Systems & Control Transactions

Original Research Article
Peer Reviewed Conference Proceeding
ESCAPE 36 - European Symposium on Computer Aided Process Engineering
Sheffield, UK. 21-24 June 2026

PSE PRESS

Enhancing Control in Chemical Processes using Reinforcement from Human Feedback

Hilde Gerold[a]*, Dean Brandner[a], and Sergio Lucia[a]

[a] Technische Universität Dortmund, Laboratory of Process Automation Systems, Emil-Figge-Straße 70, Dortmund 44227, Germany
* Corresponding Author: hilde.gerold@tu-dortmund.de.

ABSTRACT

Reinforcement learning (RL) presents a promising alternative to model-based advanced control schemes, such as model predictive control (MPC), whose application can be limited by highly complex system models. However, incorporating constraints in RL remains challenging and formulating a suitable optimization objective is not straightforward. Reinforcement learning from human feedback (RLHF) offers an approach to derive the RL reward function from human expert preferences, enabling the incorporation of process knowledge. In this work, we present the application of RLHF to fine-tune an approximate MPC controller with suboptimal performance. We demonstrate that combining conventional reward formulations with RLHF, along with varying trajectory segment lengths for collecting human feedback, improves the control methodology for a batch bioreactor by enhancing safety and accounting for long-term effects. Furthermore, direct-preference based policy optimization (DPPO) represents a promising alternative for directly fine-tuning learning-based controllers while circumventing explicit reward model design.

Keywords: Model predictive control, reinforcement learning, human feedback

INTRODUCTION

Advanced process control methods, such as model predictive control (MPC), represent a powerful approach as they enable constraint handling and are applicable to nonlinear multivariable processes. However, the high computational demand resulting from repeatedly solving the associated optimization problem makes real-time application challenging for high-dimensional and complex systems. Deep reinforcement Learning (RL) [1] presents a promising data-based approach, as the resulting control policy enables fast evaluation. This policy is learned through interactions with the system environment and guided by a scalar reward. However, formulating a suitable reward encoding the underlying control objective is not always straightforward. Additionally, constraint incorporation remains challenging, posing a problem for safety-critical process industries. Reinforcement learning from human feedback (RLHF) offers a promising method to derive the RL reward from human expert preferences [2] and is commonly applied for fine-tuning large language models [3], as well as in games [4] and robotics [5]. A detailed survey of RLHF is provided by Kaufmann et al. [6].

This work builds on the recent success of RLHF approaches and investigates different fine-tuning strategies for learning-based controllers using human feedback. First, we provide a short introduction to RL, by an overview of methodologies for incorporating human feedback into the fine-tuning process. Subsequently, we illustrate the application of these methods for improving an approximate MPC controller to control a batch bioreactor. We investigate various design parameters of the RLHF approach to improve process security and incorporation of process knowledge. Finally, we summarize our findings and discuss the observed effects.

BACKGROUND ON REINFORCEMENT LEARNING

In reinforcement learning (RL), an agent learns a mapping from states to actions by interacting with an environment and receiving feedback through a scalar reward $r \in \mathbb{R}$ [1]. Although RL employs distinct notation conventions, we adopt standard control terminology throughout this paper. Accordingly, the terms state and

observation are used interchangeably, as are input and action. The goal in RL is to learn a policy $\pi: \mathcal{X} \rightarrow \mathcal{U}$, mapping the states $x \in \mathbb{R}^{n_x}$ to optimal actions $u \in \mathbb{R}^{n_u}$. The optimization objective is defined to not only maximize the immediate rewards but the accumulated rewards for the following transitions. The RL objective therefore is formalized as:

$$\pi^* = arg\max_{\pi} \mathbb{E}_{\pi}[G] = \arg\max_{\pi} \mathbb{E}_{\pi}\left[\sum_{t=0}^{T} \gamma^t r_{k+t+1}\right]. \quad (1)$$

The optimal policy π^* maximizes the expected return G, which is defined as the accumulated reward r over an episode of length T, where the time instance is denoted by k. The discount parameter $\gamma \in (0, 1]$ reduces the influence of future rewards on the cumulative return.

BACKGROUND ON FINE-TUNING VIA HUMAN FEEDBACK

Reinforcement learning from human feedback

Designing a suitable reward function in RL that aligns with the underlying goal of the process can be challenging. To circumvent excessive reward shaping, reinforcement learning from human feedback (RLHF) incorporates human preferences to obtain a reward function estimate $\hat{r}: \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$ [2]. Specifically, human feedback is obtained through comparison of two trajectory segments $\{\sigma^1, \sigma^2\}$, sampled from a set of trajectories $\{\tau^1, \ldots, \tau^i\}$. Each snippet σ consists of a sequence of state-action pairs of length L_s:

$$\sigma = ((x_0, u_0), (x_1, u_1), \ldots, (x_{L_s-1}, u_{L_s-1})) \in (\mathcal{X} \times \mathcal{U})^{L_s}. \quad (2)$$

