposed methodology is able to learn any given target behavior using a lesser number of human feedback as compared to existing PbRL baselines.

This work is very relevant to the problem of value alignment Soares and Fallenstein (2015); Russell *et al.* (2016). Value alignment is the process of ensuring that the values and goals of an AI or robotic system are aligned with the values and goals of its human operators, users, and other stakeholders . It is an especially important issue in the current robotics domain because robots are becoming increasingly autonomous and capable of making decisions on their own Russell (2019); Bostrom (2014). Therefore, it is necessary that such they act in ways that are consistent with human preferences to promote trust and cooperation in shared workspace. However, the important question is, how can humans express their desired objectives to robotic systems in order to bridge the gap between the human's desired model of the robot and the it's actual model, where the model can be deemed representative of how the robot behaves and responds to it's environment. In recent years, works have tried to solve this problem using demonstrations or preferences, Ng and Russell (2000); Arora and Doshi (2021); Schaal (1996,?); Christiano *et al.* (2023); Warnell *et al.* (2018); Zhang *et al.* (2019); Lee *et al.* (2021a). However, major issues faced with applying them to real-world applications is that demonstrations or preferences are expensive to collect, and are not feasible to be used by non-expert end users. Demonstrations limited applicability for

tasks beyond the expertise of end users, while methods relying on trajectory rankings as preferences provide an impoverished way of expressing preferences leading noisy and conflicting feedback. Considering the non-expertise and limited system knowledge of the end user, our objective is to develop a methodology that effectively caters to their requirements and capabilities. As discussed before, the plant model encapsulates the dynamics, characteristics, and responses of the robot within its environment. Through repeated interactions, non-expert end users of robotic systems can develop a notable level of familiarity and intuitive understanding with the plant dynamics. This enables them to provide valuable feedback about the plant and communicate their objectives to the robot, however not enough to directly define the plant parameters corresponding to the desired behavior. Instead, we focus on designing a framework for users to effectively express their feedback by modifying these parameter, thereby enabling them to actively participate in shaping and adapting the plant to their needs and preferences, enhancing their overall interaction and effectiveness with the robot. The proposed system acknowledges that users may face difficulties in providing feedback on all aspects of the system's behavior and correctly guide the system, due to limited knowledge and understanding of the system's functioning and complexities. To tackle these challenges, the proposed approach introduces partially specified preferences over plant parameters, which is an novel framework for users to express their preferences and desired objectives to the robot. The question which we then ask is; Given these partially-specified preferences over the plant parameters, how do we efficiently update the controller and plant to achieve the desired robot behavior?

The uniqueness of our preference collection method renders past PbRL approaches inadequate for addressing the problem at hand. These existing methods typically rely on trajectory-based preferences that pertain to low-level parameters, and use these preferences to learn the optimal control of the robot. In doing so, they also

subsequently learn the optimal high-level parameters. Our proposed approach deviates from this and queries the human to provide on these high-level parameters itself. We establish that for most existing task domains, we can utilize already existing works to implement the control of robots in these domains. As a result, our focus lies not in learning the control of these robots from scratch, but rather in augmenting them with the capability to modify and adapt based on user preferences. While PbRL aims to learn optimal control, we aim to learn flexible control. Additionally, approaches that utilize high-level attribute preferences! depend on complete specifications of these preferences. Furthermore, these methods focus on learning the desired global behavior of the robot, whereas our proposal aims to modify the local behavior of the robot within the given environment.

