UBP2: UNCERTAINTY-BALANCED PREFERENCE PLANNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Preference-based RL provides an approach to learning reward models from pairwise comparisons of behaviors, bypassing the need for explicit reward design. However, existing methods typically rely on passive data collection and suffer from poor sample efficiency, especially during the early stages of learning. We introduce a model-based approach that actively directs exploration by jointly reasoning over uncertainties in the reward, dynamics, and value functions. Our method, Uncertainty-Balanced Preference Planning (UBP2), uses ensembles of reward, dynamics, and value function models to evaluate candidate trajectories according to a unified score that combines expected reward, terminal value, and epistemic uncertainty. Planning under this objective yields an explicit tradeoff between exploitation and information acquisition without requiring ad hoc exploration heuristics. Empirically, experiments on the Meta-World benchmark show UBP2 achieves substantially higher sample efficiency than model-free preference-based methods and non-optimistic model-based baselines using simulated and real human preference annotations. The source code of this project is released at `https://anonymous.4open.science/r/ubp2-B2A1`.

1 INTRODUCTION

In reinforcement learning, agents must learn effective behavior from interaction, often without prior knowledge of the dynamics of the environment or access to a densely specified reliable reward signal. This challenge is particularly acute in unsafe environments, where interactions are costly or risky, forcing agents to identify favorable actions and infer dynamics using as few samples as possible. Meanwhile, designing accurate reward functions for complex tasks is difficult and often leads to misspecified objectives that fail to capture the true intention Abel et al. (2022); Singh et al. (2009). Preference-based reward learning (PbRL) addresses this limitation by replacing manual reward design with pairwise trajectory comparisons provided by an oracle, such as a human Christiano et al. (2017); Wirth et al. (2017). Instead of relying on explicit numerical rewards, the agent learns a reward model from preferences, which can guide decision-making Lee et al. (2021). However, effectively leveraging such learned rewards requires reasoning over hypothetical trajectories rather than relying only on real environment interactions. Model-based reinforcement learning offers a natural solution by learning a

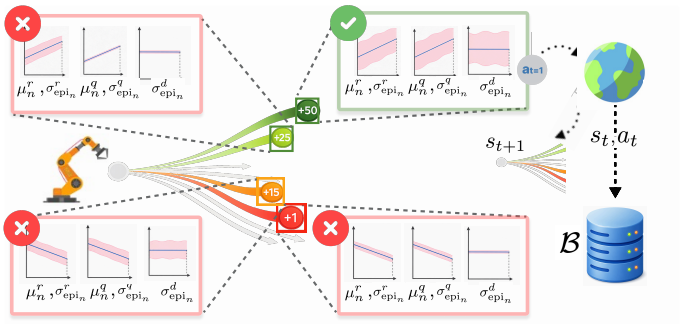


Figure 1: Illustration of planning in UBP2. The agent rolls out the dynamics model to generate imagined trajectories. The planner selects the trajectory maximizing expected return and epistemic uncertainty by solving Equation (1) and then executes $a_{t=1}$.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

predictive dynamics model to enable planning over imagined trajectories, thus improving sample efficiency, particularly during early learning stages. Moerland et al. (2022); Nagabandi et al. (2017).

In this paper, we propose an approach that learns environment dynamics and a reward model from preference feedback, and uses this learned reward as the only supervision to train a value function and policy. A model-based planner is used for efficient exploration and action selection under a limited preference feedback budget. After the budget is exhausted, control transitions to the learned policy.

In the context of PbRL, purely model-free schemes passively collect trajectories (often via random or heuristic exploration) and only afterward fit the reward model Hejna & Sadigh (2022); Liang et al. (2022); Lee et al. (2021). In contrast, our model-based approach plans under the learned reward, dynamics, and value ensembles to actively select trajectories by maximizing a trajectory-level acquisition function that combines predicted cumulative return with epistemic uncertainty in the reward, dynamics, and value estimates. This objective jointly handles exploration and exploitation during learning. This tight coupling of planning and reward learning has been found to empirically accelerate convergence by focusing data collection on areas of interest.

We propose Uncertainty-Balanced Preference Planning (UBP2), which uses ensembles for reward, dynamics, and value models and plans trajectories using a unified score that combines expected return and uncertainty. Executing informative trajectories allows UBP2 to efficiently learn the dynamics model and to learn the reward from preferences. Uncertainty-guided planning is used only when preference feedback is available; once the feedback budget is exhausted, the agent switches to fast policy execution under the learned reward. Our experiments show that uncertainty-guided planning in the preference-based setting achieves earlier success than both non-optimistic model-based preference (MBP) and model-free baselines in most tasks. Our main contributions are:

- We introduce an uncertainty-guided planning objective that combines predicted cumulative return with epistemic uncertainty over reward, dynamics, and value estimates, enabling optimistic exploration that outperforms planning based on return alone or uncertainty in a single component.
- We introduce an optimistic preference label query strategy that prioritizes trajectory segment pairs with high predicted reward and high reward-model epistemic uncertainty.
- We empirically demonstrate on Meta-World manipulation Yu et al. (2021) tasks that UBP2 operating on proprioceptive observations, achieves *on average* improved sample-efficiency compared to state of the art model-free Liang et al. (2022) Liu et al. (2022) and non-optimistic MBP baselines .
- We apply UBP2 to real human preference annotations on two Meta-World tasks and show that UBP2 achieves improved sample efficiency compared to a non-optimistic MBP baseline.

2 Related Work

We now compare our method to areas of closely-related prior work. UBP2 is a preference-based reinforcement learning approach that relies on uncertainty-driven exploration.

2.1 Preference-based Reinforcement Learning

Although comparisons provide an intuitive form of feedback (Christiano et al., 2017), achieving high sample-efficiency in human feedback is a challenge Hejna & Sadigh (2022). Furthermore, collecting environment interactions for online learning can be impractical, particularly in high-risk environments. To address feedback sample-efficiency, prior work explored smart querying strategies, such as selecting the most informative queries through disagreement sampling after collecting an initial batch of seed interactions Lee et al. (2021). Few-Shot PbRL Hejna & Sadigh (2022) uses a pretrained reward model on similar tasks for meta-learning the reward of the target task. Metcalf et al. (2022) uses a pre-collected dataset of transitions to encode environment dynamics in reward learning. MoP-RL Liu et al. (2023) handles learning efficiency by relying on pre-collected human demonstrations and a pretrained dynamics model to improve sample-efficiency before reward learning begins. In contrast, our method, to the best of our knowledge, is the first model-based PbRL approach to learn the reward function entirely from scratch with no precollected agent interactions, no pretrained reward models, and no offline pretrained dynamics model. This setup forces the agent to learn purely through online environment interactions and preference feedback, providing a more realistic test of feedback sample efficiency.

2.2 EPISTEMIC UNCERTAINTY IN MODEL-BASED REINFORCEMENT LEANING (MBRL)

Prior work on MBRL estimate multiple sources of uncertainty for different intentions. MOPO estimates epistemic uncertainty in learned dynamics and penalizes the reward proportionally to this uncertainty, to discourage rollouts that leave the dataset support Yu et al. (2020). RUNE, however, uses epistemic reward uncertainty optimistically as an intrinsic exploration bonus to improve feedback efficiency in PbRL Liang et al. (2022). UNISafe treats epistemic uncertainty of the dynamics model as a proxy for Out Of Distribution (OOD) risk, so uncertainty is used to synthesize a safety filter Seo et al. (2025). Feng et al. (2023) uses epistemic uncertainty in the value function as a regularizer of test-time planning, so candidate action sequences that appear high return but uncertain are de-prioritized, reducing extrapolation error during online finetuning. In contrast to using uncertainty in a single component of the world model, UBP2 plans with a unified trajectory score that combines predicted reward, terminal value, and epistemic uncertainty of the reward, dynamics, and value models, explicitly balancing exploration and exploitation in PbRL in a single planning objective.

3 PRELIMINARIES

We consider a discrete-time Markov decision process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, p, r, \gamma, \rho_0)$, where $s \in \mathcal{S}$ with dimension d_s and $a \in \mathcal{A}$ with dimension d_a denote the continuous state and action spaces, $p : \mathcal{S} \times \mathcal{A} \mapsto \mathcal{S}$ is the unknown transition distribution that captures process noise, $r : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is a generic reward function, ρ_0 is the initial state distribution and $\gamma \in (0, 1)$ is the discount factor. The transition dynamics $p^*(s' \mid s, a)$ is assumed to be unknown. A parametric dynamics model $p_\phi(s' \mid s, a)$ is learned from data collected through online interaction with the environment and used during planning.

During model-based planning, the Model Predictive Control (MPC) planner generates the control actions using the models learned after the n^{th} episode of interaction with the real environment, $n \in \{1, ..., N\}$. This implicitly defines a policy through the solution of the planning objective, of which its first action is executed in the real environment; we denote this induced policy by π_n^{ind}. In addition, we learn an explicit parametric policy $\pi_\varphi^{\text{learn}}$, trained using Equation (6).

The objective is to learn a control policy $\pi : \mathcal{S} \mapsto \mathcal{A}$ that maximizes the expected discounted return over H timesteps under real dynamics p^* and the true reward r^* denoted by η, despite the absence of direct true reward observations. This desired policy, denoted π^*, is: $\pi^\star = \arg\max_\pi \ \eta(\pi, p^*) = \arg\max_\pi \ \mathbb{E}_{\pi, p^*}\left[\sum_{t=0}^{H-1} \gamma^t r^*(s_t, a_t)\right]$.

3.1 PREFERENCE-BASED REWARD LEARNING

In the PbRL setting, the agent does not observe samples from the ground-truth reward function $r^*(s, a)$.Instead, learning is guided by preference-based feedback in the form of pairwise comparisons between trajectory segments, where segments are uniformly sampled over the entire trajectory, rather than restricted to any particular sub-trajectory. Because preference queries are available only on a limited budget, their acquisition must be carefully optimized through informative exploration.

Given trajectory segments τ_i and τ_j each of length L, a preference oracle indicates which segment is preferred. We model preferences using a learned reward function $r_\theta(s, a)$. Reward model parameters, θ, are optimized by maximizing the log-likelihood of observed preference labels using the Bradley-Terry model, in which the probability that τ_i is preferred over τ_j is $P(\tau_i \succ \tau_j) = \frac{\exp\left(\sum_{t=0}^{L-1} r_\theta(s_t^i, a_t^i)\right)}{\exp\left(\sum_{t=0}^{L-1} r_\theta(s_t^i, a_t^i)\right) + \exp\left(\sum_{t=0}^{L-1} r_\theta(s_t^j, a_t^j)\right)}$.

3.2 MODELING TRUE REWARD IN PBRL

In PbRL, preference data cannot uniquely identify r^*, as the oracle only reveals comparisons, i.e. information about which segment has larger relative utility, not the actual reward scale Wirth et al. (2017). This creates an *equivalence class* of rewards - a set of functions that are *indistinguishable from the perspective of the preference oracle* because they induce the exact same preference outcomes. Rather than claiming that PbRL recovers the literal, unobserved r^*, we interpret r^* as a latent utility function that generates preferences and contend that the learning procedure produces some *proxy reward*, denoted $r(s, a)$, within the same equivalence class as r^*, and hence is the subject of our study

instead. We list further conditions on this equivalence class in Section 5, Assumption 5.2, which makes our theoretical analysis feasible. For more discussion of modeling the true reward in PbRL, please refer to Appendix D.1. Unless otherwise noted, henceforth the "reward" we refer to in all subsequent methodology and theory denotes the proxy reward r. The notation r^* and the phrase "true reward" are reserved specifically when referring to the unobserved true reward.

4 UBP2: Uncertainty-Balanced Preference Planning

We present UBP2, (Algorithm 1), our approach to efficient optimistic exploration in MBRL. UBP2 uses ensembles of reward, dynamics, and value function models to guide exploration using epistemic uncertainty, enabling preference-based learning that is sample-efficient and informative.

Uncertainty Representation: The uncertainty in the dynamics and value function models is estimated through ensemble disagreement, following previous work Kidambi et al. (2021); Yu et al. (2020), which reflects total uncertainty that encompasses both epistemic and aleatoric effects. For the reward model, we adopt the Jensen–Rényi divergence (JRD) R'enyi (1961) to estimate epistemic uncertainty, following Seo et al. (2025). JRD explicitly separates epistemic and aleatoric uncertainty and has been shown to outperform total-uncertainty and density-based estimates in downstream tasks Seo et al. (2025). In our setting, we apply JRD for measuring epistemic uncertainty only to the reward model, and use disagreement based total uncertainty for the dynamics and value function ensembles. We found doing so to be empirically sufficient for our results. Our approach is based on TD-MPC2 Hansen et al. (2024a), using MPC as the primary control mechanism during the preference-based reward learning phase to allow efficient exploration under uncertainty. Unlike TD-MPC2 our approach does not assume access to scalar rewards throughout training, rather learns the reward from preference feedback and transition to a learned policy initialized from the value function once the preference budget is exhausted and model uncertainty decreases.

4.1 Uncertainty-Guided Optimistic Planning

At each decision step t in the real environment during the n^{th} episode, UBP2 optimizes the parameters $(\mu_{\text{mpc}}, \sigma_{\text{mpc}})$ of a time-dependent Gaussian distribution over action sequences by approximately solving Equation (1).

$$\underset{\mu_{\text{mpc}},\,\sigma_{\text{mpc}}}{\arg\max}\ \mathbb{E}_{\mathbf{a}_t}\left[\sum_{t=0}^{H-1}\gamma^t\left(\mu_n^r(\hat{s}_t,a_t)+\lambda_r\sigma^r_{\text{epi}_n}(\hat{s}_t,a_t)+\lambda_d\sigma^d_{\text{epi}_n}(\hat{s}_t,a_t)\right)+\gamma^H\left(\mu_n^q(\hat{s}_H,a_H)+\lambda_q\sigma^q_{\text{epi}_n}(\hat{s}_H,a_H)\right)\right], \quad (1)$$

where $\{a_t\}_{t=0}^{H-1} \sim \mathcal{N}(\mu_{\text{mpc}}, \sigma^2_{\text{mpc}})$ and $a_H \sim \pi^{\text{learn}}(\cdot \mid \hat{s}_H)$.

Here, $(\mu_{\text{mpc}}, \sigma_{\text{mpc}})$ parameterize the planner's time-dependent action distribution, while $\mu_n^{(\cdot)}$ and $\sigma^{(\cdot)}_{\text{epi}_n}$ denote ensemble means and epistemic uncertainties of the learned reward, dynamics, and value models in the n^{th} episodes of actual environment interaction (hence, these means are fixed for each planning procedure). During planning, the trajectory $(\hat{s}_1, \ldots, \hat{s}_H)$ is generated by rolling out the learned dynamics model conditioned on sampled action sequences $p_\phi(\hat{s}' \mid \hat{s}, a)$. Only the first action a_0 of the optimized sequence is executed in the real environment, after which the planning process is repeated at the next timestep. For long horizon reasoning beyond the planning horizon, we learn a value function $q_\psi(s, a)$ using temporal-difference learning under the learned reward using $r_\theta(s, a)$.