The human expert compares both trajectories and selects the preferred one. Both segments, together with the indicated label μ, present the triple $(\sigma^1, \sigma^2, \mu) \in \mathcal{D}$, which is stored in a dataset $\mathcal{D}$. In case that the first segment is preferred over the second one, the corresponding judgmental label is defined as with $\mu(1) = \mu(\sigma^1 \succ \sigma^2) = \{1, 0\}$. If the second segment is preferred, it is marked by $\mu(2) = \mu(\sigma^2 \succ \sigma^1) = \{0, 1\}$. When no distinct preference exists, the clips are rated equally with $\mu(\sigma^1 \sim \sigma^2) = \{0.5, 0.5\}$. Following the methodology of Christiano et al. [2], it is assumed that the human preference for one segment is reasoned in a higher sum of underlying reward of the presented state-action pairs. The preference reward estimate $\hat{r}^{\Phi}$, which is typically presented by a neural network (NN) with its parameters Φ, is then trained to maximize the likelihood (MLE) of reward predictions and true estimates, formalized as:

$$\min_{\phi} -\sum_{(\sigma^1,\sigma^2,\mu)\in\mathcal{D}} \mu(1) \log \hat{P}\left[\sigma^1 \succ \sigma^2, \phi\right] + \mu(2) \log \hat{P}\left[\sigma^2 \succ \sigma^1, \phi\right], \quad (3)$$

where

$$\hat{P}[\sigma^1 \succ \sigma^2, \Phi] = \frac{\exp \sum_t \hat{r}^{\Phi}(x_t^1, u_t^1)}{\exp \sum_t \hat{r}^{\Phi}(x_t^1, u_t^1) + \exp \sum_t \hat{r}^{\Phi}(x_t^2, u_t^2)}. \quad (4)$$

The trained preference reward NN can be implemented into any arbitrary RL framework. The described procedure can be iteratively performed until the desired policy performance is achieved.

Direct preference-based policy optimization

Several challenges arise when using RLHF [7, 8]. The reward model derived from human preferences can fail to capture the true underlying reward of the task [9]. Furthermore, the quality of the learned reward $\hat{r}^{\Phi}$ can only be assessed through evaluation of the resulting fine-tuned policy. Therefore, the entire optimization process not only relies on the supervised reward training, but also on the maximization problem during policy learning [7]. As a result, direct preference-based policy optimization (DPPO) [9] has emerged as an alternative to RLHF, directly optimizing the policy parameters using the preference dataset $\mathcal{D}$ without requiring an explicit reward model. In DDPO the applied score metric quantifies the distance between policy predictions and actions of the labeled trajectory segments by any aggregation of distance d. Applying MLE to adjust the policy parameters, the training loss is defined similar to Equation (3):

$$\min_{\theta} -\sum_{(\sigma^1,\sigma^2,\mu)\in\mathcal{D}} \mu(1) \log \tilde{s}\left(\pi^{\theta}, \sigma^1 \succ \sigma^2, \lambda_{\tilde{s}}\right) + \mu(2) \log \tilde{s}\left(\pi^{\theta}, \sigma^2 \succ \sigma^1, \lambda_{\tilde{s}}\right), \quad (5)$$

where

$$\tilde{s}\left(\pi^{\theta}, \sigma^1 \succ \sigma^2\right) = \frac{\exp\left(-d(\pi^{\theta}, \sigma^1)\right)}{\exp\left(-d(\pi^{\theta}, \sigma^1)\right) + \exp\left(-\lambda_{\tilde{s}} d(\pi^{\theta}, \sigma^2)\right)}. \quad (6)$$

The regularization factor $\lambda_{\tilde{s}} \in (0, 1)$ is introduced to penalize the increase of both distances. Without this regularization, the ratio $\tilde{s}$ could stay in a similar range even when the overall distance to both segments increases [9]. A comparison of RLHF with such a direct policy optimization (DPO) [10] method has found that DPO has the potential to outperform RLHF, especially with a sufficient large preference dataset [11]. However, DPO can be sensitive to distribution shifts between the provided preference data and can lead to insufficient generalization [12]. As in RLHF the reward model might provide incorrect high rewards to unseen data, this misspecification can also happen for DPO but directly impacts the policy [12].

IMPROVEMENT OF APPROXIMATE MPC FOR BATCH BIOREACTOR

Batch bioreactor model

To investigate the fine-tuning capabilities using human feedback, we consider the batch bioreactor system

from Srinivasan et al. [13], where penicillin is produced by microorganisms. Assuming an isothermal and ideally mixed reactor, the system is described by the following set of ordinary differential equations:

$$\dot{X}_s = \mu_s(S_s)X_s - \frac{u_{\text{inp}}}{\text{V}_\text{s}}\text{X}_\text{s}, \tag{7a}$$

$$\dot{S}_s = -\frac{\mu_s(S_s)X_s}{Y_x} - \frac{vX_s}{Y_p} + \frac{u_{\text{inp}}}{V_s}(S_{\text{in}} - S_s), \tag{7b}$$

$$\dot{P}_s = vX_s - \frac{u_{\text{inp}}}{V_s}P_s, \tag{7c}$$

$$\dot{V}_s = u_{\text{inp}}, \tag{7d}$$

where

$$\mu_s(S_s) = \frac{\mu_m S_s}{K_m + S_s + \left(\frac{S_s^2}{K_i}\right)}. \tag{8}$$