To develop a versatile control framework that can effectively handle partially-specified parameter preferences, we employ the Model Reference Adaptive Control (MRAC) framework. Specifically, we demonstrate its applicability in the drone-swarm domain by formulating the agent as a controller and plant. Through user interactions, we solicit their preferences, which are then utilized to derive the reference plant model. Since we allow the user to modify the local behavior of a robot in it's environment, the reference model is time-varying. Recognizing that complete error specification is essential for adapting the controller parameters, we account for the incompleteness in the desired plant parameter preferences. In our approach, we assume that the plant parameters exhibit underlying correlations, and to capture this structural dependency, we employ a latent variable distribution learning method. This enables us to effectively represent and model the relationships within the parameters, and estimate a completely-defined reference model. Subsequently, we derive the controller adaptation laws utilizing the above framework and show how the proposed approach can be successfully implemented with standard Model Reference Adaptive Control

(MRAC) approaches, resulting in the attainment of user-adaptive control, wherein the plant aka the robot effectively adapts to user preferences.

Future Work

Owing to the novel nature of the proposed approach, there are several areas that require further exploration and development for the proposed approach to ensure the viability and effectiveness of the approach in real-world scenarios. Given that the reference plant in our approach exhibits time-varying characteristics, the standard MRAC framework needs to be revised with an adaptation mechanism that facilitates simultaneous controller adaptation while ensuring stability and expedites error convergence. This crucial for achieving real-time tracking of the reference plant by implementing timely response to changes in the reference plant, thereby enabling real-time modifications to the robot's behavior in accordance with user preferences. Furthermore, in the formulation of the proposed approach, we make the assumption that the plant parameters are known and do not explicitly consider the inherent uncertainty in the plant as it adapts to the reference model. However, it is crucial to acknowledge the presence of noise and uncertainty in the plant and design the adaptation mechanism accordingly. By incorporating measures to account for these factors, the proposed approach can effectively accommodate the variations and uncertainties that may arise during the adaptation process, resulting in improved performance and stability. While enabling users to modify the plant parameters can offer significant advantages across various application domains while facilitating the desired flexibility and adaptability of robotic systems, unrestricted modifications by non-expert users can potentially introduce safety risks. Hence, it becomes paramount to carefully navigate the trade-off between accommodating user preferences and maintaining system safety. In the context of implementing an adaptive control framework, by integrating

safety measures within the adaptive control framework, the overall system can strike a delicate balance between user preferences and safety considerations, safeguarding both users and the system from undesirable outcomes. Utilizing a safety model or a separate safety analysis module, the behavior of the system can be continuously compared against safety specifications. This verification process helps identify any deviations or violations and triggers corrective actions to maintain safety. To further improve generalizability of the latent variable model, the proposed approach can also incorporate environmental features as inputs during the prediction of desired parameters across diverse environmental settings. The desired parameter values can be augmented with contextual information derived from the environment. This augmentation enables the predictions to generalize across different environment configurations, thereby reducing the dependence on extensive user feedback. Consequently, the proposed approach will minimizes the feedback requirements by leveraging the inherent relationships between the environmental features and desired parameter values, leading to improved generalization and adaptability across varying environmental contexts. In the proposed approach, the novel concept of directly incorporating user preferences on plant parameters can be further enhanced by learning a mapping between these parameters and high-level behaviors. The user can then provide feedback on the observed behavior of the robot, enabling the derivation of desired plant parameters. Subsequently, the reference plant model can be constructed based on these desired parameters. This mapping between user-expressed behavioral preferences and plant parameters will facilitate a more intuitive and user-friendly interaction, allowing users to provide feedback on the desired behavior rather than explicitly modifying plant parameters.

REFERENCES

Abbeel, P. and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning", in "Proceedings of the Twenty-First International Conference on Machine Learning", ICML '04, p. 1 (Association for Computing Machinery, New York, NY, USA, 2004), URL https://doi.org/10.1145/1015330.1015430.

Ajzen, I., T. C. Brown and F. Carvajal, "Explaining the discrepancy between intentions and actions: The case of hypothetical bias in contingent valuation", Personality and Social Psychology Bulletin **30**, 9, 1108–1121, URL https://doi.org/10.1177/0146167204264079, pMID: 15359015 (2004).