The coefficients λ_r, λ_d, and λ_q control the relative contribution of the reward, the dynamics, and the value uncertainty, respectively, and are automatically tuned online. We use the "autotuning" approach from Sukhija et al. (2025a) to optimize each coefficient using the current policy π and a Polyak-averaged target policy $\bar{\pi}$ to define the loss:

$$\mathcal{L}(\lambda) = \mathbb{E}_{s\sim\mathcal{B}}\Big[\log(\lambda)\big(\sigma^{(\cdot)}_{epi}(s,\pi(s)) - \sigma^{(\cdot)}_{epi}(s,\bar{\pi}(s))\big)\Big]. \quad (2)$$

4.2 Optimistic Preference Selection

Prior preference-based RL methods use query strategies such as uniform sampling, disagreement or entropy-based uncertainty, and coverage-based diversity Lee et al. (2021). These strategies are reward-agnostic; they treat all regions of the state-action space as equally worth querying, provided the models disagree or are uncertain in such regions. Disagreement, however, can be high in regions

Algorithm 1: UBP2

Input: Replay buffer $\mathcal{B}$, preference buffer $\mathcal{P}$
Params: model ensembles $(p_\phi, r_\theta, q_\psi)$, policy $\pi_\varphi^{\text{learn}}$
Hyperparameters: horizon H, total steps after N episodes NT, pref period F_{pref}, pref budget N_{pref}
Initialize $t \leftarrow 0$, $n_{\text{pref}} \leftarrow 0$,$s \leftarrow$ env.reset
while $t \leq NT$ **do**
  if $(n_{\text{pref}} < N_{\text{pref}})$ **then**
    $a \leftarrow \text{MPC}(s; \lambda_r, \lambda_d, \lambda_q)$
  else
    $a \leftarrow \pi_\varphi^{\text{learn}}(s)$
  end if
  Step env: $(s', r) \leftarrow \text{env.step}(a)$;
  $\mathcal{B} \leftarrow \mathcal{B} \cup (s, a)$; $s \leftarrow s', t \leftarrow t + 1$
  Update p_ϕ, r_θ, q_ψ using Equations (3) to (5)
  Update $\pi_\varphi^{\text{learn}}$ using Equation (6)
  Update $\{\lambda_r, \lambda_d, \lambda_q\}$ using Equation (2)
  if episode done (t==T) **then**
    if F_{pref} and $(n_{\text{pref}} < N_{\text{pref}})$ **then**
      $\mathcal{P} \leftarrow \mathcal{P} \cup (\tau_1, \tau_2, y)$ using Algorithm 2;
      update n_{pref}
    end if
    $s \leftarrow \text{env.reset}()$
  end if
end while

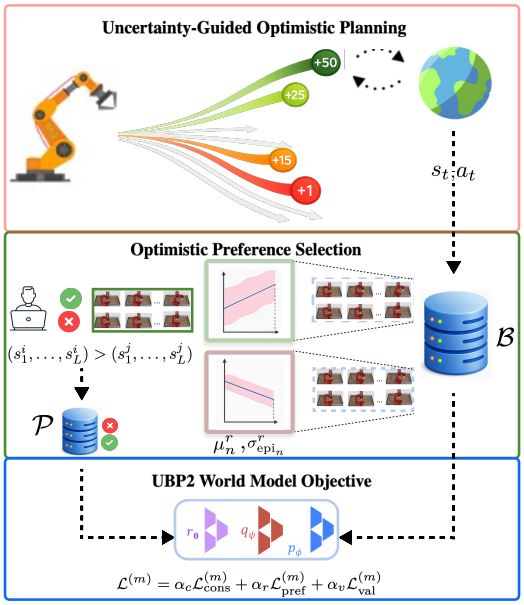


Figure 2: UBP2 uses ensembles of reward, dynamics, and value models to perform optimistic planning over imagined trajectories by combining predicted returns with epistemic uncertainty. Preferences are then selected optimistically from uniformly sampled segments of executed trajectories. Both trajectories and preferences are used to jointly train the dynamics, reward, and value models.

that are irrelevant to solving the task (e.g., states the policy rarely needs to revisit once it has learned to avoid them). Investing preference labeling budget resolving uncertainty in such regions is wasteful. Our labeling approach (Algorithm 2) departs from pure disagreement-based selection in two ways: **(1) Optimistic scoring**: Pairwaise trajectories are ranked using an Upper Confidence Bound (UCB) criterion that combines the predicted preference probability and the reward model's epistemic uncertainty. This optimistic acquisition strategy prioritizes queries that are likely to contain high reward behavior but remain uncertain under the current reward ensemble. Consequently, the annotation effort is focused on rewarding regions of the state space that are relevant to learning. **(2) Global selection**: Pairwaise trajectories are ranked globally across the replay buffer. This reduces the chance of missing a highly informative comparison that a small local batch may not include. We empirically validate the benefit of optimistic scoring and global selection in Section 6.

4.3 UBP2 World Model Objective

Model objective. UBP2 is based on the TD-MPC2 world-model objective, with the following key differences: (i) rewards are learned from preference feedback, (ii) TD targets are computed using predicted rewards, and (iii) reward and dynamics models are represented as ensembles.

Given a horizon segment H and a preference pair segment L, we optimize a loss per-member that combines consistency, reward, and value losses,

$$\mathcal{L}^{(m)} = \alpha_c \mathcal{L}_{\text{cons}}^{(m)} + \alpha_r \mathcal{L}_{\text{pref}}^{(m)} + \alpha_v \mathcal{L}_{\text{val}}^{(m)}.$$

The ensemble members m of the dynamics and rewards models are independently trained as in Seo et al. (2025). To optimize the dynamics model, we compute a rollout $\{\hat{s}_t^{(m)}\}$ and minimize the consistency loss,

$$\mathcal{L}_{\text{cons}}^{(m)} = \frac{1}{H} \sum_{t=0}^{H-1} \|\hat{s}_{t+1}^{(m)} - s_{t+1}\|_2^2, \tag{3}$$

For the reward model, we minimize the preference loss,

$$\mathcal{L}_{\text{pref}}^{(m)} = -\mathbb{E}_{(\tau_i, \tau_j, y) \sim \mathcal{P}} \Big[y \log P_{\theta^{(m)}}(\tau_i \succ \tau_j) + (1-y) \log\big(1 - P_{\theta^{(m)}}(\tau_i \succ \tau_j)\big) \Big], \tag{4}$$

where y is the true preference label provided by a preference oracle, which in our experiments is simulated using the true reward from the environment.

The value function ensemble is trained using temporal-difference regression. For each member m, the TD targets td_t are computed using the predicted reward of the same member,

$$td_t^{(m)} = r_\theta^{(m)}(s_t, a_t) + \gamma \bar{q}_{\min}(s_{t+1}, \pi^{\text{learn}}(s_{t+1})),$$

where $\bar{q}_{\min}$ denotes the target value function. The value loss is:

$$\mathcal{L}_{\text{val}}^{(m)} = \frac{1}{HE_Q} \sum_{t=0}^{H-1} \gamma^t \sum_{j=1}^{E_Q} \Big[\big(q_\psi^{(j)}(\hat{s}_t, a_t) - td_t^{(m)} \big)^2 \Big]. \tag{5}$$

We adopt the stochastic maximum-entropy policy objective from TD-MPC2. The policy $\pi_\varphi(a \mid s)$ is optimized to maximize value while encouraging exploration through entropy regularization over imagined rollouts. The policy objective is given in Equation (6).

$$\mathcal{L}(\varphi) = -\frac{1}{H} \sum_{t=0}^{H-1} \gamma^t \big(\mu^q(\hat{s}_t, \pi_\varphi(\hat{s}_t)) + \alpha_{\mathcal{H}}\, \mathcal{H}(\pi_\varphi) \big), \tag{6}$$

where $\mathcal{H}(\pi_\varphi)$ denotes the policy entropy. Gradients of $\mathcal{L}_\pi$ are taken with respect to the policy parameters φ only, with value estimates fixed during policy optimization.

5 Theoretical Analysis

We provide theoretical motivation for the design of UBP2; however we do not present it as a main contribution of this work, because the theoretical assumptions made in this section differ from our practical implementation, although similar simplifying assumptions are standard insofar as they have appeared in prior work. The purpose of this analysis is to show that, under standard smoothness (RKHS) assumptions and well-calibrated Gaussian process models, jointly planning optimistically over reward, dynamics, and value uncertainty yields finite and infinite-horizon regret bounds that grow sublinearly in the number of episodes. First, we make standard continuity common in literature Curi et al. (2020); Sussex et al. (2023); Sukhija et al. (2023).

Assumption 5.1 (Continuous dynamics, continuous and bounded reward)**.** The true dynamics p^*, all policies $\pi \in \Pi$, and true reward r^* are continuous. Furthermore, r^* is bounded, i.e. $r^* : \mathcal{S} \times \mathcal{A} \mapsto [0, R_{\text{max}}]$, and the process noise, captured by $p^*(s'|s, a)$, is i.i.d. Gaussian with variance σ^2.

After the n^{th} episode of environment interaction, we learn a model of the true dynamics, with a mean $\mu_n^d(s, a)$ and uncertainty estimate $\sigma_{\text{epi}_n}^d(s, a)$, and a model of the proxy reward with a mean $\mu_n^r(s, a)$ and uncertainty $\sigma_{\text{epi}_n}^r(s, a)$. Similar to Sukhija et al. (2025b), we model each dynamics coordinate $p_j^*,\ j \in \{1, ..., d_s\}$ using a Gaussian process (GP). The scalar reward r is treated analogously. For clarity, our discussion is limited to a setting with Gaussian noise, though this is not a requirement for the final regret bound. For settings with subgaussian process noise refer to Appendix D.4.

Assumption 5.2 (RKHS regularity)**.** The ground truth dynamics p^* and the proxy reward r reside in their own respective Reproducing Kernel Hilbert Spaces (RKHS). Each dynamics coordinate p_j^* lies in an RKHS $\mathcal{H}_{k_d}$ induced by kernel k_d with bounded norm $\|p_j^*\|_{\mathcal{H}_{k_d}} \leq B_d$, and k_d is uniformly bounded along the diagonal, i.e. $\sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} k_d\big((s, a), (s, a)\big) \leq \sigma_{\max}^d$. Furthermore, we assume $r \in \mathcal{H}_{k_r}$ with bounded norm $\|r\|_{\mathcal{H}_{k_r}} \leq B_r$ and with k_r uniformly bounded along the diagonal, i.e. $\sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} k_r\big((s, a), (s, a)\big) \leq \sigma_{\max}^r$.

Assumption 5.2, common in the Bayesian optimization and RL literature Srinivas et al. (2012); Curi et al. (2020); Sukhija et al. (2025b), allows modeling and learning the true dynamics and reward using GPs. This admits a closed form for the posterior mean and variance of the learned models, presented in Appendix D.1.

Given dynamics p we define the mean action-value function $\bar{Q}$ after the n^{th} episode of environment interaction to be

$$\bar{Q}_{p,n}^\pi(s, a) := \mathbb{E}_{\pi,p}\left[\sum_{t=0}^{\infty} \gamma^t\, \mu_n^r(s_t, a_t) \middle| s_0 = s, a_0 = a \right],$$

and denote our estimated action-value function as $\hat{Q}_n^\pi$. Following Sikchi et al. (2021), we have:

Assumption 5.3 (Q-function sub-optimality)**.** At the n^{th} episode of environment interaction, the estimation error of Q is bounded by a constant $\epsilon_{q,n}$, i.e., $\|\hat{Q}_n^\pi - \bar{Q}_{p,n}^\pi\|_\infty \leq \epsilon_{q,n}$.

To quantify the epistemic uncertainty of Q, we have the following assumption:

Assumption 5.4 (Epistemic Uncertainty of Q)**.** For every episode n, the epistemic uncertainty σ_n^q of the action–value function $\bar{Q}_n^\pi$ is proportional to the discounted cumulative epistemic uncertainty arising from the dynamics and reward models, i.e.,

$$\sigma_n^q \;\propto\; \mathbb{E}_{\pi,p}\left[\sum_{t=0}^{\infty}\gamma^t\,\|\sigma_n^d(s_t,a_t)\|\right] \;+\; \mathbb{E}_{\pi,p}\left[\sum_{t=0}^{\infty}\gamma^t\,\sigma_n^r(s_t,a_t)\right].$$

This assumption reflects that uncertainty in the action–value function accumulates from epistemic uncertainty in both the dynamics and reward models along trajectories induced by policy π. A detailed theoretical justification and discussion are provided in Appendix D.3.

For theoretical analysis, we provide the following definition of the UBP planner objective optimized by the planner at each planning step using fixed learned dynamics and reward models $\mu_n^{(\cdot)}$, $\sigma_{\text{epi}_n}^{(\cdot)}$:

Definition 5.5 (Optimistic planning objective)**.** Let π denote a candidate policy (i.e. the policy at the current planning iteration) and consider rollouts using the mean dynamics μ_n^d with the *same* process noise as the true environment. We define the UCB planning objective over planner horizon H:

$$\eta_{\gamma,n}^{\text{UCB}}(\pi,\mu_n^d) := \mathbb{E}_{\pi,\mu_n^d}\left[\sum_{t=0}^{H-1}\gamma^t\Big(\mu_n^r(\hat{s}_t,a_t)+a_n\sigma_n^r(\hat{s}_t,a_t)+b_n\sigma_n^d(\hat{s}_t,a_t)\Big)+\gamma^H\Big(\mu_n^q(\hat{s}_t,a_t)+c_n\sigma_n^q(\hat{s}_t,a_t)\Big)\right], \quad (7)$$

where a_n, b_n, c_n are parameters that control the strength of reward, dynamics exploration, with their theoretical bounds provided in Lemma 5.7 and justified in Appendix D.3. They correspond to hyperparameters $\lambda_r, \lambda_d, \lambda_q$ in Equation (1).

To derive a bound, we define the *cumulative infinite-horizon regret* for a particular policy π_n as

$$R_{\gamma,N} = \sum_{n=1}^{N} r_{\gamma,n} = \sum_{n=1}^{N} \eta_\gamma(\pi^*,p^*) - \eta_\gamma(\pi_n,p^*). \quad (8) \qquad \eta_\gamma(\pi,p) = \mathbb{E}_{\pi,p}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\right]. \quad (9)$$

We make the following assumption on the planner:

Assumption 5.6 (No planner sub-optimality)**.** The policy π_n^{ind} *induced* by the planning procedure through optimizing equation 7 is equal to the policy π_n^* that maximizes equation 7. That is, $\pi_n^{\text{ind}} = \pi_n^*$.

We refer to the policy as being *induced* by the planner because the planner does not train a parameterized policy. Rather, it repeatedly solves equation 7 and executes only the first action. Note that this is not the learned parametric policy $\pi_\varphi^{\text{learn}}$. For theoretical results that consider planner suboptimality, please refer to Appendix D.5. We now present the following lemma, which states:

Lemma 5.7 (Optimism of the UCB planning objective)**.** *Let Assumptions 5.1, 5.2, 5.3, and 5.4 hold, and let the learned GP dynamics and reward models be well-calibrated. Then there exist coefficients $a_n \in \mathcal{O}\big(\sqrt{\Gamma_{r,N}}\big)$, $b_n \in \mathcal{O}\big(\sqrt{\Gamma_{d,N}}\big)$ and $c_n \in \mathcal{O}\big(\max\{\sqrt{\Gamma_{r,N}}, \sqrt{\Gamma_{d,N}}\}\big)$ such that for all episodes n and all policies π, with probability at least $1-\delta$:*

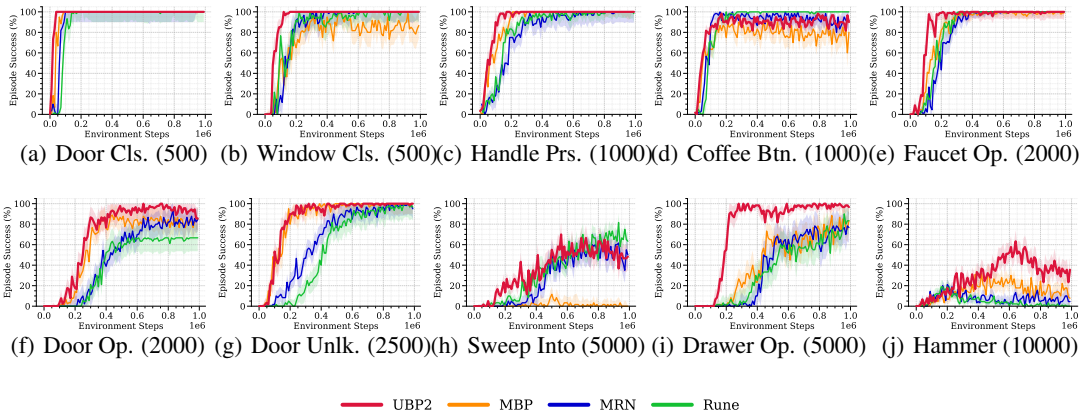
$$\eta_\gamma(\pi,p^*) \;\le\; \eta_{\gamma,n}^{\text{UCB}}(\pi,\mu_n^d) + \gamma^H\epsilon_{q,n}. \quad (10)$$

$\Gamma_{d,N}$ is the state and action dimension-dependent maximum information gain Srinivas et al. (2012); Sukhija et al. (2025b), defined in Appendix D.1, for a function in the RKHS induced by k_d over N episodes of environment interaction and similarly for $\Gamma_{r,N}$. Lemma 5.7 states that the planning objective upper-bounds the true return uniformly over policies up to some Q estimation error, i.e., the agent plans using an optimistic estimate of the environment. We first establish an analogous result for a reduced objective in Appendix D.2. The proof of Lemma 5.7 builds on this reduced case and is provided in Appendix D.3.

Theorem 5.8 (Infinite-horizon regret bound with dynamics, reward and value uncertainty bonuses)**.** *Let Assumptions 5.1, 5.2, 5.3, 5.4 and 5.6 hold, and let the learned GP dynamics and reward models be well-calibrated. The cumulative infinite-horizon regret after N episodes of real environment interaction satisfies, with probability $1-\delta$:*

$$R_{\gamma,N} \le \mathcal{O}\left(H^3\sqrt{N}\left(\Gamma_{d,N\log N}^{3/2}+\Gamma_{r,N\log N}^{3/2}\right)+e_{q,N}\right) \quad (11) \qquad \textit{where } e_{q,N} := \textstyle\sum_{n=1}^{N}\epsilon_{q,n}.$$

If we additionally assume that the growth rate of $e_{q,N}$ is sublinear, then Theorem 5.8 yields a sublinear regret rate governed by the maximum information gain of the dynamics and reward kernels, and parallels the infinite-horizon GP regret guarantees established for past work Sukhija et al. (2025b).