The concentration of biomass producing the pharmaceutical product is denoted by X_s and the concentration of penicillin by P_s. Substrate at concentration S_s is supplied to the microorganisms via the input feed u_{inp}. The reactor volume is described by V_s. Overall, the system is defined by the four physical states $x_{\text{phys}} = (X_s, S_s, P_s, V_s)^T$ and one control input $u_{\text{phys}} = \left(u_{\text{inp}}\right)^T$. The kinetics of the bioprocess are defined by μ_m, K_m, K_i and v. The microorganisms exhibit substrate inhibition as described by Equation (8), meaning that a higher substrate concentration leads to a decrease of the kinetic term $\mu_s(S_s)$ for biomass growth. The yield coefficients of biomass and penicillin are Y_x and Y_p, respectively. The concentration of the inlet feed is denoted by S_{in}.

The process aim is to produce as much penicillin as possible in a fixed batch time, where the concentration of biomass is restricted to $X_{s,\,\max} = 3.7$ mol L^{-1} to ensure sufficient supply of oxygen. The system can be controlled using MPC, where the corresponding stage cost is formalized by:

$$l(x_t, u_t) = -\left(P_{s,t} - \Delta u_{\text{inp},t}^2\right). \tag{9}$$

For smooth control trajectories changes of the control input are penalized, where $\Delta u_{\text{inp},t} = u_{\text{inp},t} - u_{\text{inp},t-1}$.

Design of approximate MPC

Approximate model predictive control (AMPC) circumvents the computational limitations inherent to conventional MPC by employing NN-based function approximators to learn the mapping from system states to optimal control actions [14]. This enables the deployment of predictive control strategies in time-critical applications where online optimization would be computationally prohibitive. However, the achieved accuracy highly depends on the available training data. We therefore investigate the fine-tuning capabilities using human feedback for an AMPC controller with suboptimal performance.

A MPC controller is designed for the batch bioreactor system according to the stage cost described in Equation (9) and the described constraint with a prediction horizon of 20 min**.** We apply trajectory-based sampling to obtain a dataset for approximating the optimal control law of the MPC. The input data for the trained NN $\text{x}_{\text{AMPC},t} = \left(x_{\text{phys},t}^T, u_{\text{phys},t-1}^T\right)^T \in X_{\text{AMPC}}$ is defined by the states and preceding control inputs to enable learning the control objective as defined in Equation (9). The output data $u_{\text{AMPC},t} = \left(u_{\text{phys},t}\right)^T \in Y_{\text{AMPC}}$ is defined according to the control actions. To obtain an AMPC with sufficient potential for improvement, $25,000$ data points are used to train a NN with parameters θ to minimize the following mean square error (MSE) objective:

$$\min_\theta \frac{1}{N}\sum_{i=1}^{N} || \, u_{\text{AMPC},\,0}^{*,[i]} - \pi\left(x_{\text{AMPC},\,0}^{[i]}; \theta\right)||_2^2. \tag{10}$$

Reward model from human preferences

We now design a reward model from human preference according to Christiano et al. [2] to reflect the intentions of a human expert. Following this approach, a set of trajectories $\left\{\tau^1, \dots, \tau^i\right\}$ is sampled to cut out trajectory segments σ out of it. We use MPC trajectories, which are available from training the AMPC, as well as closed-loop evaluations of the AMPC. Overall, multiple iterations are required for reward design until the preference dataset sufficiently covers the state space and the reward model adequately reflects the intended behavior. If the trajectory segments labeled for the preference dataset do not cover sufficiently the state space, incorrect high rewards are might be given to undesired states and control actions. Fine-tuning the policy with this inadequate reward leads to undesired control behavior and is additionally used to generate trajectories. Thus, a broad range of different control trajectories are included in the preference dataset. Different initial conditions from the feasible state space are used to generate the trajectories and the start of each segment is selected uniformly and randomly out of those trajectories. The segment length L_s is set to ten, reflecting 10 min, sufficient for evaluating the system's current state and estimating its development in this initial investigation. For labeling, 350 snippets are generated. Two segments are randomly selected from the trajectory set $\{\sigma_1, \sigma_2, \dots, \sigma_{350}\}$, and presented to a process expert, who indicates which trajectory is preferable for system control, or labels them as equally. The segment data and corresponding label are saved automatically and indistinguishable pairs are skipped.

In total $2,133$ preferences are collected for the reward dataset according to the following labeling criteria: Segments with constraint violations (CVs), where biomass concentration exceeds 3.7 mol L^{-1}, are disfavored, though if both segments exhibit CVs, they are rated equally. Similarly, substrate concentrations $S_s \geq 1$ mol L^{-1} are penalized due to substrate inhibition. Further, if one

trajectory's initial penicillin concentration exceeds the other's final concentration, that segment is preferred, as maximizing penicillin production is the primary objective. These decision rules were derived from observed human expert labeling behavior and subsequently implemented as automatic labeling criteria, thereby augmenting the manual expert annotations. Comparisons not meeting these criteria rely on human judgment. The human's expertise is especially interesting in this case study when deciding between the trade-off of higher penicillin concentration and a greater amount of biomass, which promises faster penicillin production in the future.