Alqaudi, B., H. Modares, I. Ranatunga, S. Tousif, F. Lewis and D. Popa, "Model reference adaptive impedance control for physical human-robot interaction", Control Theory and Technology **14**, 68–82 (2016).

Argall, B. D. and A. G. Billard, "A survey of tactile human–robot interactions", Robotics and Autonomous Systems **58**, 10, 1159–1176, URL https://www.sciencedirect.com/science/article/pii/S0921889010001375 (2010).

Arora, S. and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress", Artificial Intelligence **297**, 103500, URL https://www.sciencedirect.com/science/article/pii/S0004370221000515 (2021).

Arteaga, M. and Y. Tang, "Adaptive control of robots with an improved transient performance", IEEE Transactions on Automatic Control **47**, 7, 1198–1202 (2002).

Bae, J. and M. Tomizuka, "A gait rehabilitation strategy inspired by an iterative learning algorithm", Mechatronics **22**, 2, 213–221, URL https://www.sciencedirect.com/science/article/pii/S0957415812000116 (2012).

Bayoumi, M. and L. Mo, "Adaptive decoupling control of mimo system", IFAC Proceedings Volumes **21**, 9, 109–113, URL https://www.sciencedirect.com/science/article/pii/S1474667017547114, 8th IFAC/IFOORS Symposium on Identification and System Parameter Estimation 1988, Beijing, PRC, 27-31 August (1988).

Bostrom, N., *Superintelligence: Paths, Dangers, Strategies* (Oxford University Press, Inc., USA, 2014), 1st edn.

Buerger, S. P. and N. Hogan, "Complementary stability and loop shaping for improved human–robot interaction", IEEE Transactions on Robotics **23**, 2, 232–244 (2007).

Cailloux, O. and S. Destercke, "Reasons and means to model preferences as incomplete", CoRR **abs/1801.01657**, URL http://arxiv.org/abs/1801.01657 (2018).

Calanca, A., L. Capisani and P. Fiorini, "Robust force control of series elastic actuators", Actuators **3**, 3, 182–204, URL https://www.mdpi.com/2076-0825/3/3/182 (2014).

Carlucho, I., M. De Paula, S. Villar and G. Acosta, "Incremental q-learning strategy for adaptive pid control of mobile robots", Expert Systems with Applications **80** (2017).

Chakraborti, T., S. Kambhampati, M. Scheutz and Y. Zhang, "Ai challenges in human-robot cognitive teaming", arXiv preprint arXiv:1707.04775 (2017).

Chan, L., A. Critch and A. Dragan, "The impacts of known and unknown demonstrator irrationality on reward inference", URL `https://openreview.net/forum?id=CzRSsOG6JDw` (2021).

Chen, S., K. Boggess, D. Parker and L. Feng, "Multi-objective controller synthesis with uncertain human preferences", (2022).

Chipman, S. E. F., *The Oxford Handbook of Cognitive Science* (Oxford University Press, 2017), URL `https://doi.org/10.1093/oxfordhb/9780199842193.001.0001`.

Choudhury, R., G. Swamy, D. Hadfield-Menell and A. D. Dragan, "On the utility of model learning in hri", in "2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)", pp. 317–325 (IEEE, 2019).

Christiano, P., J. Leike, T. B. Brown, M. Martic, S. Legg and D. Amodei, "Deep reinforcement learning from human preferences", (2023).

Datta, A. and P. Ioannou, "Performance analysis and improvement in model reference adaptive control", IEEE Transactions on Automatic Control **39**, 12, 2370–2387 (1994).

de Graaf, M., S. Allouch and J. A. Van Dijk, "Long-term acceptance of social robots in domestic environments: Insights from a user's perspective", (2016).

de Graaf, M. M. and S. Ben Allouch, "Exploring influencing variables for the acceptance of social robots", Robotics and Autonomous Systems **61**, 12, 1476–1486, URL `https://www.sciencedirect.com/science/article/pii/S0921889013001334` (2013).