(a) Door Cls. (500) (b) Window Cls. (500) (c) Handle Prs. (1000) (d) Coffee Btn. (1000) (e) Faucet Op. (2000)

(f) Door Op. (2000) (g) Door Unlk. (2500) (h) Sweep Into (5000) (i) Drawer Op. (5000) (j) Hammer (10000)

Figure 3: Performance comparison across MetaWorld tasks. Curves show IQM performance with shaded standard error. Numbers between brackets indicate the maximum preference feedback budget.

6 EXPERIMENTS

We design our experiments to answer the following: **Q1:** Does uncertainty-guided optimistic planning improve sample efficiency in preference-based reinforcement learning compared to model-free and non-optimistic model-based baselines? **Q2:** How do uncertainty-aware planning components (dynamics, reward, and value), together with optimistic preference selection, individually affect success-rate performance? **Q3:** Can UBP2 benefit from varying the planning horizon? **Q4:** How does UBP2 perform under constrained preference feedback budgets? **Q5:** Does UBP2 remain effective with real human preference labels? Overall, we find that our approach is substantially more sample-efficient than strong model free approaches, and that all components of uncertainty are generally helpful to boosting sample efficiency and robustness.

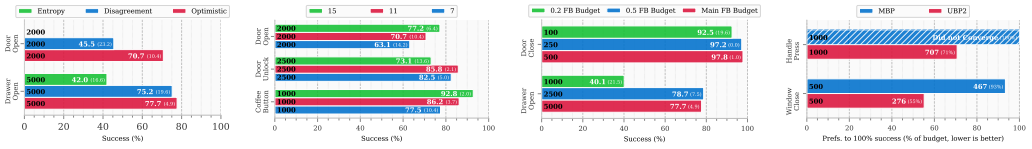
Table 1: Performance comparison on MetaWorld tasks over 1M environment steps. We report IQM success averaged over seeds and time, with standard error at the final evaluation step, along with average performance and average rank across tasks. Best results are shown in bold. Asterisks denote statistically significant improvements of UBP2 over the corresponding baseline under a Wilcoxon signed-rank test ($p < 0.05$)

Method	Door Cls.	Window Cls.	Handle Press	Coffee Btn.	Faucet Op.	Door Op.	Door Unlk.	Sweep In	Drawer Op.	Hammer	Avg ↑	Rank ↓
UBP2	**97.8 (1.0)**	**93.1 (1.0)**	**93.4 (0.0)**	86.2 (3.7)	**89.5 (7.9)**	**70.7 (10.4)**	**85.8 (2.1)**	**36.4 (12.3)**	**77.7 (4.9)**	**29.9 (11.5)**	**76.1**	**1.20**
MBP	95.6 (0.0)*	74.7 (13.5)*	90.8 (4.0)*	75.3 (10.7)	83.6 (5.0)*	61.5 (11.7)	85.2 (10.0)	1.1 (2.0)*	43.7 (12.8)*	14.8 (9.1)	62.6	2.60
Rune	91.5 (10.0)*	85.2 (9.2)*	84.5 (8.1)*	**90.4 (1.0)**	83.1 (1.4)*	41.7 (13.1)*	56.1 (13.1)*	35.5 (9.3)	35.9 (12.8)*	4.6 (0.9)*	60.8	3.10
MRN	94.2 (0.0)*	84.9 (10.2)*	81.5 (10.3)*	87.7 (11.0)	80.1 (0.0)*	48.2 (11.4)	65.6 (9.8)*	28.3 (14.5)	38.1 (14.9)*	7.5 (7.2)*	61.6	3.10
TDMPC2	98.8 (0.0)	98.7 (0.0)	98.6 (0.0)	98.4 (0.0)	95.9 (0.0)	96.1 (0.0)	97.7 (0.0)	94.1 (0.0)	95.5 (0.0)	94.5 (0.0)	96.8	–

Table 2: Ablation study on MetaWorld tasks over 1M environment steps. Config indicates whether Reward, Dynamics, Value Uncertainty, and Optimistic Preference Selection respectively are enabled (1) or disabled (0). Same metrics as Table 1. We also report worst-case regret (maximum per-task drop relative to the best variant)

Method	Config	Door Cls.	Window Cls.	Handle Press	Coffee Btn.	Faucet Op.	Door Op.	Door Unlk.	Sweep In	Drawer Op.	Hammer	Avg ↑	Regret ↓
UBP2	1111	**97.8 (1.0)**	93.1 (1.0)	93.4 (0.0)	86.2 (3.7)	89.5 (7.9)	70.7 (10.4)	85.8 (2.1)	36.4 (12.3)	**77.7 (4.9)**	29.9 (11.5)	**76.1**	**13.8**
	0111	96.7 (2.0)	**94.4 (0.0)**	91.6 (0.0)	**90.4 (9.7)**	82.8 (4.0)	**84.5 (2.4)**	85.0 (0.0)	**38.9 (16.4)**	68.3 (4.0)	22.4 (19.7)	75.5	15.8
	1001	94.8 (0.0)	93.1 (0.0)	91.2 (0.0)	82.8 (11.7)	**90.3 (0.0)**	67.7 (14.0)	78.4 (0.0)	20.3 (13.2)	44.0 (19.6)	14.1 (18.1)	67.7	33.7
	1011	95.9 (0.0)	94.2 (0.0)	92.0 (0.0)	84.2 (8.7)	89.7 (2.4)	58.5 (24.5)	83.3 (0.0)	29.3 (13.2)	58.4 (19.6)	12.4 (10.7)	69.8	26.0
	0101	97.5 (0.0)	93.6 (0.0)	91.0 (20.0)	76.8 (9.7)	85.6 (20.0)	82.4 (16.0)	81.1 (0.0)	31.4 (22.3)	53.9 (18.9)	15.1 (19.4)	70.8	23.8
	1101	96.4 (20.0)	94.2 (0.0)	92.1 (20.0)	90.1 (3.2)	84.7 (6.0)	78.8 (0.0)	**88.6 (0.0)**	4.0 (15.6)	76.0 (4.0)	**38.2 (22.6)**	74.3	34.9
	0011	97.3 (8.0)	94.0 (0.0)	**93.7 (0.0)**	89.8 (8.0)	80.0 (23.3)	66.9 (2.0)	83.6 (4.0)	15.0 (15.7)	71.8 (2.0)	33.3 (19.7)	72.5	23.9
	1110	96.0 (0.0)	89.8 (0.0)	88.8 (0.0)	79.5 (12.4)	87.7 (0.0)	68.9 (8.0)	76.0 (6.3)	11.2 (9.7)	68.2 (16.0)	27.4 (19.6)	69.3	27.7
MBP	0000	95.6 (0.0)	74.7 (13.5)	90.8 (4.0)	75.3 (10.7)	83.6 (5.0)	61.5 (11.7)	85.2 (10.0)	1.1 (2.0)	43.7 (12.8)	14.8 (9.1)	62.6	37.8

Experimental Setup. We evaluate UBP2 on 10 manipulation tasks of varying degrees of complexity from the MetaWorld benchmark Yu et al. (2021) using proprioceptive observations and task-specific success metrics Oquab et al. (2024). For our main results, we use feedback budgets commonly reported in the literature Liu et al. (2022); Liang et al. (2022); Lee et al. (2021). We compare UBP2 with the following baselines: **(1) RUNE** Liang et al. (2022): A model-free, PbRL method that enhances exploration by augmenting the extrinsic reward with an intrinsic bonus based on



(a) Pref. Selection Method (b) Planning Horizon (c) Reduced Feedback (d) Human Annotation

Figure 4: Sensitivity analysis of UBP2 across three key hyperparameters on MetaWorld tasks. **(a)** Pref. selection strategy. **(b)** Planning horizon. **(c)** Feedback budget. IQM success averaged over seeds and evaluation steps is reported in white, with standard error at the final evaluation step in brackets; feedback budget is reported in black on the left of each bar.In **(a)**, entropy-based preference selection achieves the lowest average IQM of 0.1% (20) on Door Open. In **(d)**, number of preferences needed to reach 100% success, as a percentage of the feedback budget is reported for UBP2 and MBP. The hatched bar did not converge within the budget.

disagreement, **(2) MRN (Meta-Reward-Net)** Liu et al. (2022): A model-free, PbRL approach that employs bi-level optimization to learn the reward function based on the performance of the Q-function. **(3) MBP** (Model-Based PbRL): A non-optimistic variant of UBP2 that removes uncertainty-aware planning and optimistic preference selection. This baseline isolates the effect of MBRL without optimism or uncertainty-driven exploration. This variant is conceptually similar to MoP-RL Liu et al. (2023). **(4) TDMPC2** We additionally report Hansen et al. (2024b) as an upper bound baseline trained using dense true rewards **Evaluation Metric**: We report Interquartile Mean (IQM) success rate over environment steps where faster gains indicate higher sample efficiency in preference-based learning. We report results using the IQM in our plots and tables, as a robust aggregate metric for reinforcement learning evaluation Agarwal et al. (2022). IQM reduces the influence of outlier runs and provides a more stable estimate compared to the mean.

Sample Efficiency in PbRL: Figure 3 shows that UBP2 reaches high success rates earlier in 9 out of the 10 tasks, demonstrating the benefit of optimistic planning in PbRL. Table 1 shows UBP2 achieves both the highest overall average performance and the best average rank across tasks.

Component Ablations: We evaluate the contribution of each component in UBP2 through an ablation study against the full model and MBP, where all optimistic components are disabled. Table 2 shows that UBP2 provides the best balance between performance and robustness. Although some ablations slightly improve individual tasks, these gains are inconsistent and incur higher regret elsewhere. Results highlight the importance of each optimistic component for strong consistent performance.

Optimistic Preference Selection We compare UBP2 against preference-query heuristics discussed in prior work: disagreement-based and entropy-based selection. Figure 4(a) shows that optimistic preference selection consistently outperforms both alternatives in challenging manipulation tasks

Varying Horizon We evaluate the effect of the planning horizon in Figure 4(b), a key degree of freedom in model-based planning enabled by UBP2. Unlike model free methods, UBP2 allows adapting the planning horizon to better match task structure. Because the optimal horizon varies across tasks, we tune it separately for each environment. Across the three representative tasks shown, increasing the horizon from 7 to 11 consistently improves performance. Although even longer horizons may provide additional gains, they come at a higher computational cost during planning.

Reduced Feedback We evaluate UBP2 under constrained preference feedback budget, 2 tasks receive a limited feedback budget. The results in Figure 4(c) indicate that UBP2 retains its effectiveness even when the available feedback is significantly reduced.

Human Preference Feedback: We evaluate UBP2 using real human annotations for preferences against MBP. The results in Figure 4(d) show that the UBP2 remains superior to MBP under human preferences, indicating that UBP2's queries selection is easier for human labeling.

7 Conclusion

This work presents Uncertainty-Balanced Preference Planning (UBP2), an MBRL preference approach that actively guides exploration. By planning under a unified optimistic objective that balances expected return with epistemic uncertainty, UBP2 explicitly balances exploitation and information acquisition, avoiding uninformative data collection. We acknowledge computational overhead as a limitation due to ensembles and planning. For more discussion please refer to Appendix B.2

AI USE STATEMENT

In this work, we have not used generative AI tools for any of the tasks with required disclosure; all such tasks were either carried out without AI assistance or are not applicable to this work. We used generative AI tools for tasks with recommended disclosure, specifically to edit the paper for grammar and readability, and to edit the code used to create scientific figures (plotting code). We have reviewed all AI-assisted work. All AI-suggested grammatical edits were read and approved by the authors to ensure that the meaning and technical content of the text were unchanged, and all AI-edited plotting code was checked by the authors to verify that the resulting figures accurately reflect the underlying experimental data. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This research adheres to the ICLR Code of Ethics. All experiments are conducted on publicly available benchmark datasets and environments that do not involve human subjects or personally identifiable information. No new data collection was performed, and no sensitive or private information is included. The contributions of this work are methodological, aiming to advance machine learning. While reinforcement learning and related methods may be applied in safety-critical or socially sensitive domains, this paper does not directly address or deploy in such contexts. We have taken care to report results honestly, acknowledge limitations, and follow best practices for research integrity. No conflicts of interest or ethical concerns beyond standard research conduct arise from this work.

REPRODUCIBILITY STATEMENT

We have taken several steps to ensure the reproducibility of our results. The full source code for our method, baselines, and experiments is publicly released at `https://anonymous.4open.science/r/ubp2-B2A1`, including scripts to reproduce our reported results and figures. Section 4.3 describes the proposed method in detail, and Appendix B.1 lists the complete set of hyperparameters used for each task, along with implementation details.

REFERENCES

David Abel, Will Dabney, Anna Harutyunyan, Mark K. Ho, Michael L. Littman, Doina Precup, and Satinder Singh. On the expressivity of markov reward, 2022. URL https://arxiv.org/abs/2111.00876.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice, 2022. URL https://arxiv.org/abs/2108.13264.

Arthur Argenson and Gabriel Dulac-Arnold. Model-based offline planning, 2021. URL https://arxiv.org/abs/2008.05556.

Arunkumar Byravan, Leonard Hasenclever, Piotr Trochim, Mehdi Mirza, Alessandro Davide Ialongo, Yuval Tassa, Jost Tobias Springenberg, Abbas Abdolmaleki, Nicolas Heess, Josh Merel, and Martin Riedmiller. Evaluating model-based planning and planner amortization for continuous control, 2021. URL https://arxiv.org/abs/2110.03363.

Sayak Ray Chowdhury and Aditya Gopalan. On kernelized multi-armed bandits, 2017. URL https://arxiv.org/abs/1704.00445.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Sebastian Curi, Felix Berkenkamp, and Andreas Krause. Efficient model-based reinforcement learning through optimistic policy search and planning, 2020. URL https://arxiv.org/abs/2006.08684.

Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 2005. URL https://people.smp.uq.edu.au/DirkKroese/ps/aortut.pdf.

Yunhai Feng, Nicklas Hansen, Ziyan Xiong, Chandramouli Rajagopalan, and Xiaolong Wang. Finetuning offline world models in the real world, 2023. URL https://arxiv.org/abs/2310.16029.

Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming, 2013. URL https://arxiv.org/abs/1309.5549.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=Oxh5CstDJU.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control, 2024b. URL https://arxiv.org/abs/2310.16828.

Joey Hejna and Dorsa Sadigh. Few-shot preference learning for human-in-the-loop rl, 2022. URL https://arxiv.org/abs/2212.03363.

David Janz, David R. Burt, and Javier González. Bandit optimisation of functions in the matérn kernel rkhs, 2023. URL https://arxiv.org/abs/2001.10396.

Jihwan Jeong, Xiaoyu Wang, Jingmin Wang, Scott Sanner, and Pascal Poupart. Reflect-then-plan: Offline model-based planning through a doubly bayesian lens, 2025. URL https://arxiv.org/abs/2506.06261.

Sham Kakade, Akshay Krishnamurthy, Kendall Lowrey, Motoya Ohnishi, and Wen Sun. Information theoretic regret bounds for online nonlinear control, 2020. URL https://arxiv.org/abs/2006.12466.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel : Model-based offline reinforcement learning, 2021. URL https://arxiv.org/abs/2005.05951.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. URL https://openreview.net/forum?id=ps95-mkHF_.

Xinran Liang, Katherine Shu, Kimin Lee, and Pieter Abbeel. Reward uncertainty for exploration in preference-based reinforcement learning, 2022. URL https://arxiv.org/abs/2205.12401.

Runze Liu, Fengshuo Bai, Yali Du, and Yaodong Yang. Meta-reward-net: Implicitly differentiable reward learning for preference-based reinforcement learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=OZKBReUF-wX.

Yi Liu, Gaurav Datta, Ellen Novoseller, and Daniel S Brown. Efficient preference-based reinforcement learning using learned dynamics models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2921–2928. IEEE, 2023.

Katherine Metcalf, Miguel Sarabia, and Barry-John Theobald. Rewards encoding environment dynamics improves preference-based reinforcement learning. *arXiv preprint arXiv:2211.06527*, 2022.

Thomas M. Moerland, Joost Broekens, Aske Plaat, and Catholijn M. Jonker. Model-based reinforcement learning: A survey, 2022. URL https://arxiv.org/abs/2006.16712.

Anusha Nagabandi, Gregory Kahn, Ronald S. Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning, 2017. URL https://arxiv.org/abs/1708.02596.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. URL https://arxiv.org/abs/2304.07193.

A. R'enyi. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, volume 4, pp. 547–562. University of California Press, 1961.

Jonas Rothfuss, Bhavya Sukhija, Tobias Birchler, Parnian Kassraie, and Andreas Krause. Hallucinated adversarial control for conservative offline policy evaluation, 2023. URL https://arxiv.org/abs/2303.01076.

Rajiv Sambharya, Georgina Hall, Brandon Amos, and Bartolomeo Stellato. End-to-end learning to warm-start for real-time quadratic optimization, 2022. URL https://arxiv.org/abs/2212.08260.

Junwon Seo, Kensuke Nakamura, and Andrea Bajcsy. Uncertainty-aware latent safety filters for avoiding out-of-distribution failures, 2025. URL https://arxiv.org/abs/2505.00779.

