

# 000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 OPTIMISTIC TASK INFERENCE FOR BEHAVIOR FOUNDATION MODELS

Anonymous authors

Paper under double-blind review

## ABSTRACT

Behavior Foundation Models (BFMs) are capable of retrieving high-performing policy for any reward function specified directly at test-time, commonly referred to as zero-shot reinforcement learning (RL). While this is a very efficient process in terms of compute, it can be less so in terms of data: as a standard assumption, BFMs require computing rewards over a non-negligible inference dataset, assuming either access to a functional form of rewards, or significant labeling efforts. To alleviate these limitations, we tackle the problem of task inference purely through interaction with the environment at test-time. We propose OpTI-BFM, an optimistic decision criterion that directly models uncertainty over reward functions and guides BFMs in data collection for task inference. Formally, we provide a regret bound for well-trained BFMs through a direct connection to upper-confidence algorithms for linear bandits. Empirically, we evaluate OpTI-BFM on established zero-shot benchmarks, and observe that it enables successor-features-based BFMs to identify and optimize an unseen reward function in a handful of episodes with minimal compute overhead.

## 1 INTRODUCTION

Zero-shot reinforcement learning (Touati et al., 2023) has gradually gained relevance as a powerful generalization of standard, single-reward RL (Sutton & Barto, 2018; Silver et al., 2021). Zero-shot agents are designed to distill optimal policies for a set of reward functions from unlabeled, offline data (Touati & Ollivier, 2021; Agarwal et al., 2024; Park et al., 2024b; Jajoo et al., 2025). As these methods scale to more complex and broader environments, they are often referred to as Behavior Foundation Models (BFMs) (Park et al., 2024b; Tirinzoni et al., 2025). At their core, the majority of BFMs rely on Universal Successor Features (Ma et al. (2020), USFs). These methods build upon explicit representations of states (*features*): given a set of policies, the expected discounted sum of features along each policy’s trajectory may be estimated completely offline (e.g., through TD learning) (Dayan, 1993; Barreto et al., 2017). As long as a reward function lies within the span of features, zero-shot policy evaluation can be performed through a simple scalar product, which in turn enables zero-shot policy improvement, i.e., each policy is implicitly paired with a reward function and updated towards optimality (Touati & Ollivier, 2021).

Once this unsupervised pre-training phase is complete, USFs yield a set of learned policies. Given a reward function that is linear in the features, its optimal policy is by construction indexed by the linear weights describing the reward function in the basis of features, which can in turn be interpreted as *task embeddings*. The process of finding the optimal policy, which we refer to as *task inference*, thus corresponds to solving a linear regression problem. While this process is remarkably efficient in terms of compute, it retains strict requirements in terms of data: a dataset of labeled (state, reward) pairs needs to be provided. In simple settings, this necessity is not particularly problematic: BFMs have been largely trained in simulations (Touati et al., 2023; Park et al., 2024b; Tirinzoni et al., 2025), which makes generating reward labels for the pre-training dataset, or additional data, particularly convenient. Realistically, however, (i) the pre-training dataset might be unavailable or proprietary and, most importantly, (ii) labeling states with rewards might incur significant costs. For instance, when BFMs are pre-trained directly from pixels, evaluating success from a single image is not a straightforward or cheap operation.

In order to alleviate these issues, we explore an alternative framework for task inference, which instead aims to collect a small amount of data directly during deployment (see Figure 1). This

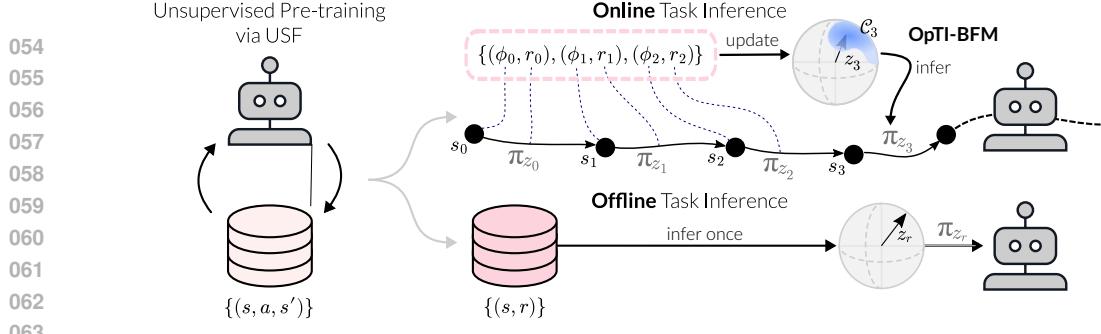


Figure 1: In contrast to the standard offline task inference pipeline for BFM, we explore an alternative online framework: instead of producing a point-estimate of the task embedding from an existing dataset, we actively collect data to build a belief over task embeddings, which results in milder labeling requirements and fast retrieval of a near-optimal policy.

removes the need to access the pre-training dataset (i) and may require fewer labels (ii), as the data can be collected *actively*. To navigate this setting, we propose OptI-BFM, a decision criterion that curates a sequence of task embeddings. Starting from an uninformed prior, OptI-BFM leverages the linear relationship between features and rewards to update its belief over the space of rewards. A BFM conditioned on meaningfully chosen task embeddings can then interact with the environment for a few steps: labeling the states that are visited allows accurate task estimation with minimal labeling efforts. Crucially, if the underlying BFM is well-trained, we show that control reduces to a bandit problem over tasks. Under this assumption, we can provide regret guarantees for OptI-BFM. When considering established zero-shot benchmarks in the Deepmind Control Suite (DMC) (Tassa et al., 2018), we find that OptI-BFM requires only a handful of episodes to correctly identify the task, eventually matching or surpassing the performance that the standard offline reward inference pipeline achieves with significantly more data.

## 2 PRELIMINARIES

**Notation** We model the environment as a reward-free Markov Decision Process (MDP)  $\mathcal{M} = (S, A, P, \mu, \gamma)$  where  $S$  and  $A$  are state and action spaces, respectively,  $P(ds'|s, a)$  is a probability measure describing the likelihood of transitions,  $\mu(ds)$  is a measure describing the initial state distribution, and  $\gamma$  is a discount factor. Given a policy  $\pi : S \rightarrow \Delta(A)$  and a state-action pair  $(s_0, a_0) \in S \times A$  we use  $\mathbb{E}[\cdot | s_0, a_0, \pi]$  to denote expectations w.r.t. state-action sequences  $((s_t, a_t))_{t \geq 0}$  defined by sampling  $a_t \sim \pi(s_t)$  and  $s_{t+1} \sim P(s_t, a_t)$ . For a given reward function  $r : S \rightarrow \mathbb{R}$ , we define the policy's state-action value function as the discounted sum of future rewards  $Q_r^\pi(s_0, a_0) = \sum_{t \geq 0} \gamma^t \mathbb{E}[r(s_t) | s_0, a_0, \pi]$ .

**Successor Features and Behavior Foundation Models** For a specific feature map  $\phi : S \rightarrow \mathbb{R}^d$ , Successor Features (Barreto et al. (2017), SFs) generalize state-action value functions by modeling the expected discounted sum of *features* under a policy  $\pi$ :

$$\psi^\pi(s_0, a_0) = \sum_{t \geq 0} \gamma^t \mathbb{E}[\phi(s_t) | s_0, a_0, \pi]. \quad (1)$$

SFs allow zero-shot policy evaluation for any reward that lies in the span of the features: if  $r(s) = z^\top \phi(s)$  for some  $z \in \mathbb{R}^d$ , the Q-function is a linear function of SFs:

$$Q_r^\pi(s_0, a_0) = \sum_{t \geq 0} \gamma^t \mathbb{E}[r(s_t) | s_0, a_0, \pi] = \sum_{t \geq 0} \gamma^t \mathbb{E}[z^\top \phi(s_t) | s_0, a_0, \pi] = z^\top \psi^\pi(s_0, a_0) \quad (2)$$

A similar structure also holds for value functions:  $V_r^\pi(s) = z^\top \psi^\pi(s)$ , when defining  $\psi^\pi(s) = \mathbb{E}_{a \sim \pi(\cdot | s)} \psi^\pi(s, a)$ . Behavior Foundation Models (BFMs) generally<sup>1</sup> capitalize on the opportunity of evaluating policies for multiple reward functions, by additionally learning a family of parameterized policies  $(\pi_z)_{z \in \mathcal{Z}}$  with respect to all rewards in the span of features  $\phi$  (Borsa et al., 2018; Touati

<sup>1</sup>There are BFMs that are not based on USFs. We refer the reader to Agarwal et al. (2025a) for a comprehensive overview.

108 & Ollivier, 2021; Park et al., 2024b; Agarwal et al., 2025b). Concretely, BFM train a family of  
 109 parameterized policies so that each policy  $\pi_z$  is optimal for the reward function  $r(s) = \phi(s)^\top z$ :  
 110

$$\pi_z(a|s) \in \arg \max_a \psi^{\pi_z}(s, a)^\top z \quad \text{for each } z \in \mathcal{Z}. \quad (3)$$

112 In practice, the set of policies, their SFs, and features  $\phi$  may be represented through function  
 113 approximation:  $\pi_z(a|s) \approx \pi_\xi(a|s, z)$ ,  $\psi^{\pi_z}(s, a) \approx \psi(s, a, z)$ , and  $\phi(s) \approx \phi(s)$ .  $\mathcal{Z}$  is a *low-*  
 114 *dimensional* space, whose elements can be seen as *task embeddings*, as they represent reward  
 115 functions in the feature basis. The low dimensionality of task embeddings is a key component  
 116 enabling efficient task inference in this work.

117 Given a reward function  $r$  at inference, the task embedding  $z_r$  parameterizing the optimal learned  
 118 policy  $\pi_{z_r}$  is found by minimizing the residual between  $r(s)$  and  $\phi(s)^\top z$  (i.e., *projecting*  $r$  onto the  
 119 span of  $\phi$ ). Given a task inference dataset  $\mathcal{D} = (s_i)_{i=1}^N$  this may be solved in closed form:  
 120

$$z_r = \arg \min_z \mathbb{E}_{s \sim \mathcal{D}}[(r(s) - z^\top \phi(s))^2] = \text{Cov}_{\mathcal{D}}(\phi)^{-1} \mathbb{E}_{s \sim \mathcal{D}}[\phi(s)r(s)]. \quad (4)$$

122 where  $\text{Cov}_{\mathcal{D}}(\phi)^{-1} = \mathbb{E}_{s, s' \sim \mathcal{D}}[\phi(s)\phi(s')^\top]$ . This process of mapping from a reward function  $r$  to  
 123 an (approximately) optimal policy  $\pi_{z_r}$  is what makes BFM based on USFs capable of zero-shot  
 124 RL (Touati et al., 2023), i.e., they can produce an optimal policy for a previously unseen reward  
 125 function. When the expectation over  $\mathcal{D}$  is computed exactly, and  $r$  lies in the span of  $\phi$ , then  $\pi_{z_r}$  is  
 126 guaranteed to be the optimal policy (Touati & Ollivier, 2021). However, in practice, the expectation  
 127 is approximated through sampling, which requires (i) the availability of the task inference dataset  $\mathcal{D}$   
 128 (potentially a subset of the pre-training data) and (ii) providing reward labels for each state  $s \in \mathcal{D}$ .  
 129 As this can be an expensive operation, potentially requiring human supervision, we will propose an  
 130 alternative online framework for retrieving  $z_r$ .  
 131

### 3 OPTIMISTIC TASK INFERENCE FOR BEHAVIOR FOUNDATION MODELS

#### 3.1 SETTING: TASK INFERENCE AT TEST-TIME

136 We consider an alternative framework for task inference in BFM, designed to remove the necessity  
 137 for storing pre-training data and, principally, to decrease the required number of reward labels. We  
 138 focus on an online setting, in which the agent can update the task embedding  $z$  during deployment,  
 139 and directly control the collection of the data used to estimate  $z$ . While the choice of  $z$  will be  
 140 uninformed at the beginning, it will ideally be possible to rapidly identify the correct task, and thus  
 141 select the  $z$  that retrieves the optimal policy, i.e., that coincides with the true task embedding.  
 142

143 More formally, we start from a pre-trained USF-based BFM, providing a set of parameterized policies  
 $(\pi_z)_{z \in \mathcal{Z}}$ , as well as SF estimates  $\psi^{\pi_z}$  of features  $\phi$ . While we will now consider finite-horizon  
 144 SF estimates  $\psi^{\pi_z}(s_0) = \sum_{t=0}^{H-1} \gamma^t \mathbb{E}[\phi(s_t) | s_0, \pi_z]$  to streamline the presentation and analysis, we  
 145 remark that the algorithm can be easily instantiated in infinite-horizon settings, as is done in our  
 146 empirical evaluation. The agent interacts with the environment in an episodic setting with horizon  $H$   
 147 and initial state distribution  $\mu_0$ ; instead of directly selecting actions, it will select a task embedding  $z_t$   
 148 at each step  $t$ , and execute an action sampled from the respective policy  $a_t \sim \pi_{z_t}(\cdot | s_t)$ . This action  
 149 will result in observing a new state  $s_{t+1}$ , as well as the reward  $r_t$  of this transition, which constitutes  
 150 the only source of information about the task<sup>2</sup>.

151 We define the discounted return of the  $k$ -th episode as

$$\hat{G}_k = \sum_{t=0}^{H-1} \gamma^t r(s_{kH+t}). \quad (5)$$

152 where  $s_{kH} \sim \mu_0$ ,  $a_t \sim \pi_{z_t}(\cdot | s_t)$  and  $s_{t+1} \sim P(\cdot | s_t, a_t)$ . The goal of the agent is now simply to  
 153 minimize the expected regret over  $n$  episodes  
 154

$$R_n = \mathbb{E} \left[ \sum_{k=0}^{n-1} \hat{G}_k^* - \hat{G}_k \right] \quad (6)$$

155 where  $\hat{G}_k^*$  denotes the discounted return choosing  $z_r$  in each step, and the expectation is w.r.t.  $\mu_0$ ,  
 156 the MDP dynamics, the action distributions, and the choices of the task embeddings. Intuitively, the  
 157

158 <sup>2</sup>We assume that each environment interaction provides a reward label; if the agent can additionally control  
 159 when to request a reward label, more efficient schemes are possible, see Appendix B.2  
 160

162 agent needs to follow a decision rule that maps the observed history of states and rewards to a task  
 163 embedding:  $(s_0, r_0, \dots, s_t, r_t) \rightarrow z_{t+1} \in \mathcal{Z}$ . The selected task embeddings should induce informative  
 164 trajectories with respect to the reward function, while largely avoiding suboptimal behavior. This  
 165 setting is reminiscent of well-developed research directions, namely exploration in the space of  
 166 behavioral priors (Singh et al., 2020) and fast adaptation (Sikchi et al., 2025); however, existing  
 167 methods ignore the underlying structure connecting rewards and task embeddings, which we will  
 168 show may be leveraged to efficiently achieve near-optimal performance.