De Santis, A., B. Siciliano, A. De Luca and A. Bicchi, "An atlas of physical human–robot interaction", Mechanism and Machine Theory **43**, 3, 253–270, URL `https://www.sciencedirect.com/science/article/pii/S0094114X07000547` (2008).

Dragan, A. D., K. C. Lee and S. S. Srinivasa, "Legibility and predictability of robot motion", in "2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)", pp. 301–308 (IEEE, 2013).

Dubra, J., F. Maccheroni and E. A. Ok, "Expected utility theory without the completeness axiom", Journal of Economic Theory **115**, 1, 118–133, URL `https://www.sciencedirect.com/science/article/pii/S0022053103001662` (2004).

Duchaine, V. and C. M. Gosselin, "General model of human-robot cooperation using a novel velocity based variable impedance control", in "Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)", pp. 446–451 (2007).

Gibson, T., "Closed-loop reference model adaptive control : with application to very flexible aircraft", (2014).

Gong, Z. and Y. Zhang, "What is it you really want of me? generalized reward learning with biased beliefs about domain dynamics", in "Proceedings of the AAAI Conference on Artificial Intelligence", vol. 34, pp. 2485–2492 (2020).

Gong, Z. and Y. Zhang, "Explicable policy search", Advances in Neural Information Processing Systems **35**, 38859–38872 (2022).

Gonzalez, J. and G. Widmann, "A force commanded impedance control scheme for robots with hard nonlinearities", IEEE Transactions on Control Systems Technology **3**, 4, 398–408 (1995).

Haddadin, S. and E. Croft, *Physical Human–Robot Interaction*, pp. 1835–1874 (Springer International Publishing, Cham, 2016), URL `https://doi.org/10.1007/978-3-319-32552-1_69`.

Hanni, A., A. Boateng and Y. Zhang, "Safe explicable robot planning", arXiv preprint arXiv:2304.03773 (2023).

Hanni, A. and Y. Zhang, "Generating active explicable plans in human-robot teaming", in "2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)", pp. 2993–2998 (IEEE, 2021).

Heerink, M., B. Krose, V. Evers and B. Wielinga, "Assessing acceptance of assistive social agent technology by older adults: the almere model", I. J. Social Robotics **2**, 361–375 (2010).

Hensman, J., N. Fusi and N. D. Lawrence, "Gaussian processes for big data", (2013).

Hoffman, M., D. M. Blei, C. Wang and J. Paisley, "Stochastic variational inference", (2013).

Hogan, N., "Adaptive control of mechanical impedance by coactivation of antagonist muscles", IEEE Transactions on Automatic Control **29**, 8, 681–690 (1984).

Hogan, N., "Impedance Control: An Approach to Manipulation: Part III—Applications", Journal of Dynamic Systems, Measurement, and Control **107**, 1, 17–24, URL `https://doi.org/10.1115/1.3140701` (1985).

Kazerooni, H., T. Sheridan and P. Houpt, "Robust compliant motion for manipulators, part i: The fundamental concepts of compliant motion", IEEE Journal on Robotics and Automation **2**, 2, 83–92 (1986).

Khan, S. G., G. Herrmann, F. L. Lewis, T. Pipe and C. Melhuish, "Reinforcement learning and optimal adaptive control: An overview and implementation examples", Annual Reviews in Control **36**, 1, 42–59, URL `https://www.sciencedirect.com/science/article/pii/S1367578812000053` (2012).

Kingma, D. P. and M. Welling, "Auto-encoding variational bayes", (2022).

Kleinman, D., S. Baron and W. Levison, "An optimal control model of human response part i: Theory and validation", Automatica **6**, 3, 357–369, URL https://www.sciencedirect.com/science/article/pii/0005109870900518 (1970).

Kober, J., J. A. Bagnell and J. Peters, "Reinforcement learning in robotics: A survey", The International Journal of Robotics Research **32**, 11, 1238–1274, URL https://doi.org/10.1177/0278364913495721 (2013).