Harshit Sikchi, Wenxuan Zhou, and David Held. Learning off-policy with online planning, 2021. URL https://arxiv.org/abs/2008.10066.

Satinder Singh, Richard L. Lewis, and Andrew G. Barto. Where do rewards come from? In *Proceedings of the Annual Conference of the Cognitive Science Society*, pp. 2601–2606. Cognitive Science Society, 2009.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012. doi: 10.1109/TIT.2011.2182033.

Bhavya Sukhija, Lenart Treven, Cansu Sancaktar, Sebastian Blaes, Stelian Coros, and Andreas Krause. Optimistic active exploration of dynamical systems, 2023. URL https://arxiv.org/abs/2306.12371.

Bhavya Sukhija, Stelian Coros, Andreas Krause, Pieter Abbeel, and Carmelo Sferrazza. Maxinforl: Boosting exploration in reinforcement learning through information gain maximization, 2025a. URL https://arxiv.org/abs/2412.12098.

Bhavya Sukhija, Lenart Treven, Carmelo Sferrazza, Florian Dörfler, Pieter Abbeel, and Andreas Krause. Sombrl: Scalable and optimistic model-based rl, 2025b. URL https://arxiv.org/abs/2511.20066.

Scott Sussex, Anastasiia Makarova, and Andreas Krause. Model-based causal bayesian optimization, 2023. URL https://arxiv.org/abs/2211.10257.

Christian Wirth, Riad Akrour, Gerhard Neumann, and Johannes Fürnkranz. A survey of preference-based reinforcement learning methods. *Journal of Machine Learning Research*, 18(136):1–46, 2017.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization, 2020. URL https://arxiv.org/abs/2005.13239.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Avnish Narayan, Hayden Shively, Adithya Bellathur, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning, 2021. URL https://arxiv.org/abs/1910.10897.

Wenhao Zhan, Masatoshi Uehara, Wen Sun, and Jason D. Lee. Provable reward-agnostic preference-based reinforcement learning, 2024. URL https://arxiv.org/abs/2305.18505.

A ALGORITHMS

Algorithm 2 Optimistic Preference Pair Selection

Input: replay buffer $\mathcal{B}$, reward ensemble $\{r_\theta^{(m)}\}_{m=1}^{E}$, pref. buffer $\mathcal{P}$
Hyperparameters: segment length L, pairs to add K
Candidate set $\mathcal{C} \leftarrow \emptyset$
for each batch **do**
  Sample candidate segment pairs (τ_1, τ_2) (length L) from $\mathcal{B}$
  for each (τ_1, τ_2) **do**
    $R_i^{(m)} \leftarrow \sum_{t=0}^{L-1} r_\theta^{(m)}(s_t^i, a_t^i),\ i \in \{1,2\},\ m \in [E]$
    $p^{(m)} \leftarrow \mathrm{softmax}([R_1^{(m)}, R_2^{(m)}])_1$
    $\mathrm{score} \leftarrow \mu(p^{(1:E)}) + \sigma(p^{(1:E)})$
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{(\tau_1, \tau_2, \mathrm{score})\}$
  end for
end for
Add top-K pairs in $\mathcal{C}$ by score to $\mathcal{P}$

UBP2 selects the most informative preference pairs for labeling by scoring each candidate segment pair using both predicted reward and reward-model epistemic uncertainty, then globally ranking and selecting the top-K pairs to add to the preference buffer $\mathcal{P}$.

B Implementation Details

B.1 Hyper-parameters

UBP2 learns a reward model from preference feedback using short trajectory segments. Preferences are periodically collected and the reward model is updated in batches.

Table 3: Preference Learning Hyperparameters

Parameter	Door Close	Window Close	Handle Press	Coffee Button	Faucet Open	Door Open	Door Unlock	Sweep Into	Drawer Open	Hammer
number of seeds	10	10	10	10	15	15	10	10	10	15
pref_segment_length	10	10	10	10	10	10	10	10	10	10
reward_train_horizon	10	10	10	10	10	10	10	10	10	10
num_pref_per_addition	12	12	12	25	12	12	12	25	25	50
max_pref_feedback	500	500	1000	1000	2000	2000	2500	5000	5000	10000
pref_update_freq	2500	2500	2500	2500	2500	2500	2500	2500	2500	2500
reward_batch_size	50	50	50	50	50	50	50	100	100	200

We use Model Predictive Control (MPC) with the Cross-Entropy Method (CEM) for planning. At each timestep, candidate trajectories are sampled and iteratively refined using elite selection

Table 4: MPC (CEM) Planning Hyperparameters

Parameter	Door Close	Window Close	Handle Press	Coffee Button	Faucet Open	Door Open	Door Unlock	Sweep Into	Drawer Open	Hammer
iterations	6	6	6	6	6	6	6	6	6	6
number of samples	512	512	512	512	512	512	512	512	512	512
num of elites	64	64	64	64	64	64	64	64	64	64
num of policy trajectories	24	24	24	24	24	24	24	24	24	24
horizon	7	7	7	11	7	11	11	7	7	7

B.2 Compute Resources

All experiments were conducted in a shared high-performance computing cluster. We used GPU-accelerated nodes for training and evaluation. Specifically, we utilized A100 or H100 GPU partitions depending on availability. Table 5 summarizes the hardware used in our experiments.

Table 5: Compute resources used in all experiments.

Resource	Specification	Quantity
GPU (A100)	5GB MIG partition	1
GPU (H100)	10GB MIG partition (HBM3)	1

Wall-clock overhead. Table 6 reports wall-clock training time for UBP2 and the model-free baselines on three representative tasks spanning low to high preference budgets. UBP2 is slower than RUNE and MRN. We attribute most of this cost to two sources: (i) the CEM-based MPC loop, which rolls out the learned dynamics over a horizon of $H \in \{7, 11\}$ steps, compared to $H = 3$ in TDMPC2 (Hansen et al., 2024b); and (ii) evaluating the reward, dynamics, and value ensembles, and their uncertainty estimates, for every candidate trajectory at every planning step. This overhead yields the largest gains on harder, higher-budget tasks: UBP2 improves IQM success by 41.8 points over RUNE on Drawer Open and by 29.0 points on Door Open. Reducing planning cost, for example through fewer CEM iterations and parallelizing environment steps are important directions for future work.

Table 6: Wall-clock training time over 1M environment steps

Task	Method	Pref. budget	Horizon H	Time (h)	IQM ↑
Door Close	UBP2	500	7	30.0	97.8
	RUNE		7	2.4	91.5
	MRN		7	2.3	94.2
Drawer Open	UBP2	5000	7	39.5	77.7
	RUNE		7	2.4	35.9
	MRN		7	2.3	38.1
Door Open	UBP2	2000	11	47.0	70.7
	RUNE		11	2.3	41.7
	MRN		11	2.4	48.2

C Additional Experiments

C.1 Conservative Evaluation

We evaluate whether the uncertainty estimates learned by UBP2 can be used conservatively at test time by reversing the optimism term during evaluation, i.e., penalizing trajectories with high epistemic uncertainty instead of rewarding them. This experiment is motivated by the observation that optimism is beneficial during data collection and learning, where exploration is required, but may become undesirable during deployment if a risk-averse or safety-oriented policy is preferred. Importantly, optimism is not used during standard evaluation in UBP2; the learned policy is evaluated without any exploration bonus. Here, we instead explicitly introduce a conservative evaluation objective by multiplying the uncertainty bonus by -1 during planning. Results shows that conservative evaluation substantially degrades performance on most tasks We speculate that uncertainty in UBP2 primarily acts as a useful exploratory signal rather than a reliable proxy for risk, and that naively enforcing conservatism at test time can significantly reduce task performance.

Table 7: Performance comparison across MetaWorld tasks for UBP2 with conservative evaluation. Entries report average IQM success with standard error at the final evaluation step. Best results are shown in bold

Method	**Coffee Button**	**Faucet Open**	**Sweep Into**	**Drawer Open**
UBP2	**86.2** (3.7)	**89.5** (7.9)	**36.4** (12.3)	**77.7** (4.9)
UBP2 w conservative eval	51.0 (22.3)	85.9 (20.0)	6.4 (0.0)	30.8 (23.7)

C.2 Varying Horizon

In the ablation study, we compare planning horizons of 7 and 11 across tasks. While horizon 11 yields better performance on a subset of tasks reported in the experiments section, it does not consistently improve results on the tasks considered in this appendix. In contrast, a horizon of 7 provides more stable and higher overall performance for these tasks. Therefore, we adopt horizon 7 for the results reported in this section.

Table 8: Effect of planning horizon on UBP2 performance across a subset of MetaWorld tasks over 1M steps. Entries report IQM success with standard error at the final evaluation step.

Horizon	Door Close	Window Close	Handle Press	Sweep Into	Drawer Open
7	**97.8** (1.0)	**93.1** (1.0)	**93.4** (0.0)	**36.4** (12.3)	**77.7** (4.9)
11	96.3 (0.0)	91.6 (0.0)	86.1 (8.0)	15.0 (7.5)	71.5 (19.6)

C.3 Noisy Labels

We further extend our experiments to study the impact of labeling noise on the performance of UBP2 compared to baselines, on the Window Close and Handle Press tasks. We find that UBP2 remains competitive with baselines under synthetic label noise. This supports UBP2's practical robustness to real-world label imperfections.

Table 9: Effect of labeling noise on UBP2 and baselines. We report the interquartile mean (IQM) success rate (%) after 1M environment steps, with standard error at the final evaluation step in parentheses, using noise-free labels and labels where 10% of preferences are flipped. The last column shows the relative drop in IQM success caused by noise. Best result per task and column is in bold.

Task	Pref. Budget	Approach	IQM Success (No Noise)	IQM Success (10% Noise)	Drop due to Noise (%)
Window Close	500	UBP2	**92.8** (0.0)	**42.4** (24.5)	**54.3**
		MBP	74.4 (15.0)	5.1 (19.6)	93.1
		RUNE	84.8 (10.3)	26.7 (24.5)	68.5
		MRN	84.4 (9.5)	35.4 (22.4)	58.1
Handle Press	1000	UBP2	**93.4** (0.0)	68.3 (7.5)	26.9
		MBP	90.7 (5.0)	59.5 (20.0)	34.4
		RUNE	84.5 (8.1)	**69.2** (15.6)	**18.1**
		MRN	81.4 (5.7)	58.7 (16.3)	27.9

C.4 Autotuning Approach Ablation

We ablate auto-tuning from Sukhija et al. (2025a) to isolate the effect of each coefficient individually, fixing it to 1 while the others are still auto-tuned, as well as a variant that disables auto-tuning for all coefficients, on the Drawer Open task. We find that auto-tuning all coefficients works well empirically without requiring hand-designed heuristics, such as manually decaying the uncertainty weights over training.

Table 10: Ablation of uncertainty-weight auto-tuning on Drawer Open. We report the IQM success rate (%) averaged over 1M environment steps across 5 seeds, with standard error at the final evaluation step in parentheses. λ_r, λ_d, and λ_v are the weights on reward, dynamics, and value uncertainty; fixing a weight to 1 disables auto-tuning for that weight only.

Task	Ablation	Avg. IQM Success over 1M Steps
Drawer Open	Default, all auto-tuned	**77.7** (4.9)
	$\lambda_r = 1$	62.0 (18.8)
	$\lambda_d = 1$	70.9 (20.0)
	$\lambda_v = 1$	50.3 (19.6)
	All fixed to 1, no auto-tuning	73.5 (19.6)

D PROOFS

D.1 DEFINITIONS AND CLARIFICATIONS

We first give some definitions and characterizations of our learned models. Later, we clarify some of the choices we have made in regard to the theory.

Closed forms of the mean and variance of GPs Let $K_n \in \mathbb{R}^{M\times M}$ denote the kernel matrix over $\mathcal{D}_n$, $k_n(s,a) \in \mathbb{R}^M$ denote the vector of kernel evaluations between (s,a) and all inputs in $\mathcal{D}_n$, and $s'^j_{1:M}$ denote the noisy measurements of p^*_j. For GP regression of the true dynamics with observation noise variance σ^2, the posterior mean and variance for coordinate j have the following standard closed forms Sukhija et al. (2025b):

$$\mu^d_{n,j}(s,a) = k_n(s,a)^\top \big(K_n + \sigma^2 I\big)^{-1} s'^j_{1:M}, \tag{12}$$

$$\big(\sigma^d_{\text{epi}_{n,j}}(s,a)\big)^2 = k((s,a),(s,a)) - k_n(s,a)^\top \big(K_n + \sigma^2 I\big)^{-1} k_n(s,a). \tag{13}$$

We collect these coordinate-wise values into vector estimates $\mu^d_n(s,a) = [\mu^d_{n,j}(s,a)]_{j\le d_s}$ and $\sigma^d_{\text{epi}_n}(s,a) = [\sigma^d_{\text{epi}_{n,j}}(s,a)]_{j\le d_s}$. We similarly fit a GP model using equation 13 on the reward with posterior mean $\mu^r_n(s,a)$ and standard deviation $\sigma^r_{\text{epi}_n}(s,a)$ using the reward kernel k_r. Furthermore, we say that these learned models are *well-calibrated* Rothfuss et al. (2023); Sukhija et al. (2025b).

Definition of well-calibrated models To connect epistemic uncertainty to optimism, in section 5 we assumed the learned models are *well-calibrated*: their posterior standard deviations provide valid confidence radii around the true functions they are attempting to estimate. Concretely, we give the following definition.

Definition D.1 (Well-calibrated learned dynamics and reward Rothfuss et al. (2023))**.** Simultaneously for all episodes n and all $(s,a) \in \mathcal{S}\times\mathcal{A}$, with probability $1-\delta$:

$$\big|r(s,a) - \mu^r_n(s,a)\big| \le \beta^r_n(\delta)\, \sigma^r_{\text{epi}_n}(s,a), \tag{14}$$

$$\forall j \le d_s: \qquad \big|p^*_j(s,a) - \mu^d_n(s,a)\big| \le \beta^d_n(\delta)\, \sigma^d_{\text{epi}_{n,j}}(s,a), \tag{15}$$

where $\beta^r_n(\delta), \beta^d_n(\delta) \in \mathbb{R}_{\ge 0}$ are non-decreasing sequences. Moreover, in the GP/RKHS setting, typically $\beta^r_n \in \mathcal{O}\big(\sqrt{\Gamma_{r,n}}\big)$ and $\beta^d_n \in \mathcal{O}\big(\sqrt{\Gamma_{d,n}}\big)$ Chowdhury & Gopalan (2017).

We invoke a lemma in Rothfuss et al. (2023) to show that GP estimates of the dynamics and reward are well-calibrated models if Assumption 5.2 holds:

Lemma D.2 (Well-calibrated confidence intervals for RKHS Rothfuss et al. (2023))**.** *Let p^* be in a RKHS with kernel k_d and the characteristics in Assumption 5.2. Suppose μ^d_n and $\sigma^d_{\text{epi}_n}$ are the posterior mean and variance of a GP with kernel k_d as defined in equation equation 13. Then, there exists $\beta^d_n(\delta)$ for which the tuple $(\mu^d_n, \sigma^d_{\text{epi}_n}, \beta^d_n(\delta))$ is a well-calibrated model of p^*. The same is true for the proxy reward function $r(s,a)$.*

Proof. Lemma D.2 is directly stated in Rothfuss et al. (2023) for an RKHS of vector-valued, scalar-output functions. Hence, this can directly be applied to the reward model and element-wise for the dynamics model. □

Definition of the maximum information gain For a kernel k and noise variance σ^2, the maximum information gain is defined to be:

$$\Gamma_M(k) := \max_{A\subset\mathcal{S}\times\mathcal{A},\ |A|\le M} \frac{1}{2}\log\det\big(I + \sigma^{-2}K_A\big), \tag{16}$$

where K_A is the kernel matrix over the set A. Intuitively, $\Gamma_M(k)$ measures the maximum amount of information that M noisy function evaluations can reveal about a function in the RKHS class parameterized by k. This quantity is sublinear in M for many common kernels Sukhija et al. (2025b). See Table 11 for a list of the asymptotic growth of Γ_M for different kernels.

Further discussion on modeling the true reward r^* in preference-based RL. As discussed in section 3.1 and the Limitations, in PbRL, the agent never observes the real scalar rewards, and is instead trained on preference labels, giving rise to the definition of the proxy reward function $r(s, a)$ within the same equivalence class of $r^*(s, a)$. This "latent-reward" view is standard in the PbRL literature, where preferences provide information about an underlying reward signal and hence allows the learning of a reward model whose ordering of information matches the ground-truth ordering Wirth et al. (2017); Christiano et al. (2017). We assume that the well-calibrated GP reward model effectively approximates the proxy reward function $r(s, a)$, satisfying equation 14. This view has enabled us to effectively produce sublinear regret bounds for the problem setting presented in this paper.