The collected preferences build the training data for the NN-based reward model. Based on the current observations and actions taken, a reward is obtained from the reward network. Hence, the NN is set up with six inputs $\mathrm{x}_{\hat{r},t} = \left(x_{\mathrm{phys},t}^T, u_{\mathrm{phys},t-1}^T, u_{\mathrm{phys},t}^T\right)^T$, consisting of the four states, the previous and the current control input. The output layer of the network consists of one neuron for the scalar reward $\hat{r}$. The parameters of the preference reward network Φ are trained by minimizing the RLHF objective as defined in Equation (3).

An ensemble of three separately trained NNs is used, a common approach in RLHF [2, 15] that helps counteract reward hacking [16] and overoptimization [17]. Predictions are obtained by averaging across all models.

Investigation of different fine-tuning approaches

Comparison of reward model designs

In a first investigation, the preference reward derived in the previous section is compared to a rigorous encoded reward as typically defined for RL training. Further, a hybrid model combining both approaches is evaluated. The three rewards are formalized as:

$$\hat{r}_{\mathrm{pref}}(x,u) = NN(x,u;\Phi), \tag{13a}$$

$$\hat{r}_{\mathrm{rig}}(x,u) = -l(x,u) - c(x,u), \tag{13b}$$

$$\hat{r}_{\mathrm{hybrid}}(x,u) = \hat{r}_{\mathrm{rig}}(x,u) + NN_{\mathrm{hybrid}}(x,u;\Phi). \tag{13c}$$

The preference reward $\hat{r}_{\mathrm{pref}}(x,u)$ is obtained from human feedback as described in the previous section. The rigorous reward $\hat{r}_{\mathrm{rig}}(x,u)$ is defined as the negative inverse stage cost from Equation (6.4), transforming the MPC minimization objective into an RL maximization objective. A penalization term $c(x,u)$ for constraint violations is added, weighting violations by factor 20. Both strategies are merged via a hybrid reward $\hat{r}_{\mathrm{hybrid}}$. For this hybrid reward, an NN is trained similarly to how it was described for the preference reward, using the same labeled dataset and loss function for training. However, instead of just passing the NN predictions as reward estimates for the preference calculation according to Equation (3), it is added first to the rigorous reward. The reward for the trajectory segment therefore consists of the rigorous reward penalizing CVs and the NN as an expansion. This hybrid reward is designed to extent the rigorous reward, encoding the main goal and limitations, by the expert knowledge included through the preference dataset.

Evaluating reward models derived from human preference to determine whether they suitably capture the task presents a major challenge in RLHF [7, 8]. As the dimensionality of the present control task is manageable, the reward behavior of reward models can be investigated using heatmaps. Predictions of the three applied reward models are displayed in Figure 1 and 2. Each plot displays the reward prediction dependent on two of the six inputs for reward estimation. The remaining inputs are kept constant.

Figure 1 displays the reward prediction as a function of biomass and penicillin concentration. High concentrations of penicillin are expected to be rewarded, and violations of constraints penalized with small rewards. The rigorous model meets this expectation. As long as the amount of biomass is not violating the constraint, it does not influence the rigorous reward. However, CVs are clearly reflected by the model, as the reward significantly decreases. The penalty for CVs is comparatively much higher than the reward for increasing penicillin concentration. The hybrid model is almost identical to the rigorous reward, but a small influence of the biomass concentration is noticeable. Smaller amounts of penicillin receive higher rewards if a high biomass concentration is present, showing the influence of the preference dataset. The reward predictions from the preference model display a different mapping. The distinction between higher and lower penicillin concentrations becomes more noticeable as CVs are relatively less penalized compared to other points, especially when compared to the rigorous and hybrid reward. Similar to the hybrid model, higher amounts of biomass are favored. From analyzing the dependency of biomass and penicillin concentration, the hybrid model seems to mainly align with the rigorous reward and adopt the rewarding of high biomass concentrations from the human preferences.

Figure 2 plots rewards against penicillin and substrate concentration, depicting more clearly the difference between hybrid and rigorous rewards. The rigorous reward is not influenced by the amount of substrate. A high concentration of substrate hindrances biomass growth and thereby penicillin production. It has been observed that some RL agents get stuck in local maxima, feeding a lot of substrate. This behavior is specifically included in the preference dataset by sampling trajectories from these agents. Therefore, both, preference and hybrid reward yield smaller rewards for large amounts of substrate.