Lalchand, V., A. Ravuri and N. D. Lawrence, "Generalised gaussian process latent variable models (gplvm) with stochastic variational inference", (2022).

Lawrence, N. D., "Gaussian process latent variable models for visualisation of high dimensional data", in "Proceedings of the 16th International Conference on Neural Information Processing Systems", NIPS'03, p. 329–336 (MIT Press, Cambridge, MA, USA, 2003).

Lee, C., J. Lee and S. Oh, "Towards accurate force control of series elastic actuators exploiting a robust transmission force observer", (2019).

Lee, K., L. Smith and P. Abbeel, "Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training", (2021a).

Lee, K., L. Smith, A. Dragan and P. Abbeel, "B-pref: Benchmarking preference-based reinforcement learning", in "Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)", (2021b), URL https://openreview.net/forum?id=ps95-mkHF_.

Li, Z., B. Huang, Z. Ye, M. Deng and C. Yang, "Physical human–robot interaction of a robotic exoskeleton by admittance control", IEEE Transactions on Industrial Electronics **65**, 12, 9614–9624 (2018).

Liu, L., S. Tian, D. Xue, T. Zhang, Y. Chen and S. Zhang, "A review of industrial mimo decoupling control", International Journal of Control, Automation and Systems **17** (2019).

Liu, R., F. Bai, Y. Du and Y. Yang, "Meta-reward-net: Implicitly differentiable reward learning for preference-based reinforcement learning", in "Advances in Neural Information Processing Systems", edited by A. H. Oh, A. Agarwal, D. Belgrave and K. Cho (2022), URL https://openreview.net/forum?id=OZKBReUF-wX.

Mitsunaga, N., C. Smith, T. Kanda, H. Ishiguro and N. Hagita, "Adapting robot behavior for human–robot interaction", IEEE Transactions on Robotics **24**, 4, 911–916 (2008).

Mnih, V., A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning", (2016).

Morse, A., "Supervisory control of families of linear set-point controllers - part i. exact matching", IEEE Transactions on Automatic Control **41**, 10, 1413–1431 (1996).

Musić, S. and S. Hirche, "Control sharing in human-robot team interaction", Annual Reviews in Control **44**, 342–354, URL https://www.sciencedirect.com/science/article/pii/S1367578817301153 (2017).

Na, J., J. Yang and G. Gao, "Reinforcing transient response of adaptive control systems using modified command and reference model", IEEE Transactions on Aerospace and Electronic Systems **56**, 3, 2005–2017 (2020).

Newman, W. S., "Stability and Performance Limits of Interaction Controllers", Journal of Dynamic Systems, Measurement, and Control **114**, 4, 563–570, URL https://doi.org/10.1115/1.2897725 (1992).

Ng, A. Y. and S. J. Russell, "Algorithms for inverse reinforcement learning", in "Proceedings of the Seventeenth International Conference on Machine Learning", ICML '00, p. 663–670 (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000).

Nguyen, N., "Time-varying reference model modification of adaptive control with multi-objective performance optimization", (2022).

Okunev, V., T. Nierhoff and S. Hirche, "Human-preference-based control design: Adaptive robot admittance control for physical human-robot interaction", in "2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication", pp. 443–448 (2012a).

Okunev, V., T. Nierhoff and S. Hirche, "Human-preference-based control design: Adaptive robot admittance control for physical human-robot interaction", in "2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication", pp. 443–448 (2012b).

Park, J., Y. Seo, J. Shin, H. Lee, P. Abbeel and K. Lee, "Surf: Semi-supervised reward learning with data augmentation for feedback-efficient preference-based reinforcement learning", (2022).

Petsagkourakis, P., I. O. Sandoval, E. Bradford, D. Zhang and E. A. del Rio Chanona, "Reinforcement learning for batch bioprocess optimization", (2019).