However, as we have acknowledged earlier, it may be possible to make the reward learning error stemming from PbRL more explicit. Namely, Zhan et al. (2024) establishes value approximation guarantees under a linear reward parameterization, where the preference learning error can be more clearly characterized and directly incorporated into the final bound. In contrast, our setting does not assume such a parametric structure on the reward. As a result, explicitly propagating preference-learning error into the regret bound would require introducing additional assumptions on the reward model and its convergence rate. Introducing assumptions like a linear reward model could incorporate reward convergence into Lemma D.4 and Lemma 5.7, but this would complicate the analysis while introducing harsher assumptions. Our current result isolates planning and provides guarantees under general reward uncertainty, avoiding strong reward model conditions.

Clarification between implemented dynamics and reward estimators and theoretical assumptions As introduced in the Limitations, practically the dynamics and reward are implemented via deep ensemble networks, while in the theoretical analysis they are assumed to be GPs. We acknowledge that this may be a strict assumption, and that our actual implementation may not fully reflect the theory.

However, we note that in practice, deep ensembles are ubiquitous in model-based RL to quantify epistemic uncertainty or model disagreement. Furthermore, similar approximations have been adopted in prior work, such as Sukhija et al. (2025b), providing a practical starting point for theoretically characterizing performance. Our intent is not to claim that the implemented ensemble estimator is an exact GP posterior. Our proof requires two properties: 1) a high-probability confidence relationship between the reward/dynamics model error and their uncertainty radii, and 2) a cumulative uncertainty bound on these uncertainty radii (in our case, via GP information gain bounds). GP posterior standard deviations provide these quantities in closed form, which is why they are used in Lemma 5.7 and Theorem 5.8.

Hence, both our implementation and theory aim to capture the same underlying notion of posterior-predictive uncertainty, but using different model classes and approximations. In the implementation, ensemble disagreement and JRD are used as sample-based approximations to the spread of the posterior predictive distribution of the reward and dynamics. We view the ensemble members as approximate samples from a posterior or posterior-like distribution over possible reward and dynamics models. Then, the ensemble mean/disagreement can be viewed as Monte Carlo estimates of the mean/variance of the approximate posterior predictive distribution, and the JRD is a divergence-based measure of disagreement among each ensemble member's predictive distributions. Therefore, both our implementation and theory use predictive uncertainty as a measurement of confidence, but only the GP case yields an explicit, clean, closed form regret bound.

Table 11: Maximum information gain bounds for common choice of kernels Sukhija et al. (2025b).

Kernel $k(s, s')$	Γ_N
Linear: $s^\top s'$	$\mathcal{O}\big(d \log N\big)$
RBF: $\exp\Big(-\frac{\|s-s'\|^2}{2l^2}\Big)$	$\mathcal{O}\big(\log^{d+1} N\big)$
Matérn: $\frac{1}{\Gamma(\nu)\,2^{\nu-1}}\Big(\frac{\sqrt{2\nu}\,\|s-s'\|}{l}\Big)^{\nu} B_\nu\Big(\frac{\sqrt{2\nu}\,\|s-s'\|}{l}\Big)$	$\mathcal{O}\Big(N^{\frac{d}{2\nu+d}}\ \log^{\frac{2\nu}{2\nu+d}}(N)\Big)$

We acknowledge, however, that finite-neural ensembles are not guaranteed to be as well-calibrated as GPs, potentially presenting challenges reconciling the derived regret bounds with the practical results.

D.2 Analysis of Discounted Finite Horizon Planning with UCB Objective

Finite-horizon objective and regret Prior to showing the theoretical justification for the full infinite-horizon regret bound and the full UCB objective, we first consider a reduced UCB objective. Define:

$$\eta_n^{\mathrm{UCB}}(\pi, \mu_n^d) := \mathbb{E}_{\pi,\mu_n^d}\left[\sum_{t=0}^{H-1} \gamma^t \Big(\mu_n^r(\hat{s}_t, a_t) + a_n \sigma_n^r(\hat{s}_t, a_t) + b_n \sigma_n^d(\hat{s}_t, a_t)\Big)\right]. \tag{17}$$

This objective corresponds to a finite-horizon variant of the full UCB objective in equation 7, with a_n, b_n defined as in Lemma 5.7. Since the objective is finite-horizon, we omit the terminal value estimation and its uncertainty bonus. In this section, we derive a finite-horizon regret bound for policies obtained by optimizing equation 17 and how that it is sublinear. This informs our infinite horizon regret bound for the full UCB planning objective.

Analogous to equation equation 8, we introduce a performance metric for the finite-horizon setting. We define the *cumulative finite-horizon regret* for a particular policy π_n as:

$$R_N = \sum_{n=1}^{N} r_n = \sum_{n=1}^{N} \eta(\pi^*, p^*) - \eta(\pi_n, p^*), \tag{18}$$

where $\eta(\cdot\,,\cdot)$ is the finite-horizon analogue of equation 9

$$\eta(\pi, p) = \mathbb{E}_{\pi,p}\left[\sum_{t=0}^{H-1} \gamma^t r(s_t, a_t)\right]. \tag{19}$$

Discounted Finite-Horizon Cumulative Uncertainty and Return We further define the following quantities. For real episode n and policy π, define (as in Sukhija et al. (2025b)):

$$\widetilde{\Sigma}_n^r(\pi) := \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \sigma^r_{\mathrm{epi_n}}(\hat{s}_t, a_t)\Big], \tag{20}$$

$$\Sigma_n^r(\pi) := \mathbb{E}_{\pi,p^*}\Big[\sum_{t=0}^{H-1} \gamma^t \sigma^r_{\mathrm{epi_n}}(s_t, a_t)\Big], \tag{21}$$

$$\widetilde{\Sigma}_n^d(\pi) := \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \left\|\sigma^d_{\mathrm{epi_{n,j}}}(\hat{s}_t, a_t)\right\|\Big], \tag{22}$$

$$\Sigma_n^d(\pi) := \mathbb{E}_{\pi,p^*}\Big[\sum_{t=0}^{H-1} \gamma^t \left\|\sigma^d_{\mathrm{epi_{n,j}}}(s_t, a_t)\right\|\Big]. \tag{23}$$

These quantities represent the sum of the estimated uncertainties of the learned reward and dynamics across timesteps under policy rollouts in the true and mean dynamics. Definitions equation 20 to equation 23 will be useful later proofs, and will be generalized to the infinite horizon setting.

Lemma D.3 (Upper Bound of Discounted Return)**.** *Consider the equations equation 19, equation 22, and equation 23 with the true dynamics p^* and mean dynamics μ_n^d having the same process noise. Then, there is a $\lambda_n \in \mathcal{O}(\sqrt{\Gamma_{d,N}})$ such that for any policy π:*

$$|\eta(\pi, p^*) - \eta(\pi, \mu_n^d)| \le \lambda_n \Sigma_n^d(\pi). \tag{24}$$

$$|\eta(\pi, p^*) - \eta(\pi, \mu_n^d)| \le \lambda_n \tilde{\Sigma}_n^d(\pi). \tag{25}$$

Proof. Lemma D.3 is simply a discounted variation of Lemma B.1. from Sukhija et al. (2025b). It can be seen that adding the discount factor has no effect on the proof. □

Lemma D.4 (Optimism of the UCB planning objective)**.** *Let Assumptions 5.1, 5.2 hold, and let the learned GP dynamics and reward models be well-calibrated. Then there exist coefficients $a_n \in \mathcal{O}\big(\sqrt{\Gamma_{r,N}}\big)$ and $b_n \in \mathcal{O}\big(\sqrt{\Gamma_{d,N}}\big)$ such that for all episodes n and all policies π, with probability at least $1-\delta$:*

$$\eta(\pi, p^*) \;\leq\; \eta_n^{\mathrm{UCB}}(\pi, \mu_n^d). \tag{26}$$

Proof. From Lemma D.3, we have:

$$\begin{aligned}
\eta(\pi, p^*) &\leq \lambda_n \tilde{\Sigma}_n^d(\pi) + \eta(\pi, \mu_n^d) \\
&= \lambda_n \tilde{\Sigma}_n^d(\pi) + \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t r(\hat{s}_t, a_t)\Big] \\
&= \lambda_n \tilde{\Sigma}_n^d(\pi) + \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \big(r(\hat{s}_t, a_t) - \mu_n^r(\hat{s}_t, a_t)\big)\Big] + \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \mu_n^r(\hat{s}_t, a_t)\Big] \\
&\leq \lambda_n \tilde{\Sigma}_n^d(\pi) + \underbrace{\mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \big|r^*(\hat{s}_t, a_t) - \mu_n^r(\hat{s}_t, a_t)\big|\Big]}_{\text{(A)}} + \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \mu_n^r(\hat{s}_t, a_t)\Big].
\end{aligned}$$

We bound (A) using Definition D.1, arriving at the optimism lemma:

$$\begin{aligned}
\eta(\pi, p^*) &\leq \lambda_n \tilde{\Sigma}_n^d(\pi) + \beta_n^r \tilde{\Sigma}_n^r(\pi) + \mathbb{E}_{\pi,\mu_n^d}\Big[\sum_{t=0}^{H-1} \gamma^t \mu_n^r(\hat{s}_t, a_t)\Big] \\
&= \eta_n^{UCB}(\pi, \mu_n^d), \qquad \text{(by definition)}
\end{aligned}$$

where a_n, b_n correspond to the functions β_n^r, λ_n, respectively. Since $\lambda_n \in \mathcal{O}(\sqrt{\Gamma_{d,N}})$, and $\beta_n^r \in \mathcal{O}(\sqrt{\Gamma_{r,N}})$, we arrive at the theoretical bounds for coefficients a_n, b_n presented in Lemma D.4. □

Now, we introduce the analogue of Assumption 5.6 for the reduced UCB objective. This will be used in the proof of Lemma D.6 below.

Assumption D.5 (No planner sub-optimality for reduced UCB objective)**.** The policy π_n^{red} induced by the planning procedure through optimizing the reduced UCB objective equation 17 is equal to the optimal policy $\pi_n^{*,\mathrm{red}}$ which maximizes equation 17. That is, $\pi_n^{\mathrm{red}} = \pi_n^{*,\mathrm{red}}$.

Lemma D.6 (Single episode regret bound)**.** *Let Assumptions 5.1, 5.2, and D.5 hold, and assume that the learned dynamics and reward are well-calibrated. Consider the definition of r_n in equation 18, and let a_n, b_n be as defined in Lemma D.4 . Then, $\forall n > 0$ with probability at least $1-\delta$,*

$$r_n \leq (b_n'^2 + 2b_n')\Sigma_n^d(\pi_n^{\mathrm{red}}) + (a_n^2 + a_n)\Sigma_n^r(\pi_n^{\mathrm{red}}), \tag{27}$$

where $b_n' = \max\{b_n, \lambda_n\}$ and π_n^{red} is the planner induced policy, assumed to maximize equation equation 17.

Proof. We start with the definition of r_n presented in equation 18:

$$\begin{aligned}
r_n &= \eta(\pi^*, p^*) - \eta(\pi_n, p^*) \\
&\leq \eta_n^{UCB}(\pi^*, \mu_n^d) - \eta(\pi_n^{\mathrm{red}}, p^*) && \text{(Lemma D.4)} \\
&\leq \eta_n^{UCB}(\pi_n^{*,\mathrm{red}}, \mu_n^d) - \eta(\pi_n^{\mathrm{red}}, p^*) && \text{(Optimization of equation 17)} \\
&= \eta_n^{UCB}(\pi_n^{\mathrm{red}}, \mu_n^d) - \eta(\pi_n^{\mathrm{red}}, p^*) && \text{(Assumption D.5)} \\
&= \eta(\pi_n^{\mathrm{red}}, \mu_n^d) + a_n \tilde{\Sigma}_n^r(\pi_n^{\mathrm{red}}) + b_n \tilde{\Sigma}_n^d(\pi_n^{\mathrm{red}}) - \eta(\pi_n^{\mathrm{red}}, p^*) \\
&\leq \lambda_n \Sigma_n^d(\pi_n^{\mathrm{red}}) + a_n \tilde{\Sigma}_n^r(\pi_n^{\mathrm{red}}) + b_n \tilde{\Sigma}_n^d(\pi_n^{\mathrm{red}}) && \text{(Lemma D.3)}
\end{aligned}$$

Let $b'_n = \max\{b_n, \lambda_n\}$. Then

$$\begin{aligned} r_n &\le 2b'_n\Sigma^d_n(\pi^{\text{red}}_n) + b'_n(\tilde{\Sigma}^d_n(\pi^{\text{red}}_n) - \Sigma^d_n(\pi^{\text{red}}_n)) + a_n\Sigma^r_n(\pi^{\text{red}}_n) + a_n(\tilde{\Sigma}^r_n(\pi^{\text{red}}_n) - \Sigma^r_n(\pi^{\text{red}}_n)) \\ &\le (2b'_n + b'^2_n)\Sigma^d_n(\pi^{\text{red}}_n) + (a^2_n + a_n)\Sigma^r_n(\pi^{\text{red}}_n). \end{aligned}$$

In the last inequality, we use the fact that $\sigma^r_n(\cdot)$ and $\left\|\sigma^d_n(\cdot)\right\|$ are bounded and positive. Hence, they qualify as valid "reward signals" which can be substituted into Lemma D.3, yielding $\tilde{\Sigma}^d_n(\pi^{\text{red}}_n) - \Sigma^d_n(\pi^{\text{red}}_n) \le b'_n\Sigma^d_n(\pi^{\text{red}}_n)$ and $\tilde{\Sigma}^r_n(\pi^{\text{red}}_n) - \Sigma^r_n(\pi^{\text{red}}_n) \le a_n\Sigma^r_n(\pi^{\text{red}}_n)$ Sukhija et al. (2025b) □

Theorem D.7 (Regret bound with dynamics and reward uncertainty bonuses)**.** *Let Assumptions 5.1, 5.2, and 5.6 hold, and let the learned GP dynamics and reward models be well-calibrated. The cumulative regret after N episodes of real environment interaction satisfies, with probability $1-\delta$:*

$$R_N \le \mathcal{O}\Big(H^3\sqrt{N}\left(\Gamma^{3/2}_{d,N} + \Gamma^{3/2}_{r,N}\right)\Big). \tag{28}$$

Proof. We begin with the definition of the cumulative regret R_N:

$$\begin{aligned} R_N &= \sum_{n=1}^{N} r_n \\ &\le \sum_{n=1}^{N}\Big[(2b'_n + b'^2_n)\Sigma^d_n(\pi^{\text{ind}}_n) + (a^2_n + a_n)\Sigma^r_n(\pi^{\text{red}}_n)\Big] \\ &\le (2b'_N + b'^2_N)\sum_{n=1}^{N}\Sigma^d_n(\pi^{\text{red}}_n) \;+\; (a^2_N + a_N)\sum_{n=1}^{N}\Sigma^r_n(\pi^{\text{red}}_n) \\ &\le (2b'_N + b'^2_N)\sum_{n=1}^{N}\mathbb{E}_{p^*}\Big[\sum_{t=0}^{H-1}\left\|\sigma^d_{\text{epi}_n}(s_t, \pi^{\text{red}}_n(s_t))\right\|\Big] \;+\; (a^2_N + a_N)\sum_{n=1}^{N}\mathbb{E}_{p^*}\Big[\sum_{t=0}^{H-1}\sigma^r_{\text{epi}_n}(s_t, \pi^{\text{red}}_n(s_t))\Big] \\ &\le (2b'_N + b'^2_N)\sqrt{NH}\sqrt{\sum_{n=1}^{N}\mathbb{E}_{p^*}\Big[\sum_{t=0}^{H-1}\left\|\sigma^d_{\text{epi}_n}(s_t, \pi^{\text{red}}_n(s_t))\right\|^2\Big]} \\ &\qquad + \;(a^2_N + a_N)\sqrt{NH}\sqrt{\sum_{n=1}^{N}\mathbb{E}_{p^*}\Big[\sum_{t=0}^{H-1}\left(\sigma^r_{\text{epi}_n}(s_t, \pi^{\text{red}}_n(s_t))\right)^2\Big]} \quad \text{(Cauchy-Schwartz + Jensen's inequality)} \\ &\le C_1(2b'_N + b'^2_N)H\sqrt{N\Gamma_{d,NH}} + C_2(a^2_N + a_N)H\sqrt{N\Gamma_{r,NH}} \quad \text{(From Lemma 17 in Curi et al. (2020))} \end{aligned}$$

Finally, notice that from Lemma D.4 and Lemma D.3 that $a_N \propto H\beta^r_n$, $b_N \propto \lambda_N$, $\lambda_N \propto H\beta^d_n$, $\beta^d_n \in \mathcal{O}(\sqrt{\Gamma_{d,N}})$, and $\beta^r_n \in \mathcal{O}(\sqrt{\Gamma_{r,N}})$. Substituting, we arrive at equation 28. □

D.3 Analysis of Discounted Infinite Horizon Planning with the Complete UCB Objective

We now utilize and generalize the ideas for the finite horizon regret proof in Appendix D.2 to derive our final infinite-horizon regret bound with the complete UCB objective equation 7.

We start by extending the definitions of the finite-horizon cumulative uncertainty (20)-(23) to be infinite-horizon.