### 170 3.2 METHOD: OPTI-BFM

172 Our method leverages a core feature of BFM: for well-trained USFs, the expectation of the return  
 173 of the  $k$ -th episode (Eq. (5)) from an initial state  $s_0$  is approximately linear w.r.t. the successor  
 174 features of the policy conditioned on the  $k$ -th task embedding  $z_k$ :  $\psi(s_0, z_k)^\top z_r \approx \mathbb{E}[\hat{G}_k | s_0, \pi_{z_k}]$ ,  
 175 where  $z_r$  is the optimal task embedding, and initially unknown. This simple property has significant  
 176 implication: Policy search reduces to online optimization of a linear function, which has been  
 177 extensively studied in the bandit literature (Dani et al., 2008; Abbasi-Yadkori et al., 2011). Building  
 178 upon these fundamental results, we propose **Optimistic Task Inference for BFM** (OpTI-BFM) in  
 179 order to efficiently explore the space of behaviors while controlling suboptimality.

180 Interestingly, the same approximately linear relationship that connects SFs and returns, also exists  
 181 between features and rewards:  $\phi(s)^\top z_r \approx r(s)$ . Following the latter property, OpTI-BFM keeps  
 182 track of a least-squares estimate of  $z_r$  given the previously observed transitions

$$184 \hat{z}_t = \arg \min_{z \in \mathcal{Z}} \sum_{i=0}^t (r_i - \phi(s_i)^\top z)^2 + \lambda \|z\|_2^2 = \left( \lambda I_d + \sum_{i=0}^t \phi(s_i) \phi(s_i)^\top \right)^{-1} \sum_{i=0}^t \phi(s_i) r_i \quad (7)$$

186 using  $l_2$ -regularization to ensure the inverse exists. Rewriting this as

$$188 \hat{z}_t = V_t^{-1} \sum_{i=0}^t \phi(s_i) r_i, \quad \text{where} \quad V_t = \lambda I_d + \sum_{i=0}^t \phi(s_i) \phi(s_i)^\top \quad (8)$$

190 allows OpTI-BFM to not only track the mean estimate  $\hat{z}_t$ , but also a confidence ellipsoid  $\mathcal{C}_t$  around  
 191  $\hat{z}_t$  that contains the true task embedding  $z_r$  with high probability:

$$192 \mathcal{C}_t = \left\{ z \in \mathbb{R}^d : \|z - \hat{z}_{t-1}\|_{V_{t-1}} \leq \beta_t \right\}, \quad (9)$$

194 where  $\beta_t$  controls the Mahalanobis distance. The estimation of confidence sets allows *optimistic*  
 195 behavior in each step by choosing the task embedding  $z_t$  which is believed to attain the largest return  
 196 among those in the confidence set:

$$198 z_t \in \arg \max_{z \in \mathcal{Z}} \max_{w \in \mathcal{C}_t} w^\top \psi(s_t, z). \quad (10)$$

200 Intuitively, this procedure conditions the BFM on the most "promising" task embedding among those  
 201 that are compatible with rewards observed so far. Note that this algorithm has one crucial difference  
 202 to Upper Confidence Bound (UCB)-based algorithms for linear contextual bandits (Abbasi-Yadkori  
 203 et al., 2011; Dani et al., 2008), as two different contexts are involved: the features  $\phi$ , which are used  
 204 for online regression and for estimating the confidence interval, and the successor features  $\psi$ , which  
 205 are instead used in the acquisition function in Eq. (10). We will discuss that using  $\phi$  for regression  
 206 results in tighter estimates in Appendix A.2.

207 Algorithm 1 instantiates OpTI-BFM for online task inference. We will establish guarantees for  
 208 OpTI-BFM in the next section, and then describe how Eq. 10 may be optimized in practice, or  
 209 avoided altogether with a Thompson Sampling (TS) variant, among others.

### 211 3.3 GUARANTEES

213 Leveraging a direct connection to principled algorithms for linear bandits (Dani et al., 2008; Abbasi-  
 214 Yadkori et al., 2011), we provide regret guarantees for OpTI-BFM. We note that this is a crucial  
 215 property for online task inference, which could otherwise fail to gather informative data, and converge  
 to suboptimal solutions. For simplicity, we study a variant of OpTI-BFM that only updates its decision

---

**216 Algorithm 1** One episode of online task inference with OpTI-BFM
 

---

217 **Require:** BFM with  $\psi^{\pi_z}(s, a)$ ,  $\phi(s)$ , and  $\pi_z(a|s)$ , starting state  $s_0 \sim \mu_0$ , online Least Squares  
 218 estimator  $(\hat{z}_{n-1}, V_{n-1})$  (potentially initialized with past experience)  
 219 **for**  $t = 0, \dots, H-1$  **do**  
 220     Find  $z_t \in \arg \max_{z \in \mathcal{Z}} \max_{w \in \mathcal{C}_{n+t}} w^\top \psi(s_t, z)$      ▷ Optimism w.r.t. cumulative reward.  
 221     Execute action  $a_t \sim \pi_{z_t}(\cdot|s_t)$   
 222     Observe reward  $r_t$ , next state  $s_{t+1}$   
 223     Update  $(\hat{z}_{n+t-1}, V_{n+t-1})$  with  $(\phi(s_t), r_t)$  through Eq. 8     ▷ Update based on reward-feedback.  
 224

---

225 rule at the beginning of each episode, i.e., we have  $z_t = z_{t-1}$  for  $t \notin \{kH\}_{k=0}^\infty$ <sup>3</sup>. This additional  
 226 constraint allows us to leverage results from linear contextual bandit literature.  
 227

228 We can show that OpTI-BFM approaches the performance of the optimal policy  $\pi_{z_r}$  (Eq. (4)) under  
 229 the following assumptions:  
 230

- 231 (A1) Perfect USF for our setting: for every  $(s, a, z) \in \mathcal{S} \times \mathcal{A} \times \mathcal{Z}$  we have  $\psi(s_0, a_0, z) =$   
 232      $\sum_{t=0}^{H-1} \gamma^t \mathbb{E}[\phi(s_t) | \pi_z, s_0, a_0]$  and  $\pi_z(a|s) > 0 \implies a \in \arg \max_{a \in \mathcal{A}} \psi(s, a, z)^\top z$ .
- 233 (A2) Linear Reward:  $r$  is in the span of features  $\phi$  up to i.i.d. mean-zero  $\sigma$ -subgaussian noise  $\eta_t$ ,  
 234     i.e.  $r(s_t) = \phi(s_t)^\top z_r + \eta_t$
- 235 (A3) Optimization Oracle: the OpTI-BFM objective, Eq. (10), can be computed exactly.
- 236 (A4) Bounded norms:  $\|z_r\|_2 \leq S$  and  $\|\phi(\cdot)\|_2 \leq L$  for some  $S > 0$  and  $L > 0$ .

237 Assumptions (A1) and (A2) are instrumental to recovering theoretical guarantees, but we found  
 238 OpTI-BFM to perform well even when they are violated (see Section 5). Note that for a sufficiently  
 239 large horizon  $H$  the mismatch between the finite discounted sum of features we assume here and the  
 240 infinite one we have in practice is negligible: the  $l_2$ -error is by  $L\gamma^H/(1-\gamma)$ . Assumption (A3) may  
 241 be empirically motivated by the efficiency of finding a good approximate solution (see Section 4).  
 242 In this setting, we can show that OpTI-BFM has sublinear regret.  
 243

244 **Proposition 1. (informal)** Under assumptions A1-A4, in an episodic discounted MDP, if OpTI-  
 245 BFM (Algorithm 1) only updates  $z_t$  at the start of each episode, it incurs an expected regret of  
 246  $R_n \leq \tilde{\mathcal{O}}(d\sqrt{n})$ .  
 247

248 *Proof.* We prove a formal version, Proposition 5, in Appendix A. The proof is similar to the standard  
 249 regret bounds for LinUCB/OFUL (Dani et al., 2008; Abbasi-Yadkori et al., 2011) except that the  
 250 confidence interval is updated  $H$ -times per step with features that differ from the context features  
 251 of the bandit.  $\square$   
 252

## 4 PRACTICAL ALGORITHM

254 Having introduced and analyzed OpTI-BFM, we now discuss a practical implementation, and present  
 255 some additional variants<sup>4</sup>.  
 256

257 The main challenge for a practical implementation of OpTI-BFM lies in optimizing the decision  
 258 criterion in Eq. (10), involving two continuous spaces  $\mathcal{Z}$  and  $\mathcal{C}_t$ , and a highly non-linear map  
 259  $z \mapsto \psi(\cdot, z)$ . Fortunately, BFM are pre-trained such that  $w \approx \arg \max_{z \in \mathcal{Z}} w^\top \psi(s_t, z)$ , i.e. the  
 260 optimal policy for a task described by  $w$  is the one conditioned on  $w$  itself. In practice, as training is  
 261 not perfect, we do not strictly rely on this property, and still search  $z$  over  $\mathcal{C}_t$  instead of  $\mathcal{Z}$ :<sup>5</sup>  
 262

$$z_t \in \arg \max_{z \in \mathcal{C}_t} \max_{w \in \mathcal{C}_t} w^\top \psi(s_t, z) \quad (11)$$

263 We note that, as  $\mathcal{C}_t$  shrinks, so does the decision space, and with it the complexity of the optimization  
 264 problem. Finally, as commonly done for linear UCB (Remark 1), we can reformulate the objective as  
 265

$$\arg \max_{z \in \mathcal{C}_t} \max_{w \in \mathcal{C}_t} w^\top \psi(s_t, z) = \arg \max_{z \in \mathcal{C}_t} \psi(s_t, z)^\top \hat{z}_{t-1} + \beta_t \|\psi(s_t, z)\|_{V_{t-1}^{-1}} \quad (12)$$

<sup>3</sup>We provide an empirical comparison to this variant in Section 5.3.

<sup>4</sup>One additional variant is presented in Appendix B.2

<sup>5</sup>We additionally consider a radius of  $2\beta_t$  instead of  $\beta_t$  for this confidence set.

270 which allows us to optimize over just one variable. In practice, we found a simple sampling approach  
 271 to be sufficient to find an approximate solution. We optimize the final objective through random shooting  
 272 with a budget of  $n = 128$  candidates, each of which is evaluated through a forward pass of the suc-  
 273 cessor feature network  $\psi$  in the BFM. Sampling from the ellipsoid  $\mathcal{C}_t$  may be done efficiently by apply-  
 274 ing the push-forward  $\xi \mapsto \hat{z}_{t-1} + V_{t-1}^{-1/2} \beta_t \xi$  to uniform samples from the unit ball (Barthe et al., 2005).  
 275

276 A practical implementation may also benefit from efficient updates to the key parameters:  $\hat{z}_{t-1}$ ,  
 277  $V_{t-1}^{-1/2}$ , and  $V_{t-1}^{-1}$ . We do so by keeping track of the information vector  $\sum_{s=1}^t \phi_t r_t$  and the Cholesky  
 278 factor of  $V_t$ , i.e.  $V_t = R_t^\top R_t$ . This enables updates and/or recomputation of all components in  $\mathcal{O}(d^2)$   
 279 (Gill et al., 1974; Seeger, 2004) making OpTI-BFM very cheap (see Appendix B.1).

280 **Thompson Sampling Variant** The connection between least squares confidence sets and Bayesian  
 281 linear regression is well established in contextual linear bandit literature (Kaufmann et al., 2012;  
 282 Agrawal & Goyal, 2013). Concretely, we may interpret OpTI-BFM’s internal components at time-step  
 283  $t$  as a Gaussian posterior  $\mathcal{N}(\hat{z}_{t-1}, V_{t-1}^{-1})$  over task embeddings that stem from a Bayesian linear  
 284 regression with prior  $\mathcal{N}(0, \frac{1}{\lambda} I_d)$  on the data  $\{(\phi_i/\sigma, r_i/\sigma)\}_{i=0}^{t-1}$  (where  $\sigma$  is a hyper-parameter in  
 285 practice). Intuitively, this can be thought of as starting from a prior on behaviors and then refining  
 286 it over time until it converges to a single behavior. Given this Bayesian interpretation, it is then  
 287 natural to consider a Thompson Sampling (TS) approach, where we simply sample a behavior from  
 288 the posterior, i.e. task embedding  $z_t \sim \mathcal{N}(\hat{z}_t, V_t^{-1})$ , which foregoes the optimization of the UCB  
 289 version. We evaluate this variant extensively in Section 5.  
 290

291 **Non-stationary Rewards Variant** Because of its online nature, OpTI-BFM can potentially adapt  
 292 to reward functions that change over time. To this end, we consider a variant leveraging a simple  
 293 idea from non-stationary bandit literature: weighing old data points less than new ones in the least  
 294 squares estimator (Russac et al., 2020). Concretely, we introduce a new hyper-parameter  $0 < \rho \leq 1$   
 295 and weight data from past time-step  $s$  at time-step  $t$  with weight  $\rho^{t-s}$ . We evaluate OpTI-BFM in a  
 296 setting with non-stationary rewards in Section 5.4.  
 297

## 298 5 EXPERIMENTS

300 The empirical evaluation is divided in several subsections, each of which will address a specific  
 301 question, as their titles suggest. In the following, we first detail some evaluation choices.  
 302

303 **Environments** To evaluate performance of various adaptation/inference algorithms we consider  
 304 the environments Walker, Cheetah, and Quadruped from the established ExORL (Yarats et al., 2022)  
 305 benchmark, with four different tasks (i.e. reward functions) each. We describe full experimental  
 306 protocols in Appendix E. In all figures, error-bars and shaded regions represent min-max-intervals  
 307 over 3 training seeds.  
 308

309 **Methods** We choose Forward-Backward (FB) framework (Touati & Ollivier, 2021) as a state-of-  
 310 the-art BFM. We adhere to the standard training and evaluation protocol for FB (Touati et al., 2023),  
 311 and described it in detail in Appendix D<sup>6</sup>. We apply OpTI-BFM on top of this BFM, as well as a  
 312 Thompson-Sampling variant, OpTI-BFM-TS. We further consider LoLA (Sikchi et al., 2025), an  
 313 approach based on policy search that was originally introduced for fast adaptation. In compliance  
 314 with our online task inference setting, we initialize LoLA with an uninformed choice of  $z$ , and  
 315 estimate on-policy returns in the standard episodic fashion, without privileged resets. Besides these  
 316 three learning algorithms, we evaluate “Random” and “Oracle” baselines that serve as a lower and  
 317 an upper bound on possible performance respectively. The former executes a random embedding in  
 318 each step, i.e.  $z_t \sim \text{Unif}(\mathcal{Z})$ , whereas the latter executes the optimal policy  $\pi_{z_r}$ , where  $z_r$  is attained  
 319 by solving the linear regression problem in Eq. (4), assuming privileged access to labeled data or the  
 320 reward function. We follow standard practice and approximate  $z_r$  with a large budget of 50k labeled  
 321 samples from the pre-training dataset in practice (Touati et al., 2023; Agarwal et al., 2025b). If not  
 322 indicated otherwise, “relative performance” is relative to the Oracle performance.  
 323

<sup>6</sup>Our code is available on the anonymous website [sites.google.com/view/opti-bfm](https://sites.google.com/view/opti-bfm).