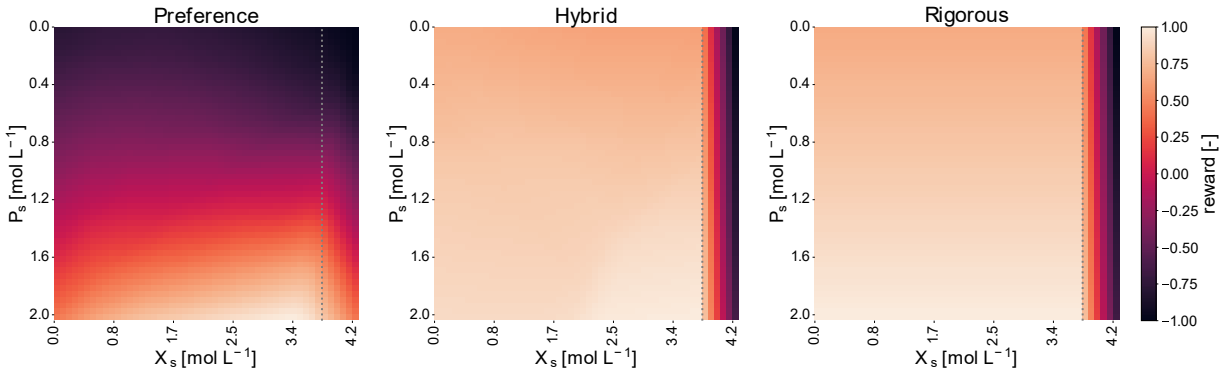

Figure 1: Evaluation of reward predictions for the preference reward model from human feedback, hybrid reward and rigorous reward. The reward is predicted in dependence of penicillin concentration P_s and biomass concentration X_s, where all other reward inputs are kept constant. Constraint indication by grey dotted line.

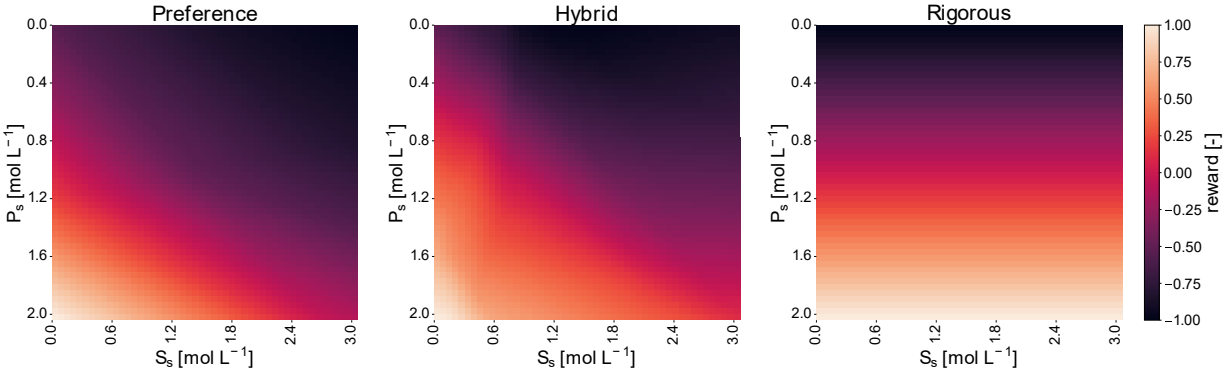

Figure 2: Evaluation of reward predictions for the preference reward model from human feedback, hybrid reward and rigorous reward. The reward is predicted in dependence of penicillin concentration P_s and substrate concentration S_s, where all other reward inputs are kept constant.

To further investigate the rewards, their fine-tuning capabilities with respect to the previously derived AMPC are investigated. The pretrained AMPC is implemented as the policy in the DDPG algorithm [19] using Stable Baselines3 [20]. For each reward model, 25 agents are learned in parallel with different seeds. Each agent is learned for $5 \cdot 10^5$ transitions, frequently evaluated and the best fine-tuned agents are saved. The closed-loop evaluations of the best three models obtained from each reward are listed in Table 1. The MPC control law π^{MPC} and the AMPC π^{AMPC} serve as benchmarks. Indices refer to the reward used for fine-tuning. To evaluate all obtained policies from the fine-tuned agents, closed-loop trajectories with a length of 150 steps for 100 initial conditions are analyzed. The average return $\bar{G}$ is defined by the undiscounted, accumulated rigorous reward and is used as the evaluation metric, along with its standard deviation. Penalization of CVs is not considered in the return. Furthermore, the number of CVs for all initial conditions is summed up as n_{CV} and the average deviation $\bar{d}_{\mathrm{CV}}$ from the constraint is computed and listed. None of the best fine-tuned agents violate the constraint in any of the closed-loop trajectories and all agents improve the average return of the AMPC. The hybrid reward leads to the highest average reward. The best agent obtained with the rigorous reward yield a slightly higher average return than the agent learned with the preference reward.

Table 1: Evaluation of fine-tuning capabilities of AMPC with different reward designs.

Reward	Policy	$\bar{G}$	CV	
			n_{CV}	$\bar{d}_{\mathrm{CV}}$
	π^{MPC}	116.16 ± 21.66	0	-
	π^{AMPC}	96.36 ± 15.97	3	0.053
Preference	$\pi^{\mathrm{AMPC}}_{\mathrm{pref}}$	106.83 ± 22.32	0	-
		106.48 ± 18.83	0	-
		103.46 ± 18.14	0	-
Rigorous	$\pi^{\mathrm{AMPC}}_{\mathrm{rig}}$	108.85 ± 19.57	0	-
		105.76 ± 21.80	0	-
		104.19 ± 17.95	0	-
Hybrid	$\pi^{\mathrm{AMPC}}_{\mathrm{hybrid}}$	110.05 ± 21.89	0	-
		104.32 ± 18.27	0	-
		100.93 ± 16.75	0	-