Discounted Infinite-Horizon Cumulative Uncertainty For episode n and policy π, define

$$\widetilde{\Sigma}^r_{\gamma,n}(\pi) := \mathbb{E}_{\pi,\mu^d_n}\Big[\sum_{t=0}^{\infty}\gamma^t\sigma^r_{\mathrm{epi_n}}(\hat{s}_t,a_t)\Big], \tag{29}$$

$$\Sigma^r_{\gamma,n}(\pi) := \mathbb{E}_{\pi,p^*}\Big[\sum_{t=0}^{\infty}\gamma^t\sigma^r_{\mathrm{epi_n}}(s_t,a_t)\Big], \tag{30}$$

$$\widetilde{\Sigma}^d_{\gamma,n}(\pi) := \mathbb{E}_{\pi,\mu^d_n}\Big[\sum_{t=0}^{\infty}\gamma^t\Big\|\sigma^d_{\mathrm{epi_{n,j}}}(\hat{s}_t,a_t)\Big\|\Big], \tag{31}$$

$$\Sigma^d_{\gamma,n}(\pi) := \mathbb{E}_{\pi,p^*}\Big[\sum_{t=0}^{\infty}\gamma^t\Big\|\sigma^d_{\mathrm{epi_{n,j}}}(s_t,a_t)\Big\|\Big]. \tag{32}$$

By Assumption 5.4, there exists constant α_d and α_r such that for all episode n,

$$\sigma^q_n(s,a) = \alpha_d\mathbb{E}\Big[\sum_{n=1}^{\infty}\gamma^t\|\sigma^d_{\mathrm{epi}_n}(s_t,a_t)\|\Big] + \alpha_r\mathbb{E}\Big[\sum_{n=1}^{\infty}\gamma^t\sigma^r_{\mathrm{epi}_n}(s_t,a_t)\Big].$$

We give a brief justification for this assumption by proving a lemma bounding the epistemic uncertainty of Q. We assume that the underlying value function V is bounded by $V_{\max}$ and Lipchitz with a Lipchitz constant L_V. We start by the following lemma:

Lemma D.8. *Let Assumption 5.1, 5.2 hold. Given dynamics p^*, μ^d_n and rewards r, μ^r_n. We have the following bound*

$$|\bar{Q}^\pi_{\mu_n}(s,a) - Q^\pi_{p^*}(s,a)| \le \beta^r_n\mathbb{E}_{\pi,\mu^d_n}\left[\sum_{t=0}^{\infty}\gamma^t\sigma^r_n(s_t,a_t)\right] + \gamma L_{V^\pi_{p^*}}\beta^d_n\mathbb{E}_{\mu^d_n,\pi}\left[\sum_{t=0}^{\infty}\gamma^t\|\sigma^d_n(s_t,a_t)\|\right]. \tag{33}$$

Proof. Define the transition operators

$$(P^\pi_p f)(s,a) := \mathbb{E}_{s'\sim p(\cdot|s,a)}\big[f(s',\pi(s'))\big], \tag{34}$$

$$(P_p g)(s,a) := \mathbb{E}_{s'\sim p(\cdot|s,a)}\big[g(s')\big]. \tag{35}$$

Then we have,

$$\begin{aligned}\bar{Q}^\pi_{\mu^d_n} - Q^\pi_{p^*} &= (\mu^r_n - r) + \gamma\big(P^\pi_{\mu^d_n}\bar{Q}^\pi_{\mu^d_n} - P^\pi_{p^*}Q^\pi_{p^*}\big)\\ &= (\mu^r_n - r) + \gamma P^\pi_{\mu^d_n}(\bar{Q}^\pi_{\mu^d_n} - Q^\pi_{p^*}) + \gamma(P^\pi_{\mu^d_n} - P^\pi_{p^*})Q^\pi_{p^*}.\end{aligned}$$

Rearranging gives the resolvent form

$$(I - \gamma P^\pi_{\mu^d_n})(\bar{Q}^\pi_{\mu^d_n} - Q^\pi_{p^*}) = (\mu^r_n - r) + \gamma(P^\pi_{\mu^d_n} - P^\pi_{p^*})Q^\pi_{p^*}.$$

Since rewards are bounded, $(I - \gamma P^\pi_{\mu^d_n})^{-1}$ exists on bounded functions, use the Neumann series $(I - \gamma P^\pi_{\mu^d_n})^{-1} = \sum_{t=0}^{\infty}(\gamma P^\pi_{\mu^d_n})^t$ to get

$$\bar{Q}^\pi_{\mu^d_n} - Q^\pi_{p^*} = \sum_{t=0}^{\infty}(\gamma P^\pi_{\mu^d_n})^t(\mu^r_n - r) \;+\; \gamma\sum_{t=0}^{\infty}(\gamma P^\pi_{\mu^d_n})^t(P^\pi_{\mu^d_n} - P^\pi_{p^*})Q^\pi_{p^*}.$$

Take absolute values pointwise and apply the triangle inequality:

$$\left|\bar{Q}^{\pi}_{\mu^d_n} - Q^{\pi}_{p^*}\right| \leq \sum_{t=0}^{\infty} \gamma^t \left|(P^{\pi}_{\mu^d_n})^t(\mu^r_n - r)\right| + \gamma \sum_{t=0}^{\infty} \gamma^t \left|(P^{\pi}_{\mu^d_n})^t (P^{\pi}_{\mu^d_n} - P^{\pi}_{p^*}) Q^{\pi}_{p^*}\right|.$$

Now interpret $(P^{\pi}_{\mu^d_n})^t$ as expectation over a length-t rollout under (μ^d_n, π) starting from (s, a): for any bounded g,

$$((P^{\pi}_{\mu^d_n})^t g)(s,a) = \mathbb{E}_{\mu^d_n, \pi}[g(s_t, a_t) \mid s_0 = s, a_0 = a].$$

Applying this with $g = \mu^r_n - r$ yields

$$\begin{aligned}\left|((P^{\pi}_{\mu^d_n})^t(\mu^r_n - r))(s,a)\right| &\leq \mathbb{E}_{\mu^d_n, \pi}\left[\left|\mu^r_n(s_t,a_t) - r(s_t,a_t)\right| \,\middle|\, s_0 = s, a_0 = a\right] \\ &\leq \beta^r_n \mathbb{E}_{\pi, \mu^d_n}[\sigma^r_n(s_{,t}\,, a_t)]\end{aligned}$$

For the second term, note that

$$\left((P^{\pi}_{\mu^d_n} - P^{\pi}_{p^*})Q^{\pi}_{p^*}\right)(s,a) = \mathbb{E}_{\mu^d_n}\left[V^{\pi}_{p^*}(s')\right] - \mathbb{E}_{p^*}\left[V^{\pi}_{p^*}(s')\right] = \left((P_{\mu^d_n} - P_{p^*})V^{\pi}_{p^*}\right)(s,a),$$

so

$$\begin{aligned}\left|((P^{\pi}_{\mu^d_n})^t(P^{\pi}_{\mu^d_n} - P^{\pi}_{p^*})Q^{\pi}_{p^*})(s,a)\right| &\leq \mathbb{E}_{\mu^d_n, \pi}\left[\left|(P_{\mu^d_n} - P_{p^*})V^{\pi}_{p^*}(s_t,a_t)\right| \,\middle|\, s_0 = s, a_0 = a\right] \\ &\leq L_{V^{\pi}_{p^*}} \mathbb{E}_{\mu^d_n, \pi}[W(p_{\mu^d_n}(\cdot|s_t,a_t), p_{p^*}(\cdot|s_t,a_t))] \\ &= L_{V^{\pi}_{p^*}} \mathbb{E}_{\mu^d_n, \pi}[|\mu^d_n(s_t,a_t) - p^*(s_t,a_t)|] \\ &\leq L_{V^{\pi}_{p^*}} \beta^d_n \mathbb{E}_{\mu^d_n, \pi}[\|\sigma^d_n(s_t,a_t)\|].\end{aligned}$$

$W(\cdot,\cdot)$ is the Wasserstein distance between two distributions. For two Gaussian distributions, this reduces to the distance between their means. □

The true Q-error $|Q^{\pi}_{p^*} - \bar{Q}^{\pi}_{\mu_n}|$ is unobservable. However, Lemma D.8 shows that this error can be upper-bounded by a discounted accumulation of the posterior standard deviations of the dynamics and reward models. Therefore, we utilize this bound as a *computable epistemic uncertainty proxy* for the Q-function. Intuitively, because Q is a functional of the reward and dynamics, it's within reason that the epistemic uncertainty of Q should be built from uncertainty the reward model and dynamics model.

Note that although we explicitly reference Assumption 5.1, the Gaussian nature of the dynamics and reward in order to characterize their Wasserstein distance, this does not mean that the process noise also has to be Gaussian. Please refer to Appendix D.4 for further discussions on settings with sub-gaussian noise.

We can give a proof of Lemma 5.7 following the previous proof of Lemma D.4 and also treat Q as the tail return estimator.

Proof of Lemma 5.7. By Lemma B.7 in Sukhija et al. (2025b),

$$\begin{aligned}\eta_\gamma(\pi, p^*) &\leq \lambda_n \tilde{\Sigma}^d_{\gamma,n}(\pi) + \eta_\gamma(\pi, \mu^d_n) \\ &= \lambda_n \mathbb{E}_{\pi, \mu^d_n}\left[\sum_{t=0}^{\infty} \gamma^t \|\sigma^d_{\mathrm{epi}_n}(s_t,a_t)\|\right] + \mathbb{E}_{\pi, \mu^d_n}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t)\right] \\ &= \lambda_n \mathbb{E}_{\pi, \mu^d_n}\left[\sum_{t=0}^{H-1} \gamma^t \|\sigma^d_{\mathrm{epi}_n}(s_t,a_t)\|\right] + \mathbb{E}_{\pi, \mu^d_n}\left[\sum_{t=0}^{H-1} \gamma^t (r(s_t,a_t) - \mu^r_n(s_t,a_t))\right] \\ &\quad + \mathbb{E}_{\pi, \mu^d_n}\left[\sum_{t=0}^{H-1} \gamma^t \mu^r_n(s_t,a_t)\right] + \gamma^H \mathbb{E}_{\pi, \mu^d_n}\left[\sum_{t=0}^{\infty} \gamma^t \left(\|\sigma^d_{\mathrm{epi}_n}(s_{t+H}, a_{t+H})\| + r(s_{t+H}, a_{t+H})\right)\right] \\ &\leq \lambda_n \tilde{\Sigma}^d_{\gamma,n}(\pi) + \beta^r_n \tilde{\Sigma}^r_{\gamma,n}(\pi) + \mathbb{E}\left[\sum_{t=0}^{H-1} \gamma^t \mu^r_n(s_t,a_t)\right]\end{aligned}$$

$$
\begin{aligned}
&+ \gamma^H \lambda_n \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \|\sigma^d_{\mathrm{epi}_n}(s_{t+H}, a_{t+H})\|\right] + \gamma^H \beta^r_n \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \|\sigma^r_{\mathrm{epi}_n}(s_{t+H}, a_{t+H})\|\right] \\
&+ \gamma^H \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mu^r_n(s_{t+H}, a_{t+H})\right] \\
&\leq \lambda_n \tilde{\Sigma}^d_{\gamma,n}(\pi) + \beta^r_n \tilde{\Sigma}^r_{\gamma,n}(\pi, \mu^d_n) + \mathbb{E}\left[\sum_{t=0}^{H-1} \gamma^t \mu^r_n(s_t, a_t)\right] \\
&\quad + \gamma^H (c_n \mathbb{E}\left[\sigma^q_{\mathrm{epi}_n}(s_H, a_H)\right] + \mathbb{E}\big[\bar{Q}^{\pi}_{\mu_n}(s_H, a_H)\big]) \\
&\leq \lambda_n \tilde{\Sigma}^d_{\gamma,n}(\pi) + \beta^r_n \tilde{\Sigma}^r_{\gamma,n}(\pi) + \mathbb{E}\left[\sum_{t=0}^{H-1} \gamma^t \mu^r_n(s_t, a_t)\right] \\
&\quad + \gamma^H \mathbb{E}\Big[\hat{Q}^{\pi}_{\mu_n}(s_H, a_H) + c_n \sigma^q_n(s_H, a_H)\Big] + \gamma^H \epsilon_{q,n} \\
&= \eta^{\mathrm{UBP}}_{\gamma,n}(\pi, \mu^d_n) + \gamma^H \epsilon_{q,n},
\end{aligned}
$$

where $c_n = \frac{\max\{\lambda_n, \beta^r_n\}}{\min\{\alpha_r, \alpha_d\}} \in \mathcal{O}\big(\max\{\sqrt{\Gamma_{r,N}}, \sqrt{\Gamma_{d,N}}\}\big)$.

□

Lemma D.9 (Single episode regret bound for infinite-horizon return)**.** *Let Assumptions 5.1, 5.2, and 5.6 hold, and assume that the learned dynamics and reward are well-calibrated. Consider the definition of r_n in equation 8, and let a_n, b_n, c_n be as defined in Lemma D.4 . Then, $\forall n > 0$ with probability at least $1 - \delta$,*

$$
r_{\gamma,n} \leq \left(b'^2_n + b'_n c'_n + 2b'_n + c'_n\right) \Sigma^d_{\gamma,n}(\pi^{\mathrm{ind}}_n) + \left(2a^2_n + a_n c'_n + 2a_n + c'_n\right) \Sigma^r_{\gamma,n}(\pi^{\mathrm{ind}}_n) + 2\gamma^H \epsilon_{q,n} \quad (36)
$$

where $b'_n = \max\{b_n, \lambda_n\}$, $c'_n = c_n \max\{\alpha_r, \alpha_d\}$ and π^{ind}_n is the planner induced policy, assumed to maximize equation equation 1.

Proof.

$$
\begin{aligned}
r_n &= \eta_\gamma(\pi^*, p^*) - \eta_\gamma(\pi_n, p^*) \\
&\leq \eta^{\mathrm{UCB}}_{\gamma,n}(\pi^*, \mu^d_n) - \eta_\gamma(\pi^{\mathrm{ind}}_n, p^*) + \gamma^H \epsilon_{q,n} && \text{(Lemma 5.7)} \\
&\leq \eta^{\mathrm{UCB}}_{\gamma,n}(\pi^*_n, \mu^d_n) - \eta_\gamma(\pi^{\mathrm{ind}}_n, p^*) + \gamma^H \epsilon_{q,n} && \text{(Optimization of equation 7)} \\
&= \eta^{\mathrm{UCB}}_{\gamma,n}(\pi^{\mathrm{ind}}_n, \mu^d_n) - \eta_\gamma(\pi^{\mathrm{ind}}_n, p^*) + \gamma^H \epsilon_{q,n} && \text{(Assumption 5.6)} \\
&= \eta(\pi^{\mathrm{ind}}_n, \mu_n) + a_n \tilde{\Sigma}^r_n(\pi^{\mathrm{ind}}_n) + b_n \tilde{\Sigma}^d_n(\pi^{\mathrm{ind}}_n) \\
&\quad + \gamma^H \mathbb{E}_{\mu^d_n}\left[\hat{Q}^{\pi^{\mathrm{ind}}_n}_{\mu^d_n}(s_H, a_H) + c_n \sigma^q_{\mathrm{epi}_n}(s_H, a_H)\right] - \eta_\gamma(\pi^{\mathrm{ind}}_n, p^*) + \gamma^H \epsilon_{q,n} \\
&= \eta(\pi^{\mathrm{ind}}_n, \mu_n) - \eta_\gamma(\pi^{\mathrm{ind}}_n, p^*) + a_n \tilde{\Sigma}^r_n(\pi^{\mathrm{ind}}_n) + b_n \tilde{\Sigma}^d_n(\pi^{\mathrm{ind}}_n) \\
&\quad + \gamma^H \mathbb{E}_{\mu^d_n}\left[\sum_{t=0}^{\infty} \gamma^H \big(\mu^r_n(s_{t+H}, a_{t+H}) - r(s_{t+H}, a_{t+H})\big) + c_n \sigma^q_{\mathrm{epi}_n}(s_H, a_H)\right] + 2\gamma^H \epsilon_{q,n} \\
&\leq (b'^2_n + 2b'_n) \Sigma^d_{\gamma,n}(\pi^{\mathrm{ind}}_n) + (a^2_n + a_n) \Sigma^r_{\gamma,n}(\pi^{\mathrm{ind}}_n) && \text{(Lemma B.7 from Sukhija et al. (2025b))} \\
&\quad + \gamma^H \mathbb{E}_{\mu^d_n}\left[\beta^r_n \sum_{t=0}^{\infty} \gamma^t \sigma^r_{\mathrm{epi}_n}(s_t, a_t) + c'_n \sum_{t=0}^{\infty} \gamma^t \Big(\sigma^r_{\mathrm{epi}_n}(s_{t+H}, a_{t+H}) + \|\sigma^d_{\mathrm{epi}_n}(s_{t+H}, a_{t+H})\|\Big)\right] + 2\gamma^H \epsilon_{q,n} \\
&\leq (b'^2_n + 2b'_n) \Sigma^d_{\gamma,n}(\pi^{\mathrm{ind}}_n) + (2a^2_n + 2a_n) \Sigma^r_{\gamma,n}(\pi^{\mathrm{ind}}_n) \\
&\quad + \gamma^H \mathbb{E}_{\mu^d_n}\left[c'_n \sum_{t=0}^{\infty} \gamma^t \Big(\sigma^r_{\mathrm{epi}_n}(s_{t+H}, a_{t+H}) + \|\sigma^d_{\mathrm{epi}_n}(s_{t+H}, a_{t+H})\|\Big)\right] + 2\gamma^H \epsilon_{q,n} \\
&\leq (b'^2_n + b'_n c'_n + 2b'_n + c'_n) \Sigma^d_{\gamma,n}(\pi^{\mathrm{ind}}_n) + (2a^2_n + a_n c'_n + 2a_n + c'_n) \Sigma^r_{\gamma,n}(\pi^{\mathrm{ind}}_n) + 2\gamma^H \epsilon_{q,n}.
\end{aligned}
$$

where $c'_n = c_n \max\{\alpha_r, \alpha_d\}$.