Pini, M., F. Rossi, K. Venable and T. Walsh, "Incompleteness and incomparability in preference aggregation: Complexity results", Artif. Intell. **175**, 1272–1289 (2011).

Ranatunga, I., F. L. Lewis, D. O. Popa and S. M. Tousif, "Adaptive admittance control for human–robot interaction using model reference design and adaptive inverse filtering", IEEE Transactions on Control Systems Technology **25**, 1, 278–285 (2017).

Reddy, S., A. Dragan and S. Levine, "Where do you think you're going?: Inferring beliefs about dynamics from behavior", Advances in Neural Information Processing Systems **31** (2018).

Ross, S. and D. Bagnell, "Efficient reductions for imitation learning", in "Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics", edited by Y. W. Teh and M. Titterington, vol. 9 of *Proceedings of Machine Learning Research*, pp. 661–668 (PMLR, Chia Laguna Resort, Sardinia, Italy, 2010), URL https://proceedings.mlr.press/v9/ross10a.html.

Russell, S., *Human Compatible: Artificial Intelligence and the Problem of Control* (Penguin Publishing Group, 2019), URL https://books.google.com/books?id=M1eFDwAAQBAJ.

Russell, S., D. Dewey and M. Tegmark, "Research priorities for robust and beneficial artificial intelligence", (2016).

Sastry, S., *Lyapunov Stability Theory*, pp. 182–234 (Springer New York, New York, NY, 1999), URL https://doi.org/10.1007/978-1-4757-3108-8_5.

Schaal, S., "Learning from demonstration", in "Advances in Neural Information Processing Systems", edited by M. Mozer, M. Jordan and T. Petsche, vol. 9 (MIT Press, 1996), URL https://proceedings.neurips.cc/paper_files/paper/1996/file/68d13cf26c4b4f4f932e3eff990093ba-Paper.pdf.

Sedighizadeh, M. and A. Rezazadeh, "Adaptive pid controller based on reinforcement learning for wind turbine control", Proceedings of World Academy of Science, Engineering and Technology (CESSE2008) **27** (2008).

Shahrdar, S., L. Menezes and M. Nojoumian, *A Survey on Trust in Autonomous Systems: Proceedings of the 2018 Computing Conference, Volume 2*, pp. 368–386 (2019).

Sheridan, T., "Human-robot interaction: Status and challenges", Human factors **58** (2016).

Singh, S., R. Lewis and A. Barto, "Where do rewards come from?", (2009).

Soares, N. and B. Fallenstein, "Aligning superintelligence with human interests: A technical research agenda", (2015).

Spencer, J., S. Choudhury, M. Barnes, M. Schmittle, M. Chiang, P. Ramadge and S. Srinivasa, "Learning from interventions: Human-robot interaction as both explicit and implicit feedback", in "Robotics", edited by M. Toussaint, A. Bicchi and T. Hermans, Robotics: Science and Systems (MIT Press Journals, United States, 2020).

Spielberg, S., A. Tulsyan, N. P. Lawrence, P. D. Loewen and R. Bhushan Gopaluni, "Toward self-driving processes: A deep reinforcement learning approach to control", AIChE Journal **65**, 10, e16689, URL https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.16689 (2019).

Srinivas, A., M. Laskin and P. Abbeel, "Curl: Contrastive unsupervised representations for reinforcement learning", (2020).

Stepanyan, V. and K. Krishnakumar, "Adaptive control with reference model modification", Journal of Guidance, Control, and Dynamics **35**, 1370–1374 (2012).

Sun, J., "A modified model reference adaptive control scheme for improved transient performance", in "1991 American Control Conference", pp. 150–155 (1991).

Suzuki, S. and K. Furuta, "Adaptive impedance control to enhance human skill on a haptic interface system", Journal of Control Science and Engineering **2012**, 1–10 (2012).