Influence of length of trajectory segments

The previous investigation of different models

revealed that the preference model does not clearly localize the constraint. A snippet is labeled unfavored if a CV takes place at any point. The applied labeling strategy does not differentiate between the severity of constraint violations or when they occur within the segment. Consequently, the preference model may inadequately reward states with favorable biomass concentrations for penicillin production if they are associated with segments containing any constraint violation. It is presumed that with smaller snippets, the localization of the constraint can be more clearly encoded. To investigate the influence of the length of trajectory segments, shorter trajectory snippets of $L_s = 5$ are labeled with the aim to clearly localize the constraint. Longer snippets of $L_s = 20$ are labeled to investigate the representation of long-term effects in the reward model. Furthermore, a combined dataset is compiled by sampling equally from each of the three segment lengths, with each contributing one third of the total data. Both new datasets are generated following the same procedure and triplet count as the initial dataset. Figure 3 shows heatmaps of reward dependency on biomass and penicillin concentration, illustrating the reward mapping and constraint adaptation of all models.

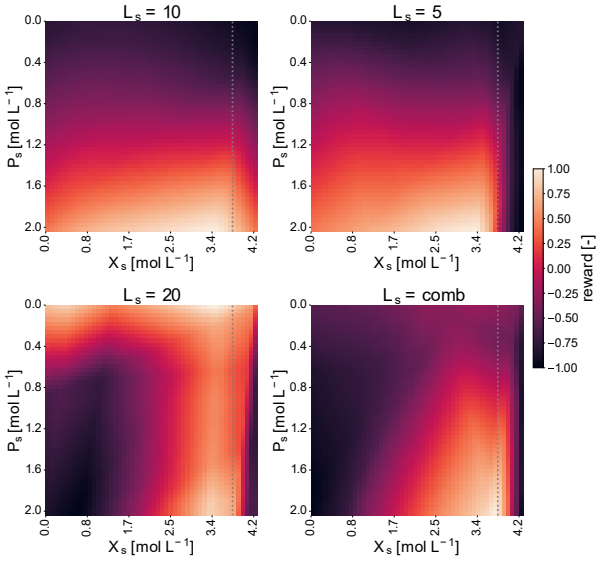

Figure 3: Evaluation of reward predictions for the preference reward models derived from different segment length. The reward is predicted in dependence of penicillin concentration P_s and biomass concentration X_s, where all other reward inputs are kept constant. Constraint indication by grey dotted line.

The model obtained from shorter trajectory length $L_s = 5$ displays the constraint restriction more clearly than the previous model from a segment length of $L_s = 10$. Slightly before reaching the maximum permitted biomass concentration, the reward decreases. Additionally, the penalization is much higher. Therefore, it is concluded that the shorter segment lengths help to identify and penalize constraints more clearly. The reward model, which is trained based on longer preference segments, also does not identify the constraints sufficiently. However, the penalization for larger violations is stronger than for the preference reward from $L_s = 10$. Furthermore, the reward from $L_s = 20$ differs from the previous models as it mainly rewards high biomass concentrations instead of product concentration. Longer segments better illustrate the development of the system and the slope of penicillin increase can be observed in more detail. Therefore, the human expert often prefers trajectories with a higher biomass concentration, as the slope of the penicillin production indicates a faster production, which is more advantageous in the long term. The incorrect high reward for very low amounts of penicillin is probably reasoned in a sparse representation of this data. The reward model obtained from the mixed dataset combines the different reward estimations. High penicillin concentrations in combination with a high amount of biomass are encouraged the most with a high reward, as a high amount of product is obtained quickly. Similar to the reward model trained with long segments, higher amounts of biomass are favored for constant amounts of product. The constraint for the biomass is clearly encoded and the reward decreases as soon as the constraint is violated.

Table 2: Evaluation of fine-tuning capabilities of AMPC with preference rewards trained with different segment lengths L_s.

Reward	Policy	$\bar{G}$	CV	
			n_{CV}	$\bar{d}_{CV}$
	π^{MPC}	116.16 ± 21.66	0	-
	π^{AMPC}	96.36 ± 15.97	3	0.053
$L_s = 10$	π^{AMPC}_{pref}	106.83 ± 22.32	0	-
		106.48 ± 18.83	0	-
		103.46 ± 18.14	0	-
$L_s = 5$	$\pi^{AMPC}_{L_s=5}$	101.68 ± 31.02	0	-
		99.20 ± 23.52	0	-
		94.11 ± 31.22	0	-
$L_s = 20$	$\pi^{AMPC}_{L_s=20}$	101.44 ± 17.31	0	-
		109.21 ± 21.23	171	0.026
		111.53 ± 20.91	283	0.041
$L_s =$ comb	$\pi^{AMPC}_{L_s=\text{comb}}$	108.37 ± 19.45	0	-
		108.34 ± 20.86	0	-
		108.18 ± 21.94	0	-

The three best policies obtained from the different reward models are listed in Table 2, with the previous references. Agents learned with the reward from a segment length of $L_s = 5$ achieve the lowest average return, with only some exceeding AMPC performance. However, no CV occur for these agents. This supports the