□

This lemma yields our bound for the infinite-horizon regret.

Proof of Theorem 5.8. First note that since a_n, b_n are monotonic, we know that c'_n is also monotonic.

$$
\begin{aligned}
R_N &= \sum_{n=1}^{N} r_n \\
&\leq \sum_{n=1}^{N} \left(b_n'^2 + b_n c'_n + 2b'_n + c'_n\right) \Sigma^d_{\gamma,n}(\pi^{\text{ind}}_n) + \left(2a_n^2 + a_n c'_n + 2a_n + c'_n\right) \Sigma^r_{\gamma,n}(\pi^{\text{ind}}_n) + 2\gamma^H \epsilon_{q,n} \\
&\leq \left(b_N'^2 + b_N c'_N + 2b'_N + c'_N\right) \sum_{n=1}^{N} \Sigma^d_{\gamma,n}(\pi^{\text{ind}}_n) + \left(2a_N^2 + a_N c'_N + 2a_N + c'_N\right) \sum_{n=1}^{N} \Sigma^r_{\gamma,n}(\pi^{\text{ind}}_n) + 2\gamma^H \sum_{n=1}^{N} \epsilon_{q,n} \\
&:= B_N \sum_{n=1}^{N} \Sigma^d_{\gamma,n}(\pi^{\text{ind}}_n) + A_N \sum_{n=1}^{N} \Sigma^r_{\gamma,n}(\pi^{\text{ind}}_n) + 2\gamma^H \sum_{n=1}^{N} \epsilon_{q,n}.
\end{aligned}
$$

Since $\mathcal{O}\big(\max\{\sqrt{\Gamma_{r,N}}, \sqrt{\Gamma_{d,N}}\}\big) = \mathcal{O}(\sqrt{\Gamma_{r,N}} + \sqrt{\Gamma_{d,N}})$, we have $A_N, B_N \in \mathcal{O}(\Gamma_{r,N} + \Gamma_{d,N})$. Consider

$$
\begin{aligned}
\sum_{n=1}^{N} \Sigma^d_{\gamma,n}(\pi^{\text{ind}}_n) &\leq \sqrt{N} \sqrt{\sum_{n=1}^{N} \left(\Sigma^d_{\gamma,n}(\pi^{\text{ind}}_n)\right)^2} \\
&\leq \sqrt{N} \sqrt{\sum_{n=1}^{N} \mathbb{E}\left[\left(\sum_{t=0}^{\infty} \gamma^t \|\sigma^d_{\text{epi}_n}\|\right)^2\right]} \\
&\leq \sqrt{N} \sqrt{\sum_{n=1}^{N} \mathbb{E}\left[\left(\sum_{t=0}^{\infty} \gamma^t\right)\left(\sum_{t=0}^{\infty} \gamma^t \|\sigma^d_{\text{epi}_n}(s_t, a_t)\|^2\right)\right]} \\
&= \sqrt{\frac{N}{1-\gamma}} \sqrt{\sum_{n=1}^{N} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \|\sigma^d_{\text{epi}_n}(s_t, a_t)\|^2\right]} \\
&\leq \sqrt{\frac{C_\gamma N \Gamma_{N \log N} \log N}{1-\gamma} + \frac{C\,(\sigma^d_{\max})^2\, N \log N}{(1-\gamma)^2}}
\end{aligned}
$$

(Proof of Theorem. 5.5 from Sukhija et al. (2025b))

with $\sigma_{\max}$ as defined in Assumption 5.2 and for some constants C and C_γ. Following similar steps, we can prove an analogous inequality for $\Sigma^r_{\gamma,n}$. By defining $e_{q,N} = \sum_{n=1}^{N} \epsilon_{q,n}$, we obtain the bound in equation 11. □

Explicit infinite-horizon regret bound While Theorem 5.8 is written in terms of the GP information gain, since both reward and dynamics are modeled as functions of state and action (i.e., input is (s, a)), the input dimension in the kernel is $d_s + d_a$. Therefore, once a kernel is chosen, the asymptotic growth of Γ_N explicitly depends on the dimension. We demonstrate that for each of the linear, RBF, and Matérn kernels, our regret bound remains sublinear.

Let $d_x = d_s + d_a$ and $M = N \log N$. Since both the reward model and each coordinate of the dynamics model are functions of the joint input $x = (s, a) \in \mathbb{R}^{d_x}$, our Theorem 5.8 gives: $R_{\gamma,N} \leq H^3 \sqrt{N}\left(\Gamma^{3/2}_{d,M} + \Gamma^{3/2}_{r,M}\right) + e_{q,N}$. Algebraically substituting the information-gain bounds for each of the kernels yields the following kernel-specific, dimension-dependent variants of Theorem 5.8:

- **Linear kernel**: $\Gamma_M = \mathcal{O}(d_x \log M)$

$$\implies \quad R_{\gamma,N} = \tilde{\mathcal{O}}\big(H^3 \sqrt{N} d_x^{3/2} + e_{q,N}\big)$$

- **RBF kernel**: $\Gamma_M = \mathcal{O}(\log^{d_x+1} M)$

$$\implies \quad R_{\gamma,N} = \tilde{\mathcal{O}}\big(H^3\sqrt{N}\log^{\frac{3}{2}(d_x+1)} N + e_{q,N}\big)$$

- **Matérn kernel**: $\Gamma_M = \mathcal{O}\left(M^{\frac{d_x}{2\nu+d_x}}\log^{\frac{2\nu}{2\nu+d_x}} M\right)$

$$\implies \quad R_{\gamma,N} = \tilde{\mathcal{O}}\left(H^3 N^{\frac{1}{2}+\frac{3d_x}{2(2\nu+d_x)}} + e_{q,N}\right)$$

Therefore, it is seen that for linear and RBF, that our regret bound remains sublinear in N for fixed state and action dimension. For the Matérn with smoothness ν, for sublinear growth a sufficient condition is $\frac{1}{2} + \frac{3d_x}{2(2\nu+d_x)} < 1$, or $d_x < \nu$. However, this may be too conservative. There exist work that show function optimization in the RKHS of a Matérn kernel obtain sublinear regret under broader conditions, such as Janz et al. (2023).

D.4 Analysis for Settings With Sub-Gaussian Process Noise

We now discuss how our infinite horizon regret bound extends beyond Gaussian process noise. In Assumption 5.1, we assume that the process noise of the true dynamics p^* is Gaussian. As our theoretical work is current presented, the Gaussianity of the noise is necessary in two locations: (i) to obtain high-probability confidence intervals for the estimated dynamics and reward, and (ii) the invocation of Lemma B.7 from Sukhija et al. (2025b), which is a simulation-lemma type bound for the value function under the true and discounted dynamics. Therefore, relaxing the Gaussian noise to subgaussian noise requires the addressing of both these points.

The first use is not inherently Gaussian - we simply assumed so to simplify the analysis. The high-probability confidence relation required between the reward and dynamics model error and their posterior GP uncertainty can indeed be obtained under conditionally subgaussian noise. In particular, we refer to analysis done by Chowdhury & Gopalan (2017); in the paper, the authors operate within an agnostic RKHS setting and also utilize the standard closed form for the GP posterior mean and variance as presented in equation 13. It is explictly stated that the Gaussian prior assumption are only an aid to algorithm design, and that the true process noise may be a conditionally R-subgaussian martingale-difference sequence. They then provide a uniform confidence bound in their Theorem 2, stated below. Note that this bound is precisely the well-calibrated confidence bound our theory requires.

$$|\mu_{t-1}(x) - f(x)| \leq \Big(B + R\sqrt{2(\gamma_{t-1} + 1 + \ln(1/\delta))}\Big)\sigma_{t-1}(x) \quad \text{(Theorem 2, [5])}$$

However, as Lemma B.7 from Sukhija et al. (2025b) relies on a Gaussian smoothing argument through their use of Lemma C.2 from Kakade et al. (2020), we need explicitly address settings with subgaussian noise. We will show that even under the subgaussian noise, *the regret bound asymptotically remains sublinear*.

We assume that the true dynamics has additive subgaussian noise: $s_{t+1} = g^*(s_t, a_t) + \xi_t$, where ξ_t is a conditionally mean-zero σ_d-subgaussian martingale-difference sequence. The learned dynamics is $\hat{s}_{t+1} = \mu_n^d(\hat{s}_t, a_t) + \xi_t$. To avoid overloading the notation, we denote the true dynamics as g^*. However, in actuality, $g^* = p^*$. We also make the following assumption:

Assumption D.10 (Lipschitzness of dynamics, policies, and reward)**.** The true dynamics g^*, the reward r, and all $\pi \in \Pi$ are L_g, L_r, and uniformly L_π Lipschitz respectively and satisfy a discounted stability condition $\gamma L_g(1 + L_\pi) < 1$

An immediate corollary of the above assumption is the following:

Corollary D.11 (Lipschitz-ness of value function)**.** *Let all conditions in Assumption D.10 hold. Then, the value function $V_g^\pi(s)$ induced by g^* under any policy $\pi \in \Pi$ is uniformly L_V-Lipschitz with*

$$L_V \leq \frac{L_r(1 + L_\pi)}{1 - \gamma L_g(1 + L_\pi)}. \tag{37}$$

Proof. Take any two starting states s_0 and $\tilde{s}_0$ and run the same deterministic policy from each state, yielding two actions, a_0 and $\tilde{a}_0$. Furthermore, consider the one-step dynamics which yield next states s_1 and $\tilde{s}_1$, where $s_1 = g^*(s_0, a_0) + \xi_0$ and $\tilde{s}_1 = g^*(\tilde{s}_0, \tilde{a}_0) + \xi_0$.

Let (s_0, a_0) denote the concatenated vector of s_0 and a_0, and let $(\tilde{s}_0, \tilde{a}_0)$ denote the concatenated vector of $\tilde{s}_0$ and $\tilde{a}_0$. We have that:

$$\begin{aligned}\|(s_0, a_0) - (\tilde{s}_0, \tilde{a}_0)\| &= \sqrt{\|s_0 - \tilde{s}_0\|^2 - \|a_0 - \tilde{a}_0\|^2} \leq \|s_0 - \tilde{s}_0\| + \|a_0 - \tilde{a}_0\| \\ &\leq (1 + L_\pi)\|s_0 - \tilde{s}_0\|.\end{aligned}$$

In the last step, we use the Lipschitz-ness of the policy, or the fact that $\|a_0 - \tilde{a}_0\| \leq L_\pi \|s_0 - \tilde{s}_0\|$. Now, we consider the difference between s_1 and $\tilde{s_1}$. Here, we cancel the process noise: $s_1 - \tilde{s}_1 = g(s_0, a_0) - g(\tilde{s}_0, \tilde{a}_0)$. Then, using the Lipschitz-ness of the dynamics we have the following:

$$\|s_1 - \tilde{s}_1\| \leq L_g \|(s_0, a_0) - (\tilde{s}_0, \tilde{a}_0)\| \leq L_g(1 + L_\pi)\|s_0 - \tilde{s}_0\| = \rho \|s_0 - \tilde{s}_0\|$$

where we define $\rho = L_g(1 + L_\pi)$

Again consider the one-step dynamics beginning at s_1 and $\tilde{s}_1$ after running the same deterministic policy from each state and taking actions a_1 and $\tilde{a}_1$. We obtain two more states $s_2 = g^*(s_1, a_1) + \xi_1$ and $\tilde{s}_2 = g^*(\tilde{s}_1, \tilde{a}_1) + \xi_1$ It can easily be seen that $\|(s_1, a_1) - (\tilde{s}_1, \tilde{a}_1)\| \leq (1 + L_\pi)|s_1 - \tilde{s}_1\|$ and hence $\|s_2 - \tilde{s}_2\| \leq \rho^2 \|s_0 - \tilde{s}_0\|$ via the same argument as above. Iterating this argument, for some timestep t,

$$\|(s_t, a_t) - (\tilde{s}_t, \tilde{a}_t)\| \leq (1 + L_\pi)\|s_t - \tilde{s}_t\|.$$

$$\|s_t - \tilde{s}_t\| \leq \rho^t \|s_0 - \tilde{s}_0\|$$

Now, we compare rewards. Using the Lipschitz-ness of r and the equations above,

$$\begin{aligned}|r(s_t, a_t) - r(\tilde{s}_t, \tilde{a}_t)| &\leq L_r \|(s_t, a_t) - (\tilde{s}_t, \tilde{a}_t)\| \leq L_r(1 + L_\pi)\|s_t - \tilde{s}_t\| \\ &\leq L_r(1 + L_\pi)\rho^t \|s_0 - \tilde{s}_0\|.\end{aligned}$$

We then define the value functions

$$V_g^\pi(s_0) = \mathbb{E}_\xi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]$$

$$V_g^\pi(\tilde{s}_0) = \mathbb{E}_\xi \left[\sum_{t=0}^{\infty} \gamma^t r(\tilde{s}_t, \tilde{a}_t) \right] \tag{38}$$

Then, comparing one coupled rollout we have

$$\begin{aligned}|V_g^\pi(s_0) - V_g^\pi(\tilde{s}_0)| &\leq \sum_{t=0}^{\infty} \gamma^t |r(s_t, a_t) - r(\tilde{s}_t, \tilde{a}_t)| \\ &\leq \sum_{t=0}^{\infty} \gamma^t L_r(1 + L_\pi)\rho^t \|s_0 - \tilde{s}_0\| \\ &\leq L_r(1 + L_\pi) \left(\sum_{t=0}^{\infty} (\gamma\rho)^t \right) \|s_0 - \tilde{s}_0\|.\end{aligned}$$

Note that the infinte sum is only finite if $\gamma\rho = \gamma L_g(1 + L_\pi) < 1$, which is precisely the discounted stability condition required in Assumption D.10. If this is the case, then, the result of the infinte sum is $\sum_{t=0}^{\infty} (\gamma\rho)^t = \frac{1}{1-\gamma\rho}$. Therefore,

$$|V^\pi(s_0) - V^\pi(\tilde{s}_0)| \leq \frac{L_r(1 + L_\pi)}{1 - \gamma L_g(1 + L_\pi)} \|s_0 - \tilde{s}_0\|.$$

So, for any policy $\pi \in \Pi$ and any starting states, V_g^π is L_V-Lipschitz with L_V upper bounded by the coefficient above. □

Now, we introduce the following lemma, which is a simulation lemma similar to Lemma D.3 and Lemma B.7 of Sukhija et al. (2025b). It will be shown that

Lemma D.12 (Simulation lemma for settings with subgaussian noise)**.** *Let Assumptions 5.1, 5.2, D.10 hold, and let L_V be as defined in Corollary D.11. Then, there is a $\lambda^d_{n,\mathrm{SG}} = \gamma L_V \beta^d_n$. such that for every policy $\pi \in \Pi$ with probability at least $1-\delta$,*

$$\left|\eta_\gamma(\pi, p^*) - \eta_\gamma(\pi, \mu^d_n)\right| \leq \lambda^d_{n,\mathrm{SG}} \Sigma^d_{\gamma,n}(\pi) \tag{39}$$

$$\left|\eta_\gamma(\pi, p^*) - \eta_\gamma(\pi, \mu^d_n)\right| \leq \lambda^d_{n,\mathrm{SG}} \widetilde{\Sigma}^d_{\gamma,n}(\pi), \tag{40}$$

Proof. For simplicity, assume there is a deterministic policy π. The stochastic policy variant of the proof is the same, with an extra expectation over the action.