Tao, G., "Multivariable adaptive control: A survey", Automatica **50**, 11, 2737–2764, URL `https://www.sciencedirect.com/science/article/pii/S0005109814003963` (2014).

Tassa, Y., Y. Doron, A. Muldal, T. Erez, Y. Li, D. de Las Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq, T. Lillicrap and M. Riedmiller, "Deepmind control suite", (2018).

Titsias, M., "Variational learning of inducing variables in sparse gaussian processes", in "Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics", edited by D. van Dyk and M. Welling, vol. 5 of *Proceedings of Machine Learning Research*, pp. 567–574 (PMLR, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 2009), URL `https://proceedings.mlr.press/v5/titsias09a.html`.

Titsias, M. and N. D. Lawrence, "Bayesian gaussian process latent variable model", in "Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics", edited by Y. W. Teh and M. Titterington, vol. 9 of *Proceedings of Machine Learning Research*, pp. 844–851 (PMLR, Chia Laguna Resort, Sardinia, Italy, 2010), URL `https://proceedings.mlr.press/v9/titsias10a.html`.

Tsetserukou, D., R. Tadakuma, H. Kajimoto, N. Kawakami and S. Tachi, "Intelligent variable joint impedance control and development of a new whole-sensitive anthropomorphic robot arm", in "2007 International Symposium on Computational Intelligence in Robotics and Automation", pp. 338–343 (2007).

Tsumugiwa, T., R. Yokogawa and K. Hara, "Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task", in "Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)", vol. 1, pp. 644–650 vol.1 (2002).

Wang, H. and K. Kosuge, "Control of a robot dancer for enhancing haptic human-robot interaction in waltz", IEEE Transactions on Haptics **5**, 3, 264–273 (2012).

Wang, Y., K. Velswamy and B. Huang, "A novel approach to feedback control with deep reinforcement learning", IFAC-PapersOnLine **51**, 31–36 (2018).

Warnell, G., N. Waytowich, V. Lawhern and P. Stone, "Deep tamer: Interactive agent shaping in high-dimensional state spaces", in "Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications

of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence", AAAI'18/IAAI'18/EAAI'18 (AAAI Press, 2018).

Wirth, C., R. Akrour, G. Neumann and J. Fürnkranz, "A survey of preference-based reinforcement learning methods", J. Mach. Learn. Res. **18**, 1, 4945–4990 (2017).

Wirth, C. and J. Fürnkranz, "A policy iteration algorithm for learning from preference-based feedback", in "Advances in Intelligent Data Analysis XII - 12th International Symposium, IDA 2013, London, UK, October 17-19, 2013. Proceedings", edited by A. Tucker, F. Höppner, A. Siebes and S. Swift, vol. 8207 of *Lecture Notes in Computer Science*, pp. 427–437 (Springer, 2013), URL `https://doi.org/10.1007/978-3-642-41398-8_37`.

Wolpert, D. M., R. Miall and M. Kawato, "Internal models in the cerebellum", Trends in Cognitive Sciences **2**, 9, 338–347, URL `https://www.sciencedirect.com/science/article/pii/S1364661398012212` (1998).

Zakershahrak, M., A. Sonawane, Z. Gong and Y. Zhang, "Interactive plan explicability in human-robot teaming", in "2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)", pp. 1012–1017 (IEEE, 2018).

Zhang, R., F. Torabi, L. Guan, D. H. Ballard and P. Stone, "Leveraging human guidance for deep reinforcement learning tasks", (2019).

Zhang, Y., S. Sreedharan, A. Kulkarni, T. Chakraborti, H. H. Zhuo and S. Kambhampati, "Plan explicability and predictability for robot task planning", in "2017 IEEE international conference on robotics and automation (ICRA)", pp. 1313–1320 (IEEE, 2017).

Åström, K., "Theory and applications of adaptive control—a survey", Automatica **19**, 5, 471–486, URL `https://www.sciencedirect.com/science/article/pii/000510988390002X` (1983).