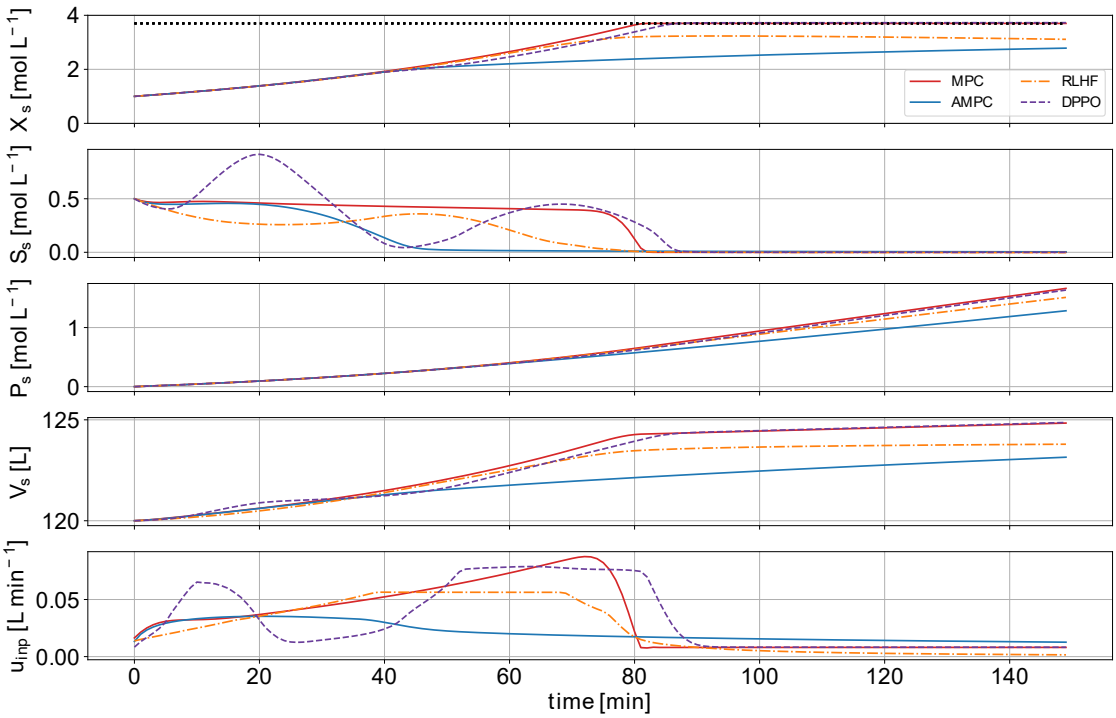

Figure 4: Closed-loop trajectories of MPC (red, solid), AMPC (blue, solid), reference policy from fine-tuned policy using RLHF (orange, dash-dot), and optimized policy using DPPO (purple, dashed). The constraint is marked by the dotted black line.

observations from the heatmap, that the constraints are recognized, but high amounts of biomass are not rewarded sufficiently. Correspondingly, the agents, learned with the reward of length $L_s = 20$, produce higher concentrations of penicillin in the same period of time under more CVs. The agents learned with the reward from the mixed dataset achieve the best results in terms of a high return without CVs, combining the strengths of all datasets.

Comparison of RLHF and DPPO

As the presented RLHF approach is prone to errors arising from the interdependence of reward model design and RL training, we investigate DPPO as an alternative approach to enable human preferences for fine-tuning. To ensure a valid comparison, the same dataset with $L_s = 10$ used for deriving the first preference reward is applied. It has been found that DPPO is sensitive to hyperparameters, especially the regularization factor $\lambda_{\bar{s}}$, which is used in the probability distribution to prevent both distances from increasing [9]. Therefore, a small grid search is performed and the best three fine-tuned agents obtained are listed together with the benchmark policies and reference policies from RLHF in Table 3.

The average return of the best policy optimized with DPPO outperforms the fine-tuned agents from RLHF in terms of the achieved return. However, the total number of CVs is very high for almost all models. The cause for that many CVs becomes evident when considering a closed-loop trajectory as shown in Figure 4. The development of biomass concentration, controlled by the policy optimized with DPPO, shows very similar behavior to the optimal MPC trajectory. It also converges to a high biomass concentration, which leads to a high product concentration. However, the DPPO policy converges to a biomass concentration which is slightly higher than the constraint, resulting in the high return in combination with many, but small CVs. As the optimized policy still presents an approximation, small approximation errors are to be expected.

Table 3: Evaluation of fine-tuning capabilities of AMPC with RLHF and DPPO.

Reward	Policy	$\bar{G}$	CV	
			n_{CV}	$\bar{d}_{CV}$
	π^{MPC}	116.16 ± 21.66	0	-
	π^{AMPC}	96.36 ± 15.97	3	0.053
RLHF	π^{AMPC}_{pref}	106.83 ± 22.32	0	-
		106.48 ± 18.83	0	-
		103.46 ± 18.14	0	-
DPPO	π^{AMPC}_{DPPO}	112.19 ± 20.79	1, 857	0.016
		111.29 ± 20.86	4, 376	0.022
		109.73 ± 19.60	1, 575	0.013

CONCLUSION

In this work, we demonstrated how to implement and enhance reinforcement learning from human feedback for controlling a biochemical batch reactor. Deriving the reinforcement learning reward from a human process expert's feedback enables encoding domain knowledge regarding both safety-critical and long-term behavior. Our case study visualized how different reward model designs and trajectory segment lengths influence the optimization objective. A hybrid reward function effectively combines formalizable constraints with intuitive expert knowledge, while varying segment lengths allows the incorporation of both short- and long-term control strategies. Finally, direct preference-based policy optimization offers a promising fine-tuning alternative that circumvents explicit reward model design.