Define two value functions, $V_*(s_0) = V^\pi_{g^*}(s_0)$ and $V_\mu(s_0) = V^\pi_{\mu^d_n}(s_0)$, corresponding to true dynamics and learned dynamics. Consider their Bellman expansions, which are $V_*(s_0) = r(s, \pi(s_0)) + \gamma \mathbb{E}_{\xi_0}[V_*(g^*(s_0, \pi(s_0)) + \xi_0)]$ and $V_\mu(s_0) = r(s_0, \pi(s_0)) + \gamma \mathbb{E}_{\xi_0}[V_\mu(\mu^d_n(s_0, \pi(s_0)) + \xi_0)]$. Let $\Delta(s) = V_\mu(s) - V_*(s)$. Then,

$$\begin{aligned}
|\Delta(s_0)| &= |\gamma \mathbb{E}_{\xi_0}\left[V_\mu(\mu^d_n(s_0, \pi(s_0)) + \xi_0)\right] - \gamma \mathbb{E}_{\xi_0}\left[V_*(g^*(s_0, \pi(s_0)) + \xi_0)\right]| \\
&= |\gamma \mathbb{E}_{\xi_0}\left[V_\mu(\mu^d_n(s_0, \pi(s_0)) + \xi_0) - V_*(\mu^d_n(s_0, \pi(s_0)) + \xi_0)\right] \\
&\quad + \gamma \mathbb{E}_{\xi_0}\left[V_*(\mu^d_n(s_0, \pi(s_0)) + \xi_0) - V_*(g^*(s_0, \pi(s_0)) + \xi_0)\right]| \\
&\leq \gamma \mathbb{E}_{\xi_0}[|\Delta(s^\mu_1)|] + \gamma \mathbb{E}_{\xi_0}[|V_*(s^\mu_1) - V_*(s^*_1)|] \\
&\leq \gamma \mathbb{E}_{\xi_0}[|\Delta(s^\mu_1)|] + \gamma \mathbb{E}_{\xi_0}[L_V \|s^\mu_1 - s^*_1\|] \qquad \text{(Lipschitz-ness of } V_*) \\
&= \gamma \mathbb{E}_{\xi_0}[|\Delta(s^\mu_1)|] + \gamma \mathbb{E}_{\xi_0}[L_V \|\mu^d_n(s_0, \pi(s_0)) - g^*(s_0, \pi(s_0))\|] \\
&\leq \mathbb{E}_{\xi_0}[|\Delta(s^\mu_1)|] + \gamma \mathbb{E}_{\xi_0}[L_V \beta^d_n \|\sigma^d_{\mathrm{epi}_n}(s_0, \pi(s_0))\|] \qquad \text{(Definition D.1)} \\
&= \mathbb{E}_{\xi_0}[|\Delta(s^\mu_1)|] + \gamma L_V \beta^d_n \|\sigma^d_{\mathrm{epi}_n}(s_0, \pi(s_0))\|
\end{aligned}$$

where $s^\mu_{t+1} = \mu^d_n(s_t, \pi(s_t)) + \xi_t$, and $s^*_{t+1} = g^*(s_t, \pi(s_t)) + \xi_t$. Now, we repeat the above process for the next state produced by the learned model s^μ_1, yielding

$$|\Delta(s^\mu_1)| \leq \gamma \mathbb{E}_{\xi_1}[|\Delta(s^\mu_2)|] + \gamma L_V \beta^d_n \|\sigma^d_{\mathrm{epi}_n}(s^\mu_1, \pi(s^\mu_1))\|$$

Substituting into the previous inequality, we obtain

$$|\Delta(s_0)| \leq \gamma L_V \beta^d_n \|\sigma^d_{\mathrm{epi}_n}(s_0, \pi(s_0))\| + \gamma^2 L_V C_d \beta^d_n \mathbb{E}_{\xi_0}[\|\sigma^d_{\mathrm{epi}_n}(s^\mu_1, \pi(s^\mu_1))\|] + \gamma^2 \mathbb{E}_{\xi_0}\mathbb{E}_{\xi_1}[|\Delta(s^\mu_2)|].$$

We repeat this expansion of s^μ_t forever. Notice that the last term vanishes as $t \to \infty$ due to the boundedness of the reward. Therefore,

$$|\Delta(s_0)| = |V_\mu(s_0) - V_*(s_0)| \leq \lambda^d_{n,\mathrm{SG}} \mathbb{E}_{\xi_0}\left[\sum_{t=0}^{\infty} \gamma^t \|\sigma^d_{\mathrm{epi}_n}(s_t, a_t)\| \mid s_0\right]$$

where $\lambda^d_{n,\mathrm{SG}} = \gamma L_V \beta^d_n$. Using equation 31 and taking the expectation over the starting state s_0 yields equation 40.

We can reuse the above procedure but instead add and subtract $\gamma \mathbb{E}_{\xi_0}[V_\mu(g^*(s_0, \pi(s_0)) + \xi_0]$ rather than $\gamma \mathbb{E}_{\xi_0}[V_*(\mu^d_n(s_0, \pi(s_0)) + \xi_0]$. This will result in terms involving $\Delta(s^*_t)$ after repeating the same expansion step, yielding equation 39. □

Theorem D.13 (Infinite-horizon regret bound in settings with subgaussian noise)**.** *Let all conditions and assumptions stated in Theorem 5.8 hold, except that the process noise is a conditionally mean-zero σ_d-subgaussian martingale-difference sequence. Then, the cumulative infinite-horizon regret after N episodes of real environment interaction satisfies, with probability $1-\delta$, the same sublinear bound as presented in Theorem 5.8 up to constants/log factors:*

$$R_{\gamma,N} \leq \mathcal{O}\left(H^3 \sqrt{N}\left(\Gamma^{3/2}_{d,N\log N} + \Gamma^{3/2}_{r,N\log N}\right) + e_{q,N}\right)$$

where $e_{q,N} := \sum_{n=1}^{N} \epsilon_{q,n}$.

Proof. We can see that Lemma D.12 has the same form of Lemma B.7. in Sukhija et al. (2025b). The subgaussian replacement only changes the coefficient of the simulation lemma. In the Gaussian proof, the coefficient $\lambda_n = C_{\max}\frac{\gamma}{1-\gamma}\frac{(1+\sqrt{d_x})\beta_{n-1}}{\sigma}$, while in the subgaussian proof, we instead use $\lambda^d_{n,\mathrm{SG}} = \gamma L_V \beta^d_n$. Since $\beta^d_n = \mathcal{O}(\sqrt{\Gamma_{d,N}})$, the coefficient has the same information-gain order as the original coefficient. Therefore, the regret rate remains the same up to constants/log-factors. □

D.5 Analysis of UCB Objective With Planner Suboptimality

Introducing Planner Sub-optimality. Recall Assumption 5.6, where we assumed that the policy π_n^{ind} induced by the finite-horizon optimization of equation 7 after the n^{th} episode of real-environment interaction coincides with the ideal policy π_n^* induced by exact maximization of equation 7. This assumption effectively treats the planner as an oracle that, under MPC-style re-planning, always selects the globally optimal first action according to the optimistic objective. In practice, however, planning is implemented via approximate optimization over some non-convex objective, hence the final solution action sequence may be suboptimal with respect to the planning objective Sikchi et al. (2021); Argenson & Dulac-Arnold (2021); Jeong et al. (2025). Hence, it is prudent to also consider this imperfection in the theoretical analysis.

We use the formulation introduced in Sikchi et al. (2021), and introduce a *planner sub-optimality* parameter ϵ_p, which upper bounds the gap between the optimistic objective value attained by the ideal MPC-induced policy and that achieved by the implemented planner. Using this, we re-state Assumption 5.6:

Assumption D.14 (Planner suboptimality in the optimistic objective)**.** At the n^{th} episode of real environment interaction, with each episode containing T timesteps, there exist nonnegative numbers $\{\epsilon_{p,n,t}\}_{t=0}^{T-1}$ such that the policy induced by the UCB objective equation 7 satisfies:

$$\eta_{\gamma,n}^{\text{UCB}}(\pi_n^*, \mu_n^d) - \eta_{\gamma,n}^{\text{UCB}}(\pi_n^{\text{ind}}, \mu_n^d) \;\le\; \epsilon_{p,n}^{\gamma}, \qquad \epsilon_{p,n}^{\gamma} := \sum_{t=0}^{T-1} \gamma^t \, \epsilon_{p,n,t}. \tag{41}$$

In particular, if $\epsilon_{p,n,t} \le \epsilon_p$ for all t, then $\epsilon_{p,n}^{\gamma} \le \frac{1-\gamma^T}{1-\gamma}\,\epsilon_p$.

Because MPC re-solves the UCB objective at each timestep and executes only the first action, suboptimality can accumulate across re-planning steps; we therefore quantify a discounted, episode-level planner error $\epsilon_{p,n}^{\gamma}$, where $\epsilon_{p,n,t}$ bounds the one-step optimization error at real environment step t in episode n. This is unlike Sikchi et al. (2021), where the MPC setting is not considered. Ultimately, our definition yields a direct additive degradation in the per-episode regret bound, which we state below.

Lemma D.15 (Single episode regret with planner suboptimality)**.** *Let Assumptions 5.1, 5.2, 5.3, 5.4, and D.14 hold (note: Assumption 5.6 no longer holds), and assume that the learned dynamics and reward are well-calibrated. Consider the definition of r_n in equation 8, and let a_n, b_n be as defined in Lemma D.4, Then, $\forall n > 0$ with probability at least $1-\delta$,*

$$r_n \le (b_n'^2 + b_n' c_n' + 2b_n' + c_n')\,\Sigma_{\gamma,n}^{d}(\pi_n^{\text{ind}}) + (2a_n^2 + a_n c_n' + 2a_n + c_n')\,\Sigma_{\gamma,n}^{r}(\pi_n^{\text{ind}}) + 2\gamma^H \epsilon_{q,n} + \epsilon_{p,n}^{\gamma}, \tag{42}$$

where a_n, b_n', c_n' are as defined in Lemma D.9 and $\epsilon_{p,n}^{\gamma}$ is as defined in Assumption D.14

Proof. We follow the proof of Lemma D.9:

$$\begin{aligned}
r_n &= \eta_\gamma(\pi^*, p^*) - \eta_\gamma(\pi_n, p^*) \\
&\le \eta_{\gamma,n}^{\text{UCB}}(\pi^*, \mu_n^d) - \eta_\gamma(\pi_n^{\text{ind}}, p^*) + \gamma^H \epsilon_{q,n} \\
&= \underbrace{\left(\eta_{\gamma,n}^{\text{UCB}}(\pi^*, \mu_n^d) - \eta_{\gamma,n}^{\text{UCB}}(\pi_n^*, \mu_n^d)\right)}_{\text{(A)}} + \underbrace{\left(\eta_{\gamma,n}^{\text{UCB}}(\pi_n^*, \mu_n^d) - \eta_{\gamma,n}^{\text{UCB}}(\pi_n^{\text{ind}}, \mu_n^d)\right)}_{\text{(B)}} \\
&\quad + \underbrace{\left(\eta_{\gamma,n}^{\text{UCB}}(\pi_n^{\text{ind}}, \mu_n^d) - \eta_\gamma(\pi_n^{\text{ind}}, p^*)\right) + \gamma^H \epsilon_{q,n}}_{\text{(C)}}.
\end{aligned}$$

Since $\pi_n^* \in \arg\max_\pi \eta_{\gamma,n}^{\text{UCB}}(\pi, \mu_n^d)$ by definition, $(\text{A}) \le 0$. Notice that term (B) is precisely the definition of the MPC planner suboptimality in Assumption D.14, so $(\text{B}) \le \epsilon_{p,n}^{\gamma}$. Finally, notice that (C) is equivalent to the original regret bound presented in the proof of Lemma D.9. Putting the terms together, we obtain the regret bound in equation 42. □

Theorem D.16 (Regret bound with planner suboptimality)**.** *Let Assumptions 5.1, 5.2, 5.3, 5.4, and D.14 hold (note: Assumption 5.6 no longer holds), and assume that the learned dynamics and reward are well-calibrated. Let $\epsilon^{\gamma}_{p,n}$ be as defined in Assumption D.14. Then, the cumulative regret after N real environment episodes satisfies, with probability $1-\delta$:*

$$R_{\gamma,N} \;\leq\; \mathcal{O}\Big(H^3\sqrt{N}\left(\Gamma^{3/2}_{d,N\log N}+\Gamma^{3/2}_{r,N\log N}\right)+e_{q,N}\Big)+\sum_{n=1}^{N}\epsilon^{\gamma}_{p,n}, \tag{43}$$

Moreover, if $\epsilon^{\gamma}_{p,n}\leq\epsilon^{\gamma}_{p,n,t}$ for all n,t, then:

$$R_{\gamma,N} \;\leq\; \mathcal{O}\Big(H^3\sqrt{N}\left(\Gamma^{3/2}_{d,N\log N}+\Gamma^{3/2}_{r,N\log N}\right)+e_{q,N}\Big) \;+\; N\cdot\frac{1-\gamma^T}{1-\gamma}\,\epsilon_p \tag{44}$$

Proof. The single episode regret bound from Lemma D.15 can be substituted into the proof procedure for Theorem 5.8 with minimal changes. The planner suboptimality simply introduces an additive term which has no effect on the rest of the proof. □

Planner suboptimality in the finite horizon regret bound It is easily shown that the same additive terms corresponding to the planner suboptimality which appear in equation 43 and equation 44 appear in the finite horizon regret bound in Theorem D.7, contributing linearly.

Discussion of planner suboptimality and regret Evidently, in order for the cumulative regret to remain sublinear, the cumulative planner error should be sublinear rather than linear (as it currently is in Theorem D.16 . If an assumption is made that $\epsilon^{\gamma}_{p,n}\in\mathcal{O}(1/\sqrt{n})$, then $\sum_{n=1}^{N}\epsilon^{\gamma}_{p,n}\in O(\sqrt{N})$ and the overall regret remains sublinear.

The question then is whether such a decay assumption is warranted in practice. Under a fixed planning budget per replanning step (e.g., a constant number of trajectory samples or gradient steps), there is generally no reason to expect $\epsilon_{p,n,t}$ to vanish with n, in which case $\sum_{n=1}^{N}\epsilon^{\gamma}_{p,n}$ is linear and can dominate the regret bound asymptotically. On the other hand, a decaying $\epsilon^{\gamma}_{p,n}$ becomes plausible if planning accuracy improves over time, either by allocating increasing compute to the planner or by reducing the effective optimization difficulty through warm-starting and amortization. Past work has demonstrated that in nonconvex stochastic optimization, many algorithmic guarantees improve monotonically with the number of iterations or gradient evaluations, yielding smaller optimization residuals as compute increases Ghadimi & Lan (2013). Similarly, sampling-based optimizers commonly used in MPC such as the cross-entropy method improve solution quality as the sampling budget increases de Boer et al. (2005). Moreover, even without explicitly increasing the compute budget, which may not always be feasible, it has been shown that methodological innovations such as warm-starting and amortization has been explicitly studied as a mechanism to accelerate nonlinear trajectory optimization by predicting high-quality initial solutions Byravan et al. (2021); Sambharya et al. (2022).

Taken together, these considerations suggest that in practice, conditions for assuming $\epsilon^{\gamma}_{p,n}\in\mathcal{O}(1/\sqrt{n})$ may be a reasonable abstraction in certain planning regimes with the aforementioned computational and methodological optimizations. In particular, practical TD-MPC2-style systems often warm-start planning Hansen et al. (2024a). However, establishing a general, specific decay rate for these methods remain challenging and are an interesting direction for future theoretical work.

E HUMAN ANNOTATION EXPIREMENT

We conducted a real human feedback experiment on the Window Close and Handle Press task. 2 volunteers labeled the queries; each was given the task description and a short practice round before the session began, and no volunteer labeled the same run twice. Annotators were blinded to which method, UBP2 or the MBP baseline, had generated the queries they were labeling. Annotators could indicate no preference when they judged two segments to be equally good or equally bad, rather than being encouraged to always choose one; such labels were treated as discarded.

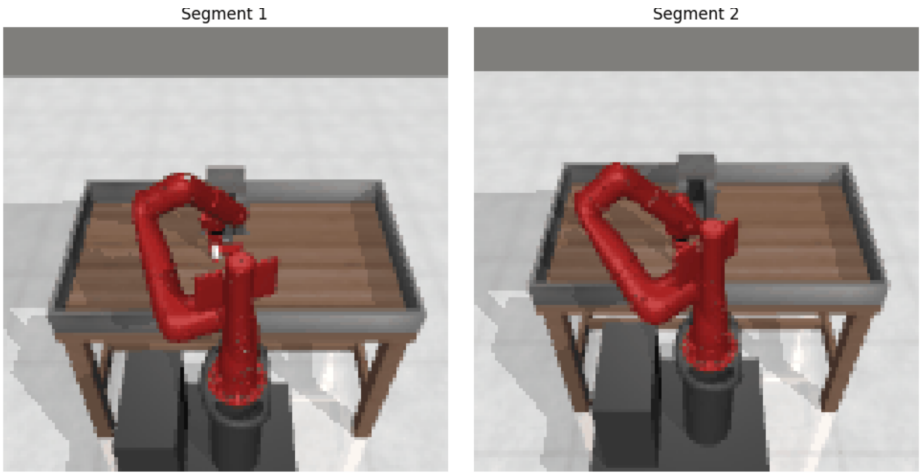


Figure 5: Interface used to collect real human preferences on the Window Close task. Annotators watch two trajectory segments side by side and indicate which one better completes the task, or that they have no preference.