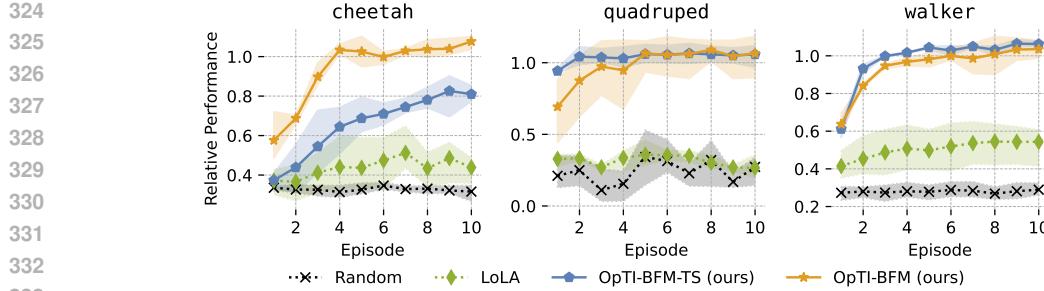


Figure 2: Mean relative performance over 10 episodes of interaction in DMC. OpTI-BFM recovers Oracle performance in 5 episodes. We report per-task absolute performance in Fig. 14 in Appendix C.

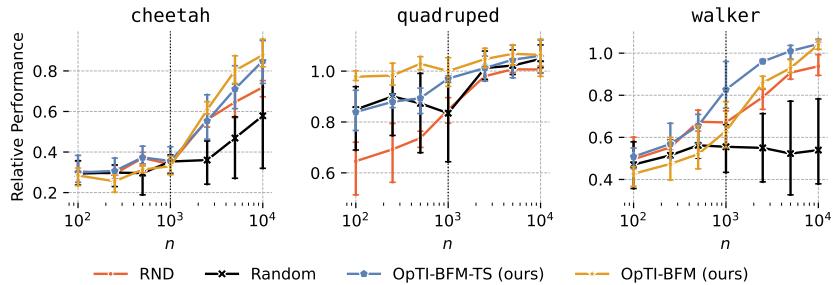


Figure 3: Relative performance after different # of environment interactions. OpTI-BFM is consistently among the top performers for all environments and time-steps. We show per task performance in Fig. 15 in Appendix C.

### 5.1 HOW DOES OPTI-BFM COMPARE TO OTHER ONLINE TASK INFERENCE METHODS?

We first evaluate all methods in the online task inference setting described in Section 3.1, and report episodic returns (Eq. (5)) in Fig. 2. We find that OpTI-BFM recovers Oracle performance on all tasks within 5 episodes (5k environment steps) of interaction. We observe a significant gap between the optimistic strategy of OpTI-BFM and its TS variant in Cheetah. Nevertheless, TS remains a promising approach, as it avoids any optimization problem, and simply samples task embeddings from its current belief. Finally, we find that LoLA, which ignores the linear structure of the problem and performs blackbox policy search, makes slower progress which is better visible over 50 episodes in Fig. 16 in Appendix C. This result is consistent with existing ablations initializing LoLA to a random task embedding (see Sikchi et al. (2025), Figure 5), as is the case in our setting.

### 5.2 IS THE DATA COLLECTED ACTIVELY BY OPTI-BFM INFORMATIVE?

While the previous evaluations focus on episodic returns, or equivalently regret minimization, we now evaluate the quality of the inferred task embeddings, e.g., in the case of OpTI-BFM, how well does  $\pi_{\hat{z}_n}$  perform? In practice, to ablate away any bias in  $l_2$ -regularized estimators, we compute the task embedding  $z_n$  by minimizing the squared error in Eq. (4) over the dataset of the first  $n$  observed transitions of each method, and evaluate the corresponding policy  $\pi_{z_n}$  in the same environment and task. We compare with two baseline data sources: (i) random trajectories from RND (Burda et al., 2018), which represents a task-agnostic exploration approach, and (ii) the first  $n$  samples from our Random baseline, which rolls out a random policy learned by the BFM. Figure 3 shows the average relative performance of  $\pi_{z_n}$  in each environment. We find that the data from an actively exploring source, i.e. OpTI-BFM and RND, outperforms the passive Random approach. Furthermore, we can see that OpTI-BFM and its TS variant tend to be more data-efficient than RND, which can be traced back to task-awareness.

### 5.3 ARE FREQUENT UPDATES IMPORTANT?

The regret bound provided in Section 3.3 assumes a version of OpTI-BFM that only updates the task embedding it executes at the start of each episode. In this section we aim to experimentally quantify this gap in theory and practice by evaluating this episodic version of OpTI-BFM. Fig. 4 shows that changing the latent on an episode-level leads to slower improvement than changing every step. The episodic versions do reach equal performance eventually as shown in Fig. 16 in

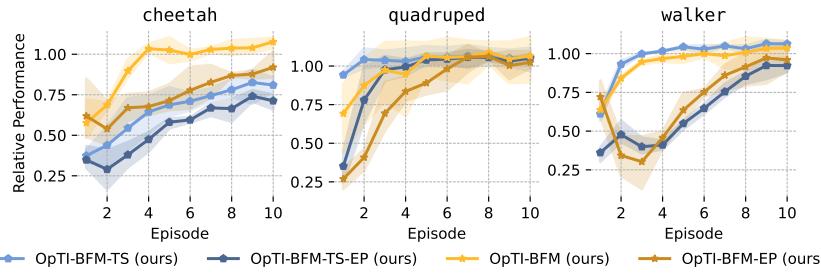


Figure 4: Relative performance of our methods, and their variants that keep task embeddings fixed for each episode (-EP). Updating the task embedding during the episode leads to faster convergence. For longer episode evaluations see Fig. 16 in Appendix C.

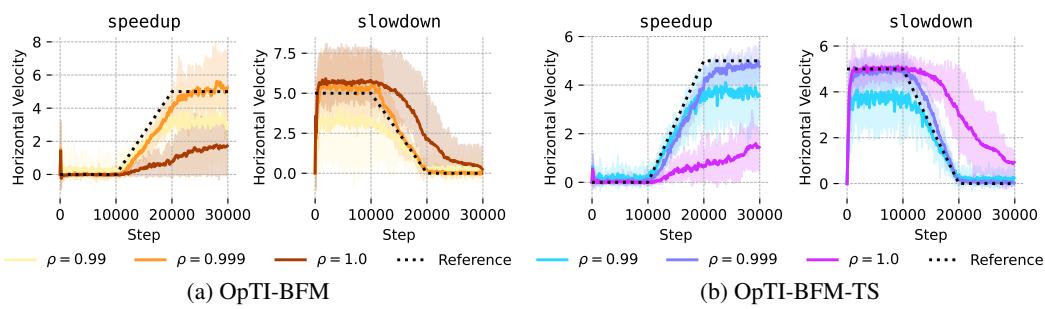


Figure 5: Horizontal velocity of OpTI-BFM in our custom velocity tracking tasks in DMC Walker for different decay rates  $\rho$ . OpTI-BFM can adapt to non-stationary reward functions when decaying the weight of old observations.

Appendix C. Intuitively, this faster improvement may be explained by the fact that the default version of OpTI-BFM can adjust faster to new information.

#### 5.4 CAN OPTI-BFM ADAPT TO NON-STATIONARY REWARDS?

It is conceivable that OpTI-BFM can adapt to reward functions that are changing over time because of its online nature. To investigate whether this is possible, we introduce two new tasks for the Walker environment that closely resemble the walk and run tasks but change the velocity target over time: speedup and slowdown increase and decrease the velocity target respectively after an initial “burn-in” phase. We evaluate OpTI-BFM in a single, 30k-step, episode in Figure 5, which shows that a direct application of OpTI-BFM ( $\rho = 1.0$ ) struggles to adapt to the changing reward function after converging to a fixed one in the first 10k steps. As soon as  $\rho$  is reduced, OpTI-BFM tracks the velocity target more accurately. Moreover, we observe that, if  $\rho$  is too small, the uncertainty is not reduced quickly enough, and the agent can converge to suboptimal behavior.

#### 5.5 CAN OPTI-BFM WARM-START FROM LABELED DATA?

In settings that provide a small amount of labeled data  $\mathcal{D} = \{(s_i, r_i)\}_{i=1}^n$  from the beginning (Sikchi et al., 2025), OpTI-BFM may be warm-started by updating the least squares estimator  $n$ -times with the given data. As shown in Fig. 6, when doing so on high-quality i.i.d. data from the training dataset, the performance of OpTI-BFM quickly improves, showing that OpTI-BFM can additionally be seen as an extension of the traditional task inference of BFM.

## 6 RELATED WORK

**Behavior Foundation Models** This work builds upon USFs (Dayan, 1993; Barreto et al., 2017; Ma et al., 2020), and specifically on BFM that utilize SFs to train a parameterized policy and the corresponding parameterized SFs. A common approach for learning SF-based BFM is to approximate of successor measures (Blier et al., 2021; Touati & Ollivier, 2021; Agarwal et al., 2025b), but other options spanning from spectral decomposition of random visitations (Wu et al., 2018) to implicit value learning (Park et al., 2024b) also exist. In this regard, we refer the reader to Agarwal

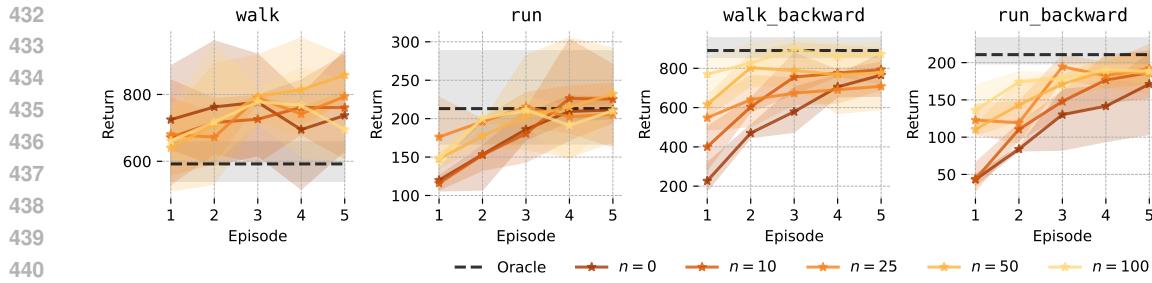


Figure 6: Return of OpTI-BFM in DMC Cheetah when warm-starting with  $n$  i.i.d. labeled states from the training dataset. Initial performance increases quickly as  $n$  grows. We report full results in Fig. 17 in Appendix C.

et al. (2025a) for a broader overview of BFM and unsupervised RL. Among these approaches, the Forward Backward (FB) framework (Touati & Ollivier, 2021) has risen to prominence, as it achieves state-of-the-art performance in standard continuous control tasks (Touati et al., 2023), while being scalable to humanoid control (Tirinzoni et al., 2025). Recent work has explored various applications of FB’s zero-shot capabilities, such as imitation learning (Pirotta et al., 2023), epistemic exploration (Urpí et al., 2025), and constrained RL (Hugessen et al., 2025). Among related works, our method is most closely related to Sikchi et al. (2025), which however focuses on a different problem: fine-tuning a zero-shot policy, which is already initialized through standard offline task inference. On the other hand, our goal is to find a strong policy purely through environment interaction, without relying on preexisting labeled data. The two approaches might nevertheless be combined.

**Reward Learning** The task inference goal of OpTI-BFM is fundamentally connected to the reward learning problem in RL. Among the many instantiations, rewards may be inferred from scalar evaluations (Knox & Stone, 2009; MacGlashan et al., 2017), preferences (Fürnkranz et al., 2012; Christiano et al., 2017), or from other types of feedback (Jeon et al., 2020). A powerful extension to reward learning explores active strategies for query selection, largely adopting a Bayesian perspective (Biyik et al., 2020; Wilde et al., 2020). While none of these works are aimed at BFM, several are close in spirit: for instance, Lindner et al. (2021) propose an information-directed method to achieve a similar goal to ours: inferring the task in a way which does not necessarily reduce the model error, but quickly produces an optimal policy.

**Linear Bandits** Linear Contextual Bandits are a well-studied problem in bandit literature (Abe & Long, 1999; Dani et al., 2008; Abbasi-Yadkori et al., 2011; Kaufmann et al., 2012; Agrawal & Goyal, 2013). This is influential to this work in two ways. First, OpTI-BFM is inspired by UCB methods in this setting (Dani et al., 2008; Abbasi-Yadkori et al., 2011). Second, we utilize guarantees and ideas from this literature (Lattimore & Szepesvári, 2020) to produce a regret bound for OpTI-BFM in an episodic setting.

## 7 CONCLUSION

By drawing a connection between practical work in scalable Behavior Foundation Models and rigorous algorithms for linear optimization, we proposed an algorithm that may efficiently infer the task at hand and retrieve a well-performing policy in high-dimensional, complex environments. By minimizing the number of reward evaluations necessary for task inference, this algorithm can enable BFM to be applied beyond domains in which rewards are readily available, for instance when learning directly from pixels.

OpTI-BFM is designed to perform task inference in a minimal number of episodes: this is mainly possible as the search space is minimal. While updating the task embedding alone enables great sample efficiency, fine-tuning additional components of the BFM may provide even better performance in the long run (Sikchi et al., 2025). Moreover, our theoretical guarantees only cover slower, episode-level updates: extending these results to (empirically stronger) per-step represents an important direction for future formal works.

486 While OptBFM builds upon linearity between features and rewards, it does not make any significant  
 487 assumption on the structure of the feature space  $\mathcal{Z}$ : understanding the properties of its elements both  
 488 formally and practically constitutes an important avenue for future work. BFM have demonstrated  
 489 scalability up to complex domains, while maintaining structured properties (linearity!) in an em-  
 490 bedding space: we believe that this is a ripe field for application of more theoretically principled  
 491 approaches, which may in turn be impactful beyond regular domains.

## 493 REFERENCES

495 Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved Algorithms for Linear Stochastic Bandits.  
 496 In *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

497 Abe, N. and Long, P. Associative reinforcement learning using linear probabilistic concepts. 06 1999.

498 Agarwal, S., Sikchi, H., Stone, P., and Zhang, A. Proto Successor Measure: Representing the Space  
 499 of All Possible Solutions of Reinforcement Learning, November 2024.

500 Agarwal, S., Chuck, C., Sikchi, H., Hu, J., Rudolph, M., Niekum, S., Stone, P., and Zhang, A.  
 501 A Unified Framework for Unsupervised Reinforcement Learning Algorithms. In *Workshop on*  
 502 *Reinforcement Learning Beyond Rewards @ Reinforcement Learning Conference 2025*, July 2025a.

503 Agarwal, S., Sikchi, H., Stone, P., and Zhang, A. Proto Successor Measure: Representing the  
 504 Behavior Space of an RL Agent, March 2025b.

505 Agrawal, S. and Goyal, N. Thompson Sampling for Contextual Bandits with Linear Payoffs. In  
 506 *Proceedings of the 30th International Conference on Machine Learning*, pp. 127–135. PMLR,  
 507 May 2013.

508 Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H. P., and Silver, D. Successor  
 509 Features for Transfer in Reinforcement Learning. In *Advances in Neural Information Processing*  
 510 *Systems*, volume 30. Curran Associates, Inc., 2017.

511 Barthe, F., Guedon, O., Mendelson, S., and Naor, A. A probabilistic approach to the geometry  
 512 of the  $\ell_p^n$ -ball. *The Annals of Probability*, 33(2), March 2005. ISSN 0091-1798. doi:  
 513 10.1214/009117904000000874.

514 Biyik, E., Huynh, N., Kochenderfer, M. J., and Sadigh, D. Active preference-based gaussian process  
 515 regression for reward learning. *arXiv preprint arXiv:2005.02575*, 2020.

516 Blier, L., Tallec, C., and Ollivier, Y. Learning Successor States and Goal-Dependent Values: A  
 517 Mathematical Viewpoint, January 2021.

518 Bogunovic, I. and Krause, A. [Misspecified gaussian process bandit optimization](#), 2021.

519 Borsa, D., Barreto, A., Quan, J., Mankowitz, D., Munos, R., van Hasselt, H., Silver, D., and Schaul,  
 520 T. Universal Successor Features Approximators, December 2018.

521 Burda, Y., Edwards, H., Storkey, A., and Klimov, O. Exploration by Random Network Distillation,  
 522 October 2018.

523 Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. Deep reinforcement  
 524 learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

525 Dani, V., Hayes, T. P., and Kakade, S. M. Stochastic Linear Optimization under Bandit Feedback.  
 526 *21st Annual Conference on Learning Theory*, 2008.

527 Dayan, P. Improving Generalization for Temporal Difference Learning: The Successor Representation.  
 528 *Neural Computation*, 5(4):613–624, July 1993. ISSN 0899-7667. doi: 10.1162/neco.1993.5.4.613.

529 Fürnkranz, J., Hüllermeier, E., Cheng, W., and Park, S.-H. Preference-based reinforcement learning:  
 530 a formal framework and a policy iteration algorithm. *Machine learning*, 89(1):123–156, 2012.

531 Ghosh, A., Chowdhury, S. R., and Gopalan, A. [Misspecified linear bandits](#), 2017.

540 Gill, P. E., Golub, G. H., Murray, W., and Saunders, M. A. Methods for modifying ma-  
 541 trix factorizations. In *Mathematics of Computation*, volume 28, pp. 505–535, 1974. doi:  
 542 10.1090/S0025-5718-1974-0343558-6.

543 Hugessen, A., Wiltzer, H., Neary, C., Zhang, A., and Berseth, G. Zero-Shot Constraint Satisfaction  
 544 with Forward- Backward Representations. In *Workshop on Reinforcement Learning Beyond*  
 545 *Rewards @ Reinforcement Learning Conference 2025*, July 2025.

546 Jajoo, P., Sikchi, H., Agarwal, S., Zhang, A., Niekum, S., and White, M. Regularized Latent  
 547 Dynamics Prediction is a Strong Baseline For Behavioral Foundation Models. In *Workshop on*  
 548 *Reinforcement Learning Beyond Rewards @ Reinforcement Learning Conference 2025*, July 2025.

549 Jeon, H. J., Milli, S., and Dragan, A. Reward-rational (implicit) choice: A unifying formalism for  
 550 reward learning. *Advances in Neural Information Processing Systems*, 33:4415–4426, 2020.

551 Kaufmann, E., Cappé, O., and Garivier, A. On Bayesian Upper Confidence Bounds for Bandit  
 552 Problems. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and*  
 553 *Statistics*, pp. 592–600. PMLR, March 2012.

554 Kiefer, J. and Wolfowitz, J. The equivalence of two extremum problems. *Canadian Journal of*  
 555 *Mathematics*, 12:363 – 366, 1960.

556 Kingma, D. P. and Ba, J. [Adam: A method for stochastic optimization](#), 2017.

557 Knox, W. B. and Stone, P. Interactively shaping agents via human reinforcement: The tamer  
 558 framework. In *Proceedings of the fifth international conference on Knowledge capture*, pp. 9–16,  
 559 2009.

560 Lattimore, T. and Szepesvári, C. *Bandit Algorithms*. Cambridge University Press, 1 edition, July  
 561 2020. ISBN 978-1-108-57140-1 978-1-108-48682-8. doi: 10.1017/9781108571401.

562 Lindner, D., Turchetta, M., Tschiatschek, S., Ciosek, K., and Krause, A. Information directed reward  
 563 learning for reinforcement learning. *Advances in Neural Information Processing Systems*, 34:  
 564 3850–3862, 2021.

565 Ma, C., Ashley, D. R., Wen, J., and Bengio, Y. Universal Successor Features for Transfer Reinforce-  
 566 ment Learning, January 2020.

567 MacGlashan, J., Ho, M. K., Loftin, R., Peng, B., Wang, G., Roberts, D. L., Taylor, M. E., and Littman,  
 568 M. L. Interactive learning from policy-dependent human feedback. In *International conference on*  
 569 *machine learning*, pp. 2285–2294. PMLR, 2017.

570 Park, S., Frans, K., Eysenbach, B., and Levine, S. OGBench: Benchmarking Offline Goal-Conditioned  
 571 RL, October 2024a.

572 Park, S., Kreiman, T., and Levine, S. Foundation Policies with Hilbert Representations, May 2024b.

573 Pirotta, M., Tirinzoni, A., Touati, A., Lazaric, A., and Ollivier, Y. Fast Imitation via Behavior  
 574 Foundation Models. In *The Twelfth International Conference on Learning Representations*,  
 575 October 2023.

576 Russac, Y., Vernade, C., and Cappé, O. Weighted Linear Bandits for Non-Stationary Environments,  
 577 March 2020.

578 Seeger, M. Low Rank Updates for the Cholesky Decomposition. 2004.

579 Sikchi, H., Tirinzoni, A., Touati, A., Xu, Y., Kanervisto, A., Niekum, S., Zhang, A., Lazaric, A., and  
 580 Pirotta, M. Fast Adaptation with Behavioral Foundation Models, April 2025.

581 Silver, D., Singh, S., Precup, D., and Sutton, R. S. Reward is enough. *Artificial Intelligence*, 299:  
 582 103535, October 2021. ISSN 0004-3702. doi: 10.1016/j.artint.2021.103535.

583 Singh, A., Liu, H., Zhou, G., Yu, A., Rhinehart, N., and Levine, S. Parrot: Data-driven behavioral  
 584 priors for reinforcement learning. *arXiv preprint arXiv:2011.10024*, 2020.

594 Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. The MIT Press, second  
595 edition, 2018.

596

597 Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D. d. L., Budden, D., Abdolmaleki, A.,  
598 Merel, J., Lefrancq, A., Lillicrap, T., and Riedmiller, M. DeepMind Control Suite, January 2018.

599

600 Tirinzoni, A., Touati, A., Farebrother, J., Guzek, M., Kanervisto, A., Xu, Y., Lazaric, A., and Pirotta,  
601 M. Zero-Shot Whole-Body Humanoid Control via Behavioral Foundation Models, April 2025.

602

603 Touati, A. and Ollivier, Y. Learning One Representation to Optimize All Rewards, October 2021.

604

605 Touati, A., Rapin, J., and Ollivier, Y. Does Zero-Shot Reinforcement Learning Exist?, March 2023.

606

607 Tucker, A. D., Biddulph, C., Wang, C., and Joachims, T. Bandits with costly reward observations. In  
608 *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, pp. 2147–  
609 2156. PMLR, July 2023.

610

611 Urpí, N. A., Vlastelica, M., Martius, G., and Coros, S. Epistemically-guided forward-backward  
612 exploration, July 2025.

613

614 Wilde, N., Kulić, D., and Smith, S. L. Active preference learning using maximum regret. In *2020  
615 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10952–10959.  
616 IEEE, 2020.

617

618 Wu, Y., Tucker, G., and Nachum, O. The laplacian in rl: Learning representations with efficient  
619 approximations. *arXiv preprint arXiv:1810.04586*, 2018.

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648 **A PROOFS**  
 649

650 This section describes the proof for Proposition 1. It closely follows Chapters 19 and 20 from  
 651 Lattimore & Szepesvári (2020). We proceed as follows:  
 652

- 653 • We reintroduce the setting with some more convenient notation in Appendix A.1.  
 654
- 655 • We show that running linear regression on feature-reward pairs is at least as efficient as  
 656 running linear regression on SF-return pairs (Appendix A.2)  
 657
- 658 • We cite a result that grants the optimal choice of  $\beta$ s in Appendix A.3.  
 659
- 660 • We proceed with the standard linear contextual bandit regret bound proof with confidence  
 661 sets based on feature-reward pairs (Appendix A.4).

662 **A.1 SETTING**  
 663

664 Throughout this section we will use the double subscript  $k, t$  to denote the  $t$ -th time-step in the  $k$   
 665 episode. Since we will analyze a version of OpTI-BFM that only switches task embedding at the  
 666 start of each episode, and since we know that the best task embedding is always  $z_r$  (Eq. 4), we define  
 667  $z_1, \dots, z_n$  be the chosen task embeddings for  $n$  episodes. Let  $\phi_{k,t} = \phi(s_{k,t})$  be the feature observed  
 668 in step  $t$  of episode  $k$ . Note that this quantity is a random variable; throughout this section we use a  
 669 simple expectation  $\mathbb{E}[\cdot]$  to indicate the expectation over initial states, action distributions of policies,  
 670 MDP dynamics, and decisions of the algorithm. We will still use  $\mathbb{E}[\cdot | \pi, s]$  to denote the expectation  
 671 over rolling out policy  $\pi$  from state  $s$ .  
 672

673 **A.1.1 ASSUMPTIONS**  
 674

675 For convenience, we repeat our assumptions here with the new notation  
 676

677 (A1) Perfect USF for our setting: for every  $(s, a, z) \in \mathcal{S} \times \mathcal{A} \times \mathcal{Z}$  we have  
 678

$$\psi(s_0, a_0, z) = \sum_{t=0}^{H-1} \gamma^t \mathbb{E}[\phi(s_t) | \pi_z, s_0, a_0] \quad (13)$$

679 and  
 680

$$\pi_z(a|s) > 0 \implies a \in \arg \max_{a \in \mathcal{A}} \psi(s, a, z)^\top z. \quad (14)$$

681 (A2) Linear Reward:  $r$  is in the span of features  $\phi$  up to i.i.d. mean-zero  $\sigma$ -subgaussian noise  $\eta_{k,t}$ ,  
 682 i.e.  
 683

$$r_{k,t} = \phi(s_{k,t})^\top z_r + \eta_{k,t} \quad (15)$$

684 (A3) Optimization Oracle: the OpTI-BFM objective in Eq. (10) can be computed exactly.  
 685

686 (A4) Bounded norms:  $\|z_r\|_2 \leq S$  and  $\|\phi(\cdot)\|_2 \leq L$  for some  $S > 0$  and  $L > 0$ .  
 687

688 **A.1.2 REGRET**  
 689

690 We also repeat the definitions of discounted return of the  $k$ -th episode (Eq. (5)) and expected regret  
 691 (Eq. (6)) for consistent notation.  
 692

$$\hat{G}_k = \sum_{t=0}^{H-1} \gamma^t r_{k,t} \quad (16)$$

$$R_n = \mathbb{E} \left[ \sum_{k=1}^n \hat{G}_k^* - \hat{G}_k \right], \quad (17)$$

693 where  $\hat{G}_k^*$  is the discounted return achieved by  $\pi_{z_r}$  which is optimal for the the reward  $r(\cdot) = \phi(\cdot)^\top z_r$   
 694 under A1. We now also define the expected discounted return of episode  $k$  for task embedding  $z$  as  
 695

$$G_{k,z} = \mathbb{E} \left[ \hat{G}_k \mid \pi_z, s_{k,0} \right]. \quad (18)$$

Notice that this implies that  $\mathbb{E}[\hat{G}_k^*] = G_{k,z_r}$  and that  $R_n = \mathbb{E}[\sum_{k=1}^n G_{k,z^*} - G_{k,z_k}]$ , the latter of which is the term we want to bound at the end of this section. We may now show that the expected discounted return is linear in the SFs:

$$G_{k,z} = \mathbb{E} \left[ \hat{G}_k \mid \pi_z, s_{k,0} \right] \quad (19)$$

$$= \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t r_{k,t} \mid \pi_z, s_{k,0} \right] \quad (20)$$

$$= \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \langle z_r, \phi(s_{k,t}) \rangle \mid \pi_z, s_{k,0} \right] \quad (21)$$

$$= \left\langle z_r, \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \phi(s_{k,t}) \mid \pi_z, s_{k,0} \right] \right\rangle \quad (22)$$

$$\stackrel{A1}{=} \langle z_r, \psi(s_{k,0}, z) \rangle \quad (23)$$

Thus, our setting resembles a contextual bandit with context  $s_0 \sim \mu_0$ , action space  $\mathcal{Z}$  and an expected payoff that is linear in  $\psi(s_0, z)$ , with the one notable difference being that more information is available on the pay-off feedback: it is not available in terms of return  $\hat{G}_k$  alone, but additionally as a sequence of features-reward pairs  $(\phi_{k,t}, r_{k,t})_{t=0}^{H-1}$ .

### A.1.3 OPTI-BFM

Let us finally reintroduce the least squares estimator with the double subscript notation:

$$\hat{z}_n = V_n^{-1} \sum_{k=1}^n \sum_{t=0}^{H-1} \phi_{k,t} r_{k,t} \quad \text{where} \quad V_n = \lambda I + \sum_{k=1}^n \sum_{t=0}^{H-1} \phi_{k,t} \phi_{k,t}^\top. \quad (24)$$

From this we then redefine the confidence sets at the start of episode  $k$  as

$$\mathcal{C}_k = \{z \in \mathbb{R}^d : \|z - \hat{z}_{k-1}\|_{V_{k-1}} \leq \beta_k\}. \quad (25)$$

Defining the UCB operator as

$$\text{UCB}_k(z) = \max_{w \in \mathcal{C}_k} \langle \psi(s_{k,0}, z), w \rangle, \quad (26)$$

then Opti-BFM picks  $z_k = \arg \max_{\mathcal{Z}} \text{UCB}_k(z)$  for the  $k$ -th episode, which we assume can be attained exactly in A3.

## A.2 RETURN-LEVEL VS. REWARD-LEVEL FEEDBACK

In this section we show a few properties conveying the intuitive fact that, if  $z_r$  is estimated from richer, reward-level feedback  $(\phi_{k,t}, r_{k,t})$ , confidence sets may be tighter than those produced by return-level feedback  $(\sum_{t=0}^{H-1} \gamma^t \phi_{k,t}, \sum_{t=0}^{H-1} \gamma^t r_{k,t})$ , which can be thought of as an aggregation of  $H$  datapoints of the former (up to expectation over dynamics).

For this analysis we define the *empirical* SF of episode  $k$  as

$$\tilde{\psi}_k = \sum_{t=0}^{H-1} \gamma^t \phi_{k,t} \quad \text{and} \quad W_n = \lambda I + \sum_{k=1}^n \tilde{\psi}_k \tilde{\psi}_k^\top. \quad (27)$$

Note that we have  $\mathbb{E} \left[ \tilde{\psi}_k \mid \pi_{z_k}, s_{k,0} \right] = \psi(s_{k,0}, z_k)$ . We further define

$$A_k = \sum_{t=0}^{H-1} \phi_{k,t} \phi_{k,t}^\top, \quad (28)$$

so that we have  $V_n = \lambda I + \sum_{k=1}^n A_k$ . Finally, the following constant <sup>7</sup> will become useful later:

$$c_H = \sum_{t=0}^{H-1} \gamma^{2t} = \frac{1 - \gamma^{2H}}{1 - \gamma^2}. \quad (29)$$

We now show that the precision matrices obtained through reward-level feedback ( $V_n$ ) grow at least as fast as those obtained through return-level feedback ( $W_n$ ) over the course of one episode.

<sup>7</sup>Note that  $c_H \geq 1$  for  $H > 0$ .

756 **Proposition 2.** *We have, in Loewner order,*

$$757 \quad 758 \quad 759 \quad 760 \quad 761 \quad 762 \quad 763 \quad 764 \quad 765 \quad 766 \quad 767 \quad 768 \quad 769 \quad 770 \quad 771 \quad 772 \quad 773 \quad 774 \quad 775 \quad 776 \quad 777 \quad 778 \quad 779 \quad 780 \quad 781 \quad 782 \quad 783 \quad 784 \quad 785 \quad 786 \quad 787 \quad 788 \quad 789 \quad 790 \quad 791 \quad 792 \quad 793 \quad 794 \quad 795 \quad 796 \quad 797 \quad 798 \quad 799 \quad 800 \quad 801 \quad 802 \quad 803 \quad 804 \quad 805 \quad 806 \quad 807 \quad 808 \quad 809 \quad \tilde{\psi}_k \tilde{\psi}_k^\top \preceq c_H A_k. \quad (30)$$

From this, it immediately follows  $V_n$  grows at least as fast as  $W_n$  over all episodes.

**Proposition 3.** *We have, in Loewner order,*

$$V_n \succcurlyeq \frac{1}{c_H} W_n \quad (31)$$

*Proof.* Using Proposition 2, we know that

$$A_k \succcurlyeq \frac{1}{c_H} \tilde{\psi}_k \tilde{\psi}_k^\top. \quad (32)$$

Summing over episodes on both sides, we get

$$\sum_{k=1}^n A_k \succcurlyeq \frac{1}{c_H} \sum_{k=1}^n \tilde{\psi}_k \tilde{\psi}_k^\top. \quad (33)$$

By adding  $\lambda I$  on both sides, we can write

$$\lambda I + \sum_{k=1}^n A_k \succcurlyeq \lambda I + \frac{1}{c_H} \sum_{k=1}^n \tilde{\psi}_k \tilde{\psi}_k^\top. \quad (34)$$

On the left, we recognize

$$\lambda I + \sum_{k=1}^n A_k = V_n, \quad (35)$$

while on the right we have

$$\lambda I + \frac{1}{c_H} \sum_{k=1}^n \tilde{\psi}_k \tilde{\psi}_k^\top = \frac{1}{c_H} \left( \lambda I + \sum_{k=1}^n \tilde{\psi}_k \tilde{\psi}_k^\top \right) + \left( 1 - \frac{1}{c_H} \right) \lambda I = \frac{1}{c_H} W_n + \left( 1 - \frac{1}{c_H} \right) \lambda I,$$

which in turn yields

$$V_n \succcurlyeq \frac{1}{c_H} W_n + \left( 1 - \frac{1}{c_H} \right) \lambda I \succcurlyeq \frac{1}{c_H} W_n, \quad (36)$$

where the last step follows from the fact that  $c_H \geq 1$ .  $\square$

We now derive a useful property from this result, that will later help us to apply the well-known elliptical potential Lemma. Intuitively it shows that reward-level feedback leads to confidence ellipsoids at least as tight as those of return-level feedback.

**Proposition 4.** *We have for any  $x \in \mathbb{R}^d$*

$$\|x\|_{V_n^{-1}} \leq \sqrt{c_H} \|x\|_{W_n^{-1}} \quad (37)$$

*Proof.* By Proposition 3,

$$V_n \succcurlyeq \frac{1}{c_H} W_n, \quad (38)$$

where both matrices are positive definite by construction. As the inverse thus reverses order,

$$V_n^{-1} \preceq c_H W_n^{-1}. \quad (39)$$

As a consequence, one can show that, for any  $x$ ,

$$\|x\|_{V_n^{-1}} = \sqrt{x^\top V_n^{-1} x} \leq \sqrt{c_H x^\top W_n^{-1} x} = \sqrt{c_H} \|x\|_{W_n^{-1}}. \quad (40)$$

$\square$

810 A.3 CONFIDENCE ELLIPSOIDS  
811

812 We will now take the chance to cite a standard result for optimal choices of radii  $\beta_k$  which  
813 guarantee that the true parameter  $z_r$  is contained in confidence sets (Eq. (9)) in each time-step with  
814 high probability.

815 **Theorem 1.** (Adapted from [Abbasi-Yadkori et al. \(2011\)](#) Theorem 2)

816 Given confidence sets  $\mathcal{C}_k$  as defined in Eq. (25), and if rewards are in the span of  $\phi$  up to  $\sigma$ -subgaussian  
817 zero-mean noise (A2), then for any  $\delta \in (0, 1)$  the following holds:  $\Pr[\exists k \in \mathbb{N}^+ | z_r \notin \mathcal{C}_k] \leq \delta$  with

$$818 \beta_k = \sqrt{\lambda}S + \sigma \sqrt{\log\left(\frac{\det V_{k-1}}{\lambda^d}\right) + 2\log\left(\frac{1}{\delta}\right)} \quad (41)$$

819 This is the classic uniform-in-time self-normalized bound based on supermartingales. Notice that the  
820 theorem makes no assumptions about the action selection process nor the process that controls the  
821 state-changes, except that they are measurable in the natural filtration of the corresponding time-step,  
822 which is the case for our decision rule (OpTI-BFM) and our episodic MDP setting. We remark that,  
823 since by definition  $V_t \succcurlyeq V_s$  for all  $t \geq s$ , we have that  $\beta_t \geq \beta_s$  as well.

824 A.4 REGRET BOUND  
825

826 We can now prove a formal version of Proposition 1.

827 **Proposition 5.** Under assumptions A1-A4, with choices for  $\beta_k$  as in Theorem 1, if OpTI-BFM is  
828 applied at the beginning of each episode and selects the task embedding  $z_k = \arg \max_{\mathcal{Z}} UCB_k(z)$   
829 for the  $k$ -th episode, it incurs an expected regret of  $\tilde{\mathcal{O}}(d\sqrt{n})$ .

830 *Proof.* We start off by investigating the event  $\mathcal{E}$  that  $z_r \in \mathcal{C}_k \forall k \in \{1, \dots, n\}$ , i.e.,  $z_r$  is in all  
831 confidence sets, which occurs with probability at least  $1 - \delta$  according to Theorem 1.

832 Since we will refer to both empirical and "expected" SFs ( $\tilde{\psi}$  and  $\psi$ , respectively), we introduce  
833 the natural filtration  $\mathcal{F}_{k-1}$  generated by the history up to the start of episode  $k$ , in particular,  
834  $s_{k,0}, z_k, V_{k-1}, W_{k-1}$  are  $\mathcal{F}_{k-1}$ -measurable. Intuitively, conditioning on  $\mathcal{F}_{k-1}$  allows us to limit  
835 randomness to within the current episode. Crucial for us will be,

$$836 \psi(s_{k,0}, z_k) = \mathbb{E} \left[ \tilde{\psi}_k \mid \mathcal{F}_{k-1} \right] \quad (42)$$

837 We will proceed as in the standard regret bound for linear contextual bandits ([Lattimore & Szepesvári,  
838 2020](#)) Let  $\tilde{z}_k = \arg \max_{w \in \mathcal{C}_k} \langle \psi(s_{k,0}, z_k), w \rangle$  be the task embedding that maximizes the UCB. We  
839 then have that

$$840 \langle \psi(s_{k,0}, z_r), z_r \rangle \leq UCB_k(z_r) \leq UCB_k(z_k) = \langle \psi(s_{k,0}, z_k), \tilde{z}_k \rangle. \quad (43)$$

841 We can therefore bound the instantaneous regret  $e_k$  by

$$842 e_k = G_{k,z_r} - G_{k,z_k} \quad (44)$$

$$843 = \langle \psi(s_{k,0}, z_r), z_r \rangle - \langle \psi(s_{k,0}, z_k), z_r \rangle \quad (45)$$

$$844 \leq \langle \psi(s_{k,0}, z_k), \tilde{z}_k \rangle - \langle \psi(s_{k,0}, z_k), z_r \rangle \quad (46)$$

$$845 = \langle \psi(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle. \quad (47)$$

846 Conditioning on previous events ( $\mathcal{F}_{k-1}$ ), and applying the Cauchy-Schwarz inequality pointwise, we  
847 have

$$848 \mathbb{E}[e_k \mid \mathcal{F}_{k-1}] \leq \mathbb{E}[\langle \psi(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle \mid \mathcal{F}_{k-1}] \quad (48)$$

$$849 = \mathbb{E} \left[ \langle \tilde{\psi}_k, \tilde{z}_k - z_r \rangle \mid \mathcal{F}_{k-1} \right] \quad (49)$$

$$850 \leq \mathbb{E} \left[ \left\| \tilde{\psi}_k \right\|_{V_{k-1}^{-1}} \left\| \tilde{z}_k - z_r \right\|_{V_{k-1}} \mid \mathcal{F}_{k-1} \right]. \quad (50)$$

851 Since we are in event  $\mathcal{E}$  we have  $z_r \in \mathcal{C}_k$ , furthermore,  $\tilde{z}_k \in \mathcal{C}_k$  by definition. So, using the definition  
852 of the confidence set Eq. (25), we have

$$853 \left\| \tilde{z}_k - z_r \right\|_{V_{k-1}} \leq 2\beta_k \leq 2\beta_n. \quad (51)$$

This is where this proof diverges from the standard regret bound: note that  $V_{k-1}$  is a sum of features  $\phi$  and not empirical SFs  $\tilde{\psi}$ , making immediate application of the elliptical potential lemma non-trivial. We can however recall the comparison of least square estimation with reward-feedback and return-feedback from Appendix A.2. In particular, one can apply Proposition 4 to upper-bound the  $V$ -norm with the  $W$ -norm that depends on the empirical SFs:

$$\|\tilde{\psi}_k\|_{V_{k-1}^{-1}} \leq \sqrt{c_H} \|\tilde{\psi}_k\|_{W_{k-1}^{-1}}. \quad (52)$$

Combining these two bounds results in

$$\mathbb{E}[e_k | \mathcal{F}_{k-1}] \leq \mathbb{E} \left[ 2\beta_n \sqrt{c_H} \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \mid \mathcal{F}_{k-1} \right]. \quad (53)$$

Recalling that, by assumption A4,  $\|\tilde{\psi}_k\|_2 \leq \frac{L(1-\gamma^H)}{1-\gamma}$  and

$$V_{k-1} = \lambda I + \sum_{i=1}^k \sum_{t=0}^{H-1} \phi_{i,t} \phi_{i,t}^\top \succcurlyeq \lambda I \quad (54)$$

so  $\|u\| \leq \lambda^{-1/2} \|u\|_{V_{k-1}}$ , we can write that, pointwise for all  $k$ ,

$$\left| \langle \tilde{\psi}_k, \tilde{z}_k - z_r \rangle \right| \leq \min \left\{ \frac{2\beta_n}{\sqrt{\lambda}} \|\tilde{\psi}_k\|, 2\beta_n \sqrt{c_H} \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \right\} \quad (55)$$

$$\leq 2\beta_n \min \left\{ \underbrace{\frac{L(1-\gamma^H)}{\sqrt{\lambda}(1-\gamma)}}_{=:B}, \sqrt{c_H} \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \right\}. \quad (56)$$

Combining the two bounds inside the expectation (as they hold pointwise), we obtain

$$\mathbb{E}[e_k | \mathcal{F}_{k-1}] \leq \mathbb{E} \left[ 2\beta_n \min \left\{ B, \sqrt{c_H} \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \right\} \mid \mathcal{F}_{k-1} \right]. \quad (57)$$

Using the fact that  $\min\{a, bx\} \leq \max\{a, b\} \min\{1, x\}$  for  $a, b > 0$ , we can then take some constants out of the expectation:

$$\mathbb{E}[e_k | \mathcal{F}_{k-1}] \leq \underbrace{\max\{B, \sqrt{c_H}\}}_{=: \alpha} \mathbb{E} \left[ 2\beta_n \min \left\{ 1, \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \right\} \mid \mathcal{F}_{k-1} \right]. \quad (58)$$

We now sum over episodes. To do this, we use the tower rule on  $e_k$ , i.e.,  $\mathbb{E}[e_k] = \mathbb{E}[\mathbb{E}[e_k | \mathcal{F}_{k-1}]]$ .

$$R_n = \mathbb{E} \left[ \sum_{k=1}^n e_k \right] = \sum_{k=1}^n \mathbb{E}[e_k] = \sum_{k=1}^n \mathbb{E}[\mathbb{E}[e_k | \mathcal{F}_{k-1}]] \quad (59)$$

Applying our (pointwise) bound on  $\mathbb{E}[e_k | \mathcal{F}_{k-1}]$  and the tower rule again, we get.

$$R_n = \sum_{k=1}^n \mathbb{E}[\mathbb{E}[e_k | \mathcal{F}_{k-1}]] \leq 2\alpha \sum_{k=1}^n \mathbb{E} \left[ \mathbb{E} \left[ \beta_n \min \left\{ 1, \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \right\} \mid \mathcal{F}_{k-1} \right] \right] \quad (60)$$

$$= 2\alpha \sum_{k=1}^n \mathbb{E} \left[ \beta_n \min \left\{ 1, \|\tilde{\psi}_k\|_{W_{k-1}^{-1}} \right\} \right]. \quad (61)$$

Using linearity of expectation and then applying the Cauchy-Schwarz inequality over episodes (pointwise), we can prepare the sum to the application of the elliptical potential Lemma on  $(\tilde{\psi}_k)_{k=1}^n$

918 and  $(W_k)_{k=1}^n$  (pathwise):  
 919

$$920 \quad R_n \leq 2\alpha \mathbb{E} \left[ \beta_n \sum_{k=1}^n \min \left\{ 1, \left\| \tilde{\psi}_k \right\|_{W_{k-1}^{-1}} \right\} \right] \quad (62)$$

$$923 \quad \leq 2\alpha \mathbb{E} \left[ \beta_n \sqrt{n \cdot \sum_{k=1}^n \min \left\{ 1, \left\| \tilde{\psi}_k \right\|_{W_{k-1}^{-1}}^2 \right\}} \right] \quad (63)$$

$$926 \quad \leq 2\alpha \mathbb{E} \left[ \beta_n \sqrt{2n \cdot \log \left( \frac{\det W_n}{\lambda^d} \right)} \right]. \quad \text{Lemma 1} \quad (64)$$

929 Inserting our choice for  $\beta_n$  we obtain  
 930

$$931 \quad R_n \leq 2\alpha \mathbb{E} \left[ \left( \sqrt{\lambda}S + \sigma \left( \sqrt{\log \left( \frac{\det V_{n-1}}{\lambda^d} \right) + 2\log(1/\delta)} \right) \right) \sqrt{2n \cdot \log \left( \frac{\det W_n}{\lambda^d} \right)} \right] \quad (65)$$

934 We recall that, so far, we investigated the expected regret conditional on the event  $\mathcal{E}$  that  $z_r \in \mathcal{C}_k \forall k \in \{1, \dots, n\}$ , which occurs with probability at least  $1 - \delta$  according to Theorem 1. We now choose  $\delta = 1/n$ . Using Proposition 6 to upper-bound the determinants (pointwise), and recalling the fact that  $\left\| \tilde{\psi}_k \right\|_2 \leq \frac{L(1-\gamma^H)}{1-\gamma}$  (by A4), we obtain

$$940 \quad R_n \leq 2\alpha \left( \sqrt{\lambda}S + \sigma \left( \sqrt{d \log \left( \frac{d\lambda + nHL^2}{d\lambda} \right) + 2\log n} \right) \right) \sqrt{2nd \log \left( \frac{d\lambda + n \frac{L^2(1-\gamma^H)^2}{(1-\gamma)^2}}{d\lambda} \right)} \\ 944 \quad \leq \tilde{\mathcal{O}}(d\sqrt{n}). \quad (66)$$

946 We finally investigate the complementary event  $\mathcal{E}^C$ . We remark that the discounted return in each  
 947 episode can be bounded by constants using the Cauchy-Schwarz inequality:

$$949 \quad |G_{k,z}| = \left| \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \phi_{k,t}^\top z_r \mid \pi_{z_k}, s_{k,0} \right] \right| \leq LS \sum_{t=0}^{H-1} \gamma^t \leq \frac{LS}{1-\gamma}. \quad (67)$$

952 It thus holds that  $R_n \leq n \cdot \frac{2LS}{1-\gamma}$  under  $\mathcal{E}^C$ .  
 953

954 To combine the analysis for both events, let  $\hat{R}_n = \sum_{k=1}^n e_k$  be the empirical regret. By the law of  
 955 total probability, we have that

$$956 \quad R_n = \mathbb{E}[\hat{R}_n] = \Pr[\mathcal{E}] \cdot \mathbb{E} \left[ \hat{R}_n \mid \mathcal{E} \right] + \Pr[\mathcal{E}^C] \cdot \mathbb{E} \left[ \hat{R}_n \mid \mathcal{E}^C \right] \quad (68)$$

$$958 \quad \leq 1 \cdot \tilde{\mathcal{O}}(d\sqrt{n}) + \frac{1}{n} \cdot n \cdot \frac{2LS}{1-\gamma} \quad (69)$$

$$960 \quad \leq \tilde{\mathcal{O}}(d\sqrt{n}), \quad (70)$$

962 which concludes the proof.  $\square$   
 963

## 964 A.5 REGRET BOUND UNDER WEAKER ASSUMPTIONS

966 Our empirical evaluations are performed in a setting which violates two assumptions described in  
 967 A.1.1: perfect USF estimation (A1) and linearity of rewards (A2). This section provides a study of the  
 968 algorithm's behavior under these violations taking some ideas from existing theory of misspecified  
 969 bandits (Ghosh et al., 2017; Bogunovic & Krause, 2021; Lattimore & Szepesvári, 2020).

971 We start by replacing A1 and A2 by weaker assumptions that quantify the mismatch between  $\phi$  and  
 $\psi$  and between  $r$  and its projection  $z_r^\top \psi$  respectively:

972 (A1') For every  $(s, a, z) \in \mathcal{S} \times \mathcal{A} \times \mathcal{Z}$  we have  
 973  
 974

$$975 \quad 976 \quad \left\| \psi(s, a, z) - \sum_{t=0}^{H-1} \gamma^t \mathbb{E} [\phi(s_t) \mid \pi_z, s, a] \right\|_2 \leq \xi \quad (71)$$

977  
 978  
 979 and  
 980  
 981

$$982 \quad \pi_z(a|s) > 0 \implies a \in \arg \max_{a \in \mathcal{A}} \psi(s, a, z)^\top z. \quad (72)$$

983  
 984  
 985

986 (A2') For reward  $r$  we have for each state  $s_{k,t}$   
 987  
 988

$$989 \quad 990 \quad \mathbb{E} [\|r(s_{k,t}) - \phi(s_{k,t})^\top z_r\|] \leq \zeta \quad (73)$$

991  
 992 where the expectation is over i.i.d. mean-zero  $\sigma$ -subgaussian noise  $\eta_{k,t}$ .  
 993

994 We may thus write  $\psi$  as a biased version of the true  $\psi^*$   
 995  
 996

$$997 \quad \psi(s, a, z) + \Delta_\psi(s, a, z) = \psi^*(s, a, z), \quad (74)$$

998  
 999

1000 and, similarly, rewrite rewards as  
 1001  
 1002

$$1003 \quad r_{k,t} = \phi(s_{k,t})^\top z_r + \Delta_r(s_{k,t}) + \eta_{k,t}. \quad (75)$$

1004  
 1005

1006 We can then establish a uniform-in-time bound as in Theorem 1 by inflating  $\beta$  to account for the  
 1007 newly introduced bias in the reward. Considering the distance between the current mean estimate  
 1008  $\hat{z}_{t-1}$  and  $z_r$ , we can decompose this into  
 1009

$$1011 \quad 1012 \quad \|z_r - \hat{z}_{t-1}\|_{V_{t-1}} = \left\| z_r - \sum_{i=1}^{t-1} \phi_i \phi_i^\top z_r - V_{t-1}^{-1} \sum_{i=1}^{t-1} \phi_i \eta_i - V_{t-1}^{-1} \sum_{i=1}^{t-1} \phi_i \Delta_r(s_i) \right\|_{V_{t-1}} \quad (76)$$

1013  
 1014

$$1015 \quad 1016 \quad = \left\| z_r - V_{t-1} z_r + \lambda z_r - V_{t-1}^{-1} \sum_{i=1}^{t-1} \phi_i \eta_i - V_{t-1}^{-1} \sum_{i=1}^{t-1} \phi_i \Delta_r(s_i) \right\|_{V_{t-1}} \quad (77)$$

1017  
 1018

$$1019 \quad = \left\| z_r - z_r - V_{t-1}^{-1} \left( \lambda z_r + \sum_{i=1}^{t-1} \phi_i \eta_i + \sum_{i=1}^{t-1} \phi_i \Delta_r(s_i) \right) \right\|_{V_{t-1}} \quad (78)$$

1020  
 1021

$$1022 \quad = \left\| \lambda z_r + \sum_{i=1}^{t-1} \phi_i \eta_i + \sum_{i=1}^{t-1} \phi_i \Delta_r(s_i) \right\|_{V_{t-1}^{-1}} \quad (79)$$

1023  
 1024

$$1025 \quad \leq \left\| \lambda z_r + \sum_{i=1}^{t-1} \phi_i \eta_i \right\|_{V_{t-1}^{-1}} + \left\| \sum_{i=1}^{t-1} \phi_i \Delta_r(s_i) \right\|_{V_{t-1}^{-1}} \quad (80)$$

1026 The first term is covered by Theorem 1, while the new bias term can be bounded through similar  
 1027 techniques to those used in the main proof:  
 1028

$$1029 \quad \left\| \sum_{i=1}^{t-1} \phi_i \Delta_r(s_i) \right\|_{V_{t-1}^{-1}} \leq \sum_{i=1}^{t-1} |\Delta_r(s_i)| \|\phi_i\|_{V_{t-1}^{-1}} \quad \| \cdot \|_{V_{t-1}^{-1}} \text{ properties} \quad (81)$$

$$1030 \quad \leq \zeta \sum_{i=1}^{t-1} \|\phi_i\|_{V_{t-1}^{-1}} \quad \text{A2'} \quad (82)$$

$$1031 \quad \leq \zeta \sum_{i=1}^{t-1} \|\phi_i\|_{V_{i-1}^{-1}} \quad V_{i-1} \preceq V_{t-1} \quad (83)$$

$$1032 \quad \leq \zeta \sqrt{t \sum_{i=1}^{t-1} \|\phi_i\|_{V_{i-1}^{-1}}^2} \quad \text{Cauchy-Schwarz} \quad (84)$$

$$1033 \quad \leq \zeta \sqrt{t \sum_{i=1}^{t-1} \min\left\{\frac{L^2}{\lambda}, \|\phi_i\|_{V_{i-1}^{-1}}^2\right\}} \quad (85)$$

$$1034 \quad \leq \frac{L\zeta}{\sqrt{\lambda}} \sqrt{t \sum_{i=1}^{t-1} \min\{1, \|\phi_i\|_{V_{i-1}^{-1}}^2\}} \quad (86)$$

$$1035 \quad \leq \frac{\sqrt{2}L\zeta}{\sqrt{\lambda}} \sqrt{t \log\left(\frac{\det V_t}{\det V_0}\right)} \quad \text{Lemma 1} \quad (87)$$

1036 As a result, we can establish a uniform-in-time guarantee if  $\beta$  is inflated as  
 1037

$$1038 \quad \beta'_t = \beta_t + \frac{\sqrt{2}L\zeta}{\sqrt{\lambda}} \sqrt{t \log\left(\frac{\det V_t}{\det V_0}\right)}. \quad (88)$$

1039 Continuing to build the regret bound, we first notice that the mismatch also impacts the discounted  
 1040 return:  
 1041

$$1042 \quad G_{k,z} = \mathbb{E} \left[ \hat{G}_k \mid \pi_z, s_{k,0} \right] \quad (89)$$

$$1043 \quad = \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t r_{k,t} \mid \pi_z, s_{k,0} \right] \quad (90)$$

$$1044 \quad \stackrel{A2'}{=} \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \langle z_r, \phi(s_{k,t}) \rangle + \Delta_r(s_{k,t}) \mid \pi_z, s_{k,0} \right] \quad (91)$$

$$1045 \quad = \left\langle z_r, \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \phi(s_{k,t}) \mid \pi_z, s_{k,0} \right] \right\rangle + \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \Delta_r(s_{k,t}) \mid \pi_z, s_{k,0} \right] \quad (92)$$

$$1046 \quad = \langle z_r, \psi^*(s_{k,0}, z) \rangle + \mathbb{E} \left[ \sum_{t=0}^{H-1} \gamma^t \Delta_r(s_{k,t}) \mid \pi_z, s_{k,0} \right] \quad (93)$$

$$1047 \quad \stackrel{A2'}{\leq} \langle z_r, \psi^*(s_{k,0}, z) \rangle + \frac{\zeta}{1-\gamma} \quad (94)$$

1048 We can then proceed along the main proof in Appendix A.4 as normal, substituting  $\beta'$  for  $\beta$  throughout.  
 1049 Given the suboptimal  $\psi$  Eq. (43) now becomes  
 1050

$$1051 \quad \langle \psi^*(s_{k,0}, z_r), z_r \rangle \leq \text{UCB}_k(z_r) \leq \text{UCB}_k(z_k) = \langle \psi^*(s_{k,0}, z_k), \tilde{z}_k \rangle. \quad (95)$$

1080 Then, in the bound on the instantaneous regret (Eq. (44)), we get an added term.  
 1081

$$1082 e_k = G_{k,z_r} - G_{k,z_k} \quad (96)$$

$$1083 \leq \langle \psi^*(s_{k,0}, z_r), z_r \rangle - \langle \psi^*(s_{k,0}, z_k), z_r \rangle + \frac{2\zeta}{1-\gamma} \quad (97)$$

$$1084 \leq \langle \psi^*(s_{k,0}, z_k), \tilde{z}_k \rangle - \langle \psi^*(s_{k,0}, z_k), z_r \rangle + \frac{2\zeta}{1-\gamma} \quad (98)$$

$$1085 = \langle \psi^*(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle + \frac{2\zeta}{1-\gamma} \quad (99)$$

$$1086 \stackrel{A2'}{=} \langle \psi(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle + \langle \Delta_\psi(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle + \frac{2\zeta}{1-\gamma} \quad (100)$$

$$1087 \leq \langle \psi(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle + \|\Delta_\psi(s_{k,0}, z_k)\|_{V_{k-1}^{-1}} \|\tilde{z}_k - z_r\|_{V_{k-1}} + \frac{2\zeta}{1-\gamma} \quad \text{Cauchy-Schwarz}$$

$$1088 \leq \langle \psi(s_{k,0}, z_k), \tilde{z}_k - z_r \rangle + \frac{\xi}{\sqrt{\lambda}} \cdot 2\beta'_k + \frac{2\zeta}{1-\gamma}. \quad (101)$$

1093 Where we used

$$1094 \|x\|_{V_{k-1}^{-1}} = \sqrt{x^\top V_{k-1}^{-1} x} \leq \sqrt{\lambda_{\max}(V_{k-1}^{-1}) x^\top x} \leq \frac{1}{\sqrt{\lambda}} \|x\|_2 \quad (102)$$

1096 in the last step.<sup>8</sup>

1097 We can treat the added term  $2\beta'_k \frac{\xi}{\sqrt{\lambda}} + \frac{2\zeta}{1-\gamma}$  separately throughout, bounding it by

$$1098 2\beta'_n n \frac{\xi}{\sqrt{\lambda}} + \frac{n\zeta}{1-\gamma} \quad (103)$$

1101 after summing over the episodes. As a result, inserting our new  $\beta'_n$  in the regret bound, we obtain

$$1102 R_n \leq 2\alpha \left( \sqrt{\lambda}S + \sigma \sqrt{d \log \left( \frac{d\lambda + nHL^2}{d\lambda} \right) + 2 \log n} + \underbrace{\frac{\sqrt{2}L\zeta}{\sqrt{\lambda}} \sqrt{ndH \log \left( \frac{d\lambda + nHL^2}{d\lambda} \right)}}_{\text{new}} \right) \\ 1103 \times \left( \sqrt{2nd \log \left( \frac{d\lambda + n \frac{L^2(1-\gamma^H)^2}{(1-\gamma)^2}}{d\lambda} \right)} + \underbrace{\frac{2\xi n}{\sqrt{\lambda}}}_{\text{new}} \right) \\ 1104 + \underbrace{\frac{n\zeta}{1-\gamma}}_{\text{new}} \quad (104)$$

1114 Because of the new term appearing in  $\beta'$  and the final constant term, the regret bound is no longer  
 1115 sublinear, but instead

$$1116 R_n \leq \tilde{\mathcal{O}}(d\sqrt{n}) + \zeta \tilde{\mathcal{O}}(nd) + \xi \tilde{\mathcal{O}}(n\sqrt{dn}) \quad (105)$$

1117 which is also the final regret when incorporating the event  $\mathcal{E}^C$ . This bound indicates that, while the  
 1118 confidence set may shrink initially, the two mismatches may introduce an irreducible uncertainty in  
 1119 the size of the respective mismatches that cannot be resolved. Crucially, the suboptimality incurred  
 1120 is directly connected to the degree of mismatch considered in assumptions (A1') and (A2').

## 1123 A.6 USEFUL PROPERTIES

### 1125 A.6.1 DETERMINANTS

1127 We list a few useful properties that are used in the proofs above

1128 **Proposition 6.** Let  $\phi_1, \dots, \phi_n$  be a sequence of vectors with  $\|\phi_t\| \leq L$  for all  $t = 1, \dots, n$ , and  
 1129  $V_t = \lambda I + \sum_{s=1}^t \phi_s \phi_s^\top$ .

$$1130 \log \left( \frac{\det V_n}{\lambda^d} \right) \leq d \log \left( \frac{d\lambda + nL^2}{d\lambda} \right) \quad (106)$$

1133 <sup>8</sup>This approximation of  $\lambda_{\max}(V_{k-1}^{-1}) \leq 1/\lambda$  is rather crude and could be improved by investigating how the  
 1134 eigenvalues of  $V_k$  evolve over time.

1134 *Proof.* First, note that  
 1135

$$1136 \det(\lambda I) = \lambda^d \text{ and } \text{Tr}(\lambda I) = d\lambda \quad (107)$$

1138 For  $V_n$  we have, using the AM-GM inequality,

$$1139 \det V_n = \prod_i \lambda_i \leq \left( \frac{1}{d} \text{Tr } V_n \right)^d \leq \left( \frac{\text{Tr } \lambda I + nL^2}{d} \right)^d \quad (108)$$

1142 because  $\|\phi\| \leq L$ . Thus we have  
 1143

$$1144 \log \left( \frac{\det V_n}{\lambda^d} \right) \leq d \log \left( \frac{d\lambda + nL^2}{d\lambda} \right) \quad (109)$$

1147  $\square$

#### 1149 A.6.2 ELLIPTICAL POTENTIAL LEMMA

1151 We here state a standard result, commonly referred to as the elliptical potential lemma, for completeness.  
 1152

1153 **Lemma 1.** Assume  $V_0 \succ 0$  be positive definite and let  $\phi_1, \dots, \phi_n$  be a sequence of vectors with  
 1154  $\|\phi_t\| \leq L$  for all  $t = 1, \dots, n$ , and  $V_t = V_0 + \sum_{s=1}^t \phi_s \phi_s^\top$ .

$$1155 \sum_{t=1}^n \min\{1, \|\phi_t\|_{V_{t-1}^{-1}}^2\} \leq 2 \log \left( \frac{\det V_n}{\det V_0} \right) \quad (110)$$

1160 *Proof.* First, note that we have for  $u \geq 0$   
 1161

$$1162 \min\{1, u\} \leq 2 \log(1 + u) \quad (111)$$

1163 and use it to get  
 1164

$$1165 \sum_{t=1}^n \min\{1, \|\phi_t\|_{V_{t-1}^{-1}}^2\} \leq 2 \sum_{t=1}^n \log \left( 1 + \|\phi_t\|_{V_{t-1}^{-1}}^2 \right) \quad (112)$$

1169 Then, notice that for  $t \geq 1$  we have  
 1170

$$1171 V_t = V_{t-1} + \phi_t \phi_t^\top = V_{t-1}^{1/2} \left( I + V_{t-1}^{-1/2} \phi_t \phi_t^\top V_{t-1}^{-1/2} \right) V_{t-1}^{1/2} \quad (113)$$

1173 so, taking the determinant on both sides we get  
 1174

$$1175 \det V_t = \det V_{t-1} \cdot \det \left( I + V_{t-1}^{-1/2} \phi_t \phi_t^\top V_{t-1}^{-1/2} \right) = \det V_{t-1} \left( 1 + \|\phi_t\|_{V_{t-1}^{-1}}^2 \right) \quad (114)$$

1177 where we used  $\det(A + B) = \det A \cdot \det B$  and  $\det(I + uu^\top) = 1 + \|u\|^2$ . Telescoping the previous  
 1178 result, we get  
 1179

$$1180 \det V_n = \det V_0 \cdot \prod_{t=1}^n \left( 1 + \|\phi_t\|_{V_{t-1}^{-1}}^2 \right) \quad (115)$$

1183 which implies  
 1184

$$1185 \sum_{t=1}^n \log \left( 1 + \|\phi_t\|_{V_{t-1}^{-1}}^2 \right) = \log \left( \frac{\det V_n}{\det V_0} \right) \quad (116)$$

1187 which proves the claim.  $\square$

1188 A.6.3 UCB OPTIMISATION  
1189

1190 We here state the derivation of a well-known property of LinUCB for completeness.

1191 **Remark 1.** *We want to show that*

1192 
$$\arg \max_{z \in \mathcal{C}_t} \max_{w \in \mathcal{C}_t} w^\top \psi(s_t, z) = \arg \max_{z \in \mathcal{C}_t} \psi(s_t, z)^\top \hat{z}_{t-1} + \beta_t \|\psi(s_t, z)\|_{V_{t-1}^{-1}} \quad (117)$$

1194 *Focusing on the inner max term, we have*

1195 
$$\max_{w \in \mathcal{C}_t} w^\top \psi(s_t, z) = \max_{\|w - \hat{z}_t\|_{V_{t-1}} \leq \beta_t} w^\top \psi(s_t, z) \quad (118)$$

1196 
$$= \hat{z}_{t-1}^\top \psi(s_t, z) + \max_{\|u\|_{V_{t-1}} \leq \beta_t} u^\top \psi(s_t, z) \quad (119)$$

1197 
$$\leq \hat{z}_{t-1}^\top \psi(s_t, z) + \max_{\|u\|_{V_{t-1}} \leq \beta_t} \|u\|_{V_{t-1}} \|\psi(s_t, z)\|_{V_{t-1}^{-1}} \quad (120)$$

1198 
$$= \hat{z}_{t-1}^\top \psi(s_t, z) + \beta_t \|\psi(s_t, z)\|_{V_{t-1}^{-1}} \quad (121)$$

1199 where the equality is attained by

1200 
$$u^* = \beta_t \frac{V_{t-1}^{-1} \psi(s_t, z)}{\|\psi(s_t, z)\|_{V_{t-1}^{-1}}} \quad (122)$$

1208 B ADDITIONAL EXPERIMENTS  
12091210 B.1 HOW MUCH COMPUTE DOES OPTI-BFM NEED?  
12111212 To estimate the compute cost of OptTI-BFM, we measure the time for the action selection and update  
1213 steps on an Nvidia RTX 4090, skipping the first few calls to avoid measuring JIT compilation time.  
1214 Table 1 shows that OptTI-BFM and OptTI-BFM-TS are about 5x and 4x slower than running just the  
1215 policy (Oracle) respectively.1216 Table 1: Computational cost of OptTI-BFM compared to running just the policy on an Nvidia RTX  
1217 4090 GPU. OptTI-BFM is about 5x and OptTI-BFM-TS about 4x slower.  
1218

	Oracle ( $\pi_{z_r}$ )	OptTI-BFM	OptTI-BFM-TS
Time per Step	0.772 ms	3.567 ms	2.756 ms
Frequency	1386 Hz	280 Hz	363 Hz

1224 B.2 REQUESTING REWARD LABELS EXPLICITLY  
12251226 We have so far assumed that each environment interaction provides a reward label, and is thus  
1227 synonymous with the labeling cost. One could also consider a setting where the agent can decide  
1228 whether it wants to observe the reward label in each transition. A very simple approach in this setting  
1229 is to threshold the D-gap (Kiefer & Wolfowitz, 1960; Lattimore & Szepesvári, 2020), which describes  
1230 how much the confidence ellipsoid shrinks for a new  $\phi_t$ .

1231 
$$\Delta(\phi_t) = \log \det(V_{t-1} + \phi_t \phi_t^\top) - \log \det V_{t-1} = \log \left(1 + \|\phi_t\|_{V_{t-1}^{-1}}^2\right) \quad (123)$$

1232 This approach can be directly applied to OptTI-BFM by introducing a hyper-parameter  $\kappa$  and only  
1233 requesting  $r_t$  if  $\Delta(\phi_t) \geq \kappa$ . We evaluate labeling efficiency for this variant: for each labeling budget  
1234  $n$  (# Samples), we measure average rewards at the time step  $t$  in which the budget is exhausted,  
1235 i.e.  $\frac{1}{t} \sum_{i=1}^t r_t$ . This is presented in Fig. 7 (see also Fig. 19 and Fig. 20 for per-task results). We  
1236 observe that  $\kappa$  trades-off environment interaction and labeling cost: for higher  $\kappa$ , good performance  
1237 is achieved with fewer labels, but more episodes are required to request the same number of labels.  
1238 Given a specific cost for labels, a slightly more involved thresholding scheme (Tucker et al., 2023)  
1239 could potentially be used to decide the threshold apriori. Surprisingly, this variant of OptTI-BFM  
1240 can maintain the performance of the main algorithm, especially in easier tasks, while reducing the  
1241 amount of labeled data by one order of magnitude.

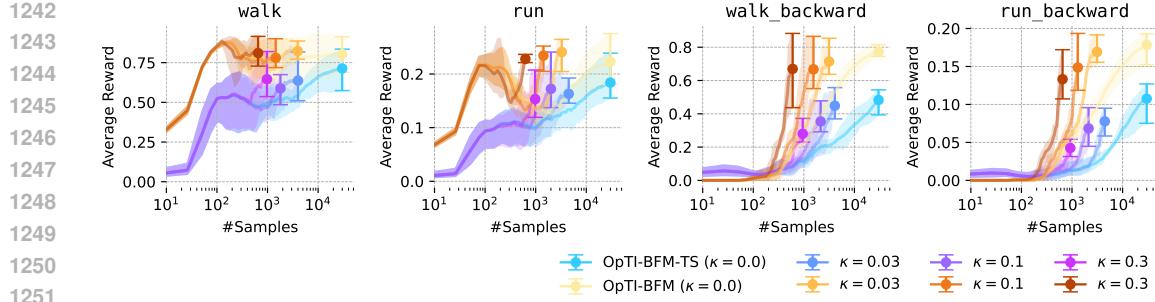


Figure 7: Average reward in Cheetah at different numbers of requested reward labels (# Samples) of our main methods for different information thresholds  $\kappa$ . We stop interaction after 30k environment steps.  $\kappa$  trades-off interaction cost with labeling cost. More than one order of magnitude less reward labels can still result in the same performance in easier tasks! We report per method results of all environments in Fig. 20 and Fig. 19

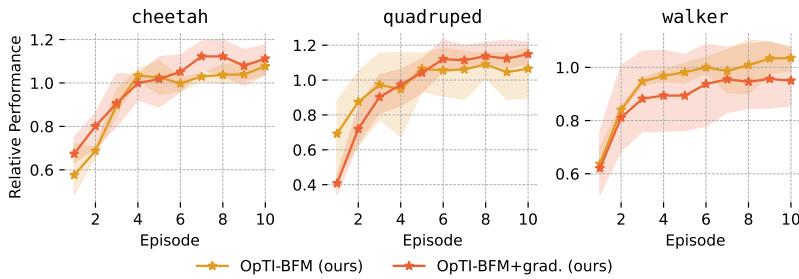


Figure 8: Relative performance after different # of environment interactions of OpTI-BFM and OpTI-BFM +grad; the latter is a variant that optimizes the USB objective through 8 gradient steps. Performances are similar; we report per-task results in Fig. 18 in Appendix C.

### B.3 FIRST-ORDER OPTIMIZATION OF THE UCB OBJECTIVE

This section introduces a variant of OpTI-BFM which uses gradient ascent to optimize the UCB objective:

$$\arg \max_{z \in \mathcal{Z}} \psi(s_t, z)^\top \hat{z}_{t-1} + \beta_t \|\psi(s_t, z)\|_{V_{t-1}^{-1}}, \quad (124)$$

as opposed to the random shooting approach we describe in Section 4. We validate this variant empirically on walker tasks, relying on the Adam (Kingma & Ba, 2017) optimizer with a learning rate of 0.1 to perform 2, 4, 8, or 16 gradient steps in each step, and maintaining the optimizer state throughout. At each step, we utilize  $\hat{z}_{t-1}$  to warm-start the optimization procedure. We report overall return curves in Fig. 8, and an ablation of compute costs in Fig. 9. We observe that, given a sufficient number of gradient steps, this variant approaches the main instantiation of our method in performance. However, compute requirements for gradient ascent scale linearly with the number of gradient steps, and surpass those of random shooting. Considering that state embeddings are, for most BFM variants, relatively low-dimensional (e.g.  $d = 50$ ), we conclude that random shooting approach should be preferred, as it is also less prone to fall into local optima. For very high-dimensional task spaces, first-order optimization remains however a promising option.

### B.4 OPTI-BFM FOR LOCO-NAVIGATION

To evaluate effectiveness beyond the locomotion experiments presented in the main text, we evaluate OpTI-BFM in antmaze-medium-navigate-v0, from the OGBench (Park et al., 2024a) benchmark. By default, this environment defines an indicator function reward that is 1 if the piloted quadruped reaches a certain goal position, at which point the episode terminates immediately. To make this environment consistent with our theory, we change the task to “holding” the goal position: the episode does not terminate until 1000 steps are accumulated. Furthermore, we slightly increase the radius of the sparse reward to cover the maze cell containing the goal, as seen in Fig. 10b in green, as reaching the goal exactly is otherwise challenging during online learning. As is common in this

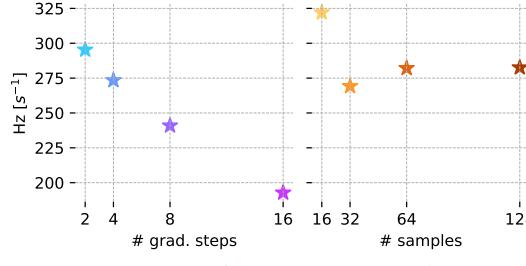
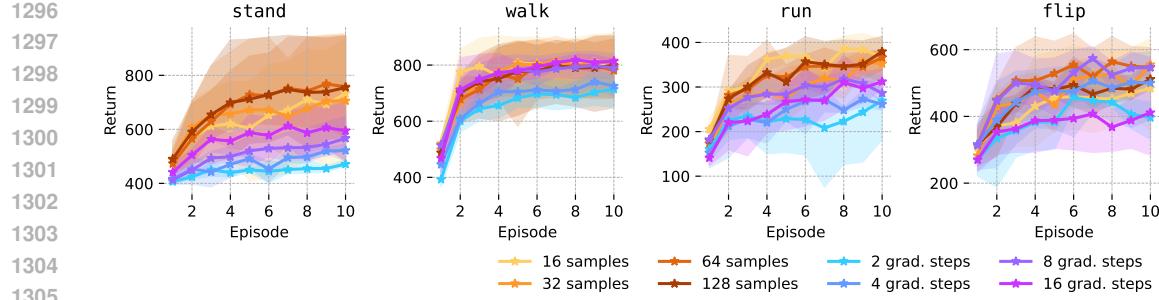


Figure 9: Performance and inference speed of OpTI-BFM, and a gradient-based variant (OpTI-BFM +grad). The performance gap is moderate, but the computational cost of the gradient-based variant is generally higher.

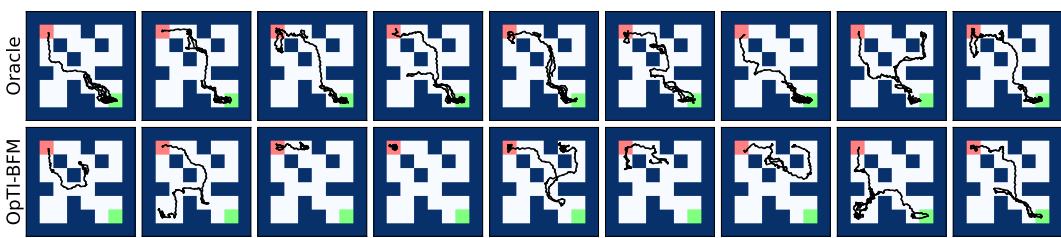
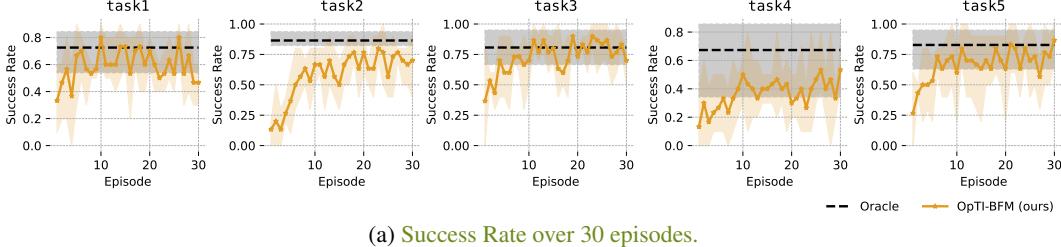


Figure 10: OpTI-BFM can infer a goal-reaching task in the OGBench (Park et al., 2024a) locomotion environment antmaze-medium-navigate-v0.

benchmark (Park et al., 2024a), we introduce a behavior cloning coefficient of 0.01 to the actor loss in pre-training, as the pre-training data contains near-expert navigation behavior as opposed to the high-coverage exploration data in ExORL. Results shown in Fig. 10 are consistent with our main evaluation: OpTI-BFM can reach near-Oracle performance within a handful of episodes.

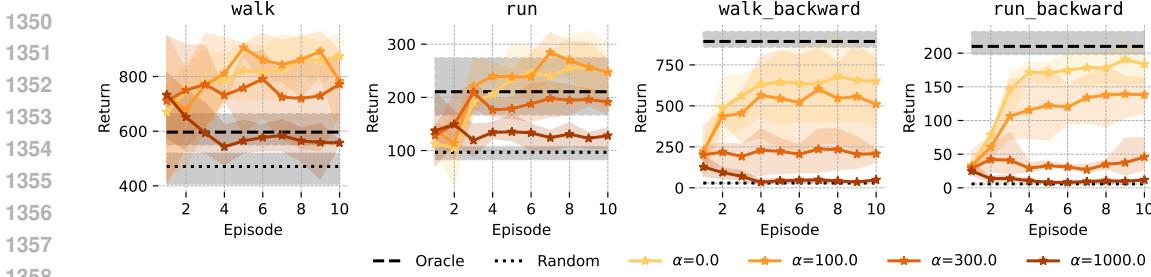


Figure 11: Performance of OpTI-BFM over 10 episodes in the Cheetah environment with a mismatch between  $\psi$  and  $\phi$  (see Eq. (125)) of magnitude  $\alpha$ . Performance deteriorates as  $\alpha$  increases.

### B.5 OPTI-BFM WITH INACCURATE SUCCESSOR FEATURES

To investigate how robust OpTI-BFM is to violations of assumption A1, we evaluate its performance when introducing a systematic mismatch between the discounted sum of  $\phi$  and  $\psi$ . Concretely, we let OpTI-BFM use

$$\psi'(s_t, z) = \psi(s_t, z) + \alpha \cdot \text{MLP}(z; \theta) \quad (125)$$

where MLP is a two layer network with ReLU activation and L2-normalized outputs, and  $\theta$  are randomly sampled weights, which we resample at the start of task inference. This additional MLP introduces a systematic,  $z$ -dependent bias, whose magnitude may be controlled through the hyperparameter  $\alpha$ . To avoid confounding factors as much as possible, we adopt an optimization procedure that samples  $z$  from the whole  $\mathcal{Z}$  to optimize the UCB objective, increasing the number of samples to  $n = 512$ . As expected, Fig. 11 shows that performance deteriorates as we increase  $\alpha$ ; very large values of  $\alpha$  ( $\approx 1000$ ) are necessary to render OpTI-BFM completely uninformative, and approach performance of the random baseline. As the norm of  $\psi$  is generally bounded by  $\frac{1}{1-\gamma} \sqrt{d} \approx 350$ , these constitute significant perturbations.

### B.6 OPTI-BFM WITH NON-LINEAR REWARDS

This section investigates how a violation of assumption A2, i.e. linearity of rewards in the features  $\phi$ , impacts performance of OpTI-BFM. In order to isolate this effect as much as possible from other factors (e.g., how hard the task specified by a reward function is), we consider a family of reward functions with increasingly larger orthogonal components to the feature space. Starting from an existing reward function  $r(s)$ , we project it to the feature space, obtaining  $z_r$ , and then extract its orthogonal component  $e(s) = r(s) - \phi(s)^\top z_r$  by training a small network. We can then control linearity of rewards and investigate performance under the reward function

$$r_\alpha(s) = z_r^\top \phi(s) + \alpha \cdot e(s). \quad (126)$$

Note that  $r_0(s) = z_r^\top \phi(s)$  which has 0 projection error, and  $r_1(s)$  recovers  $r(s)$ . While this technique can approximately disentangle reward components, for  $\alpha > 1$  the reward function might increase in scale: episodic performance might thus actually grow with projection error. Furthermore, the function approximation of  $e$  might be inaccurate, meaning that  $\alpha$  does not allow directly tuning the projection error. We can nonetheless compare the return of OpTI-BFM (upon convergence, i.e. at its 10th episode) with that of the Oracle, over a range of mean absolute projection errors, as Fig. 12. While the results are increasingly noisy, as the error increases far beyond ranges encountered for existing rewards, OpTI-BFM seems to suffer slightly more in specific tasks—generally achieving worse performance compared to the Oracle for similarly misaligned reward functions. This may be explained by the fact that OpTI-BFM relies on linearity for both data collection and task inference, while the Oracle relies on independently collected data.

### B.7 CAN OPTI-BFM LEARN FROM NOISY REWARDS?

In this section we evaluate the performance of OpTI-BFM under noisy reward feedback. To this end, we add zero-mean Gaussian noise to the observed rewards with different standard deviations  $\sigma$ . As predicted by Eq. (66), we see in Fig. 13 that convergence is slower for higher standard deviations. Adjusting the hyper-parameter  $\beta = 10$  for  $\sigma = 10$  did improve performance slightly, but with a noise level that is 10x higher than the reward of the environment  $r_t \in [-1, 1]$  convergence remains slow.

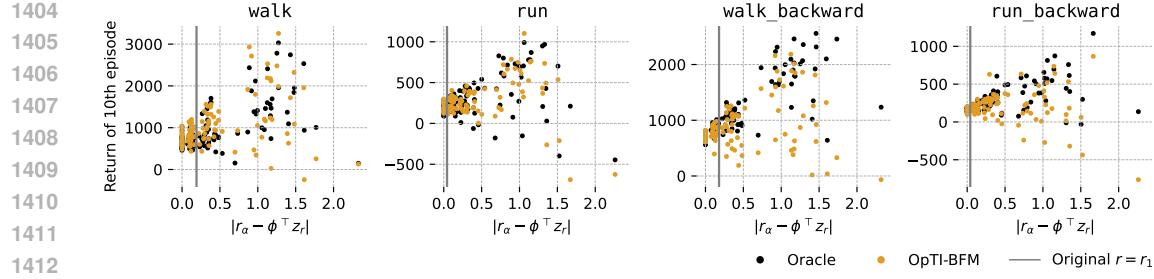


Figure 12: Return of the 10th episode of Oracle and OpTI-BFM in the Cheetah environment when artificially increasing or decreasing the projection error of the reward function onto  $\phi$  through a learned network.

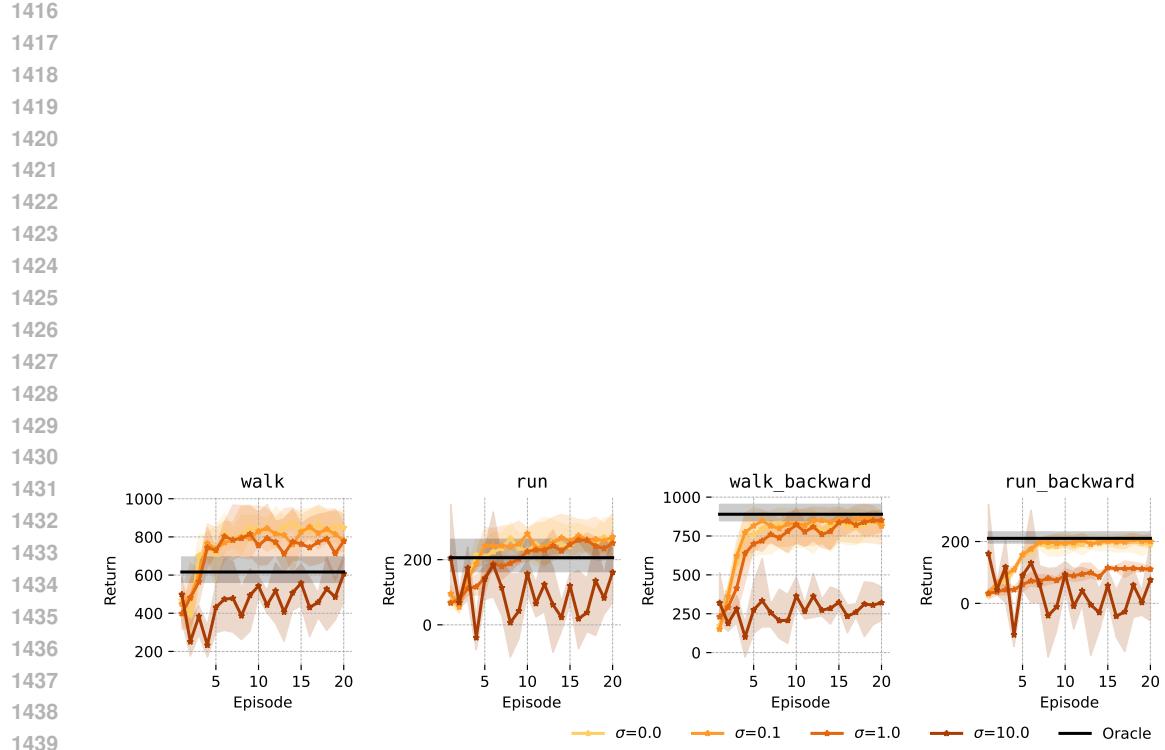


Figure 13: Performance of OpTI-BFM over 10 episodes in the Cheetah environment when adding Gaussian noise with different standard deviations to the rewards it observes. Convergence slows down as the noise is increased.

## C FULL EXPERIMENTAL RESULTS