In future research, we will investigate the potential of multi-trajectory ranking over pairwise comparisons for improving learning-based controllers.

ACKNOWLEDGEMENTS

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 466380688 – within the Priority Program "SPP 2331: Machine Learning in Chemical Engineering".

AUTHOR IDENTIFIERS

Author ORCIDs:
Gerold H: 0009-0002-9287-0128
Brandner D: 0000-0003-1500-7064
Lucia S: 0000-0002-3347-5593

REFERENCES

1. Sutton RS, Barto AG. Reinforcement Learning: An Introduction, Second edition. The MIT Press (2018)
2. Christiano PF, et al. Deep reinforcement learning from human preferences. *Adv Neural Inf Process Syst* 30 (2017)
3. Agarwal S, Almeida D, Askell A, Christiano P, Hilton J, Jiang X, Kelton F, Leike J, Lowe R, Miller L, Mishkin P, Ouyang L, Ray A, Schulman J, Simens M, Slama K, Wainwright C, Welinder P, Wu J, Zhang C. Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems 35 :27730-27744 (2022). https://doi.org/10.52202/068431-2011
4. Palan M, Shevchuk G, Charles Landolfi N, Sadigh D. Learning reward functions by integrating human demonstrations and preferences. Robotics: Science and Systems XV : (2019). https://doi.org/10.15607/rss.2019.xv.023
5. Hejna J, Sadigh D. Inverse preference learning: preference-based RL without a reward function. Advances in Neural Information Processing Systems 36 :18806-18827 (2023). https://doi.org/10.52202/075280-0825
6. Kaufmann T, et al. A survey of reinforcement learning from human feedback. *arXiv* 2312.14925 (2023) https://doi.org/10.48550/arXiv.2312.14925
7. Casper S, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv* 2307.15217 (2023) https://doi.org/10.48550/arXiv.2307.15217
8. Lambert N, Calandra R. The alignment ceiling: Objective mismatch in reinforcement learning from human feedback. *arXiv* 2311.00168 (2023) https://doi.org/10.48550/arXiv.2311.00168
9. An G, Kim KM, Kosaka N, Lee J, Song HO, Zuo X. Direct preference-based policy optimization without reward modeling. Advances in Neural Information Processing Systems 36 :70247-70266 (2023). https://doi.org/10.52202/075280-3078
10. Ermon S, Finn C, Manning CD, Mitchell E, Rafailov R, Sharma A. Direct preference optimization: your language model is secretly a reward model. Advances in Neural Information Processing Systems 36 :53728-53741 (2023). https://doi.org/10.52202/075280-2338
11. Nika A, et al. Reward model learning vs. direct policy optimization: A comparative analysis of learning from human preferences. *arXiv* 2403.01857 (2024) https://doi.org/10.48550/arXiv.2403.01857
12. Xu S, et al. Is DPO superior to PPO for LLM alignment? A comprehensive study. *arXiv* 2404.10719 (2024) https://doi.org/10.48550/arXiv.2404.10719
13. Srinivasan B, et al. Dynamic optimization of batch processes: II. Role of measurements in handling uncertainty. *Comput Chem Eng* 27:27-44 (2003) https://doi.org/10.1016/S0098-1354(02)00117-5
14. Karg B, Lucia S. Efficient representation and approximation of model predictive control laws via deep learning. IEEE Trans. Cybern. 50:3866-3878 (2020). https://doi.org/10.1109/tcyb.2020.2999556
15. Chittepu Y, Finn C, Hejna J, Knox W, Niekum S, Park R, Rafailov R, Sikchi H. Scaling laws for reward model overoptimization in direct alignment algorithms. Advances in Neural Information Processing Systems 37 :126207-126242 (2024). https://doi.org/10.52202/079017-4009
16. Zhang S, et al. Improving reinforcement learning from human feedback with efficient reward model ensemble. *arXiv* 2401.16635 (2024) https://doi.org/10.48550/arXiv.2401.16635
17. Coste T, et al. Reward model ensembles help

mitigate overoptimization. *arXiv* 2310.02743 (2024) https://doi.org/10.48550/arXiv.2310.02743
18. Henderson P, Islam R, Bachman P, Pineau J, Precup D, Meger D. Deep reinforcement learning that matters. AAAI 32: (2018). https://doi.org/10.1609/aaai.v32i1.11694
19. Lillicrap TP, et al. Continuous control with deep reinforcement learning. *arXiv* 1509.02971 (2019) https://doi.org/10.48550/arXiv.1509.02971
20. Raffin A, et al. Stable-Baselines3: Reliable reinforcement learning implementations. *J Mach Learn Res* 22:1-8 (2021)

© 2026 by the authors. Licensed to PSEcommunity.org and PSE Press. This is an open access article under the creative commons CC-BY-SA licensing terms. Credit must be given to creator and adaptations must be shared under the same terms. See https://creativecommons.org/licenses/by-sa/4.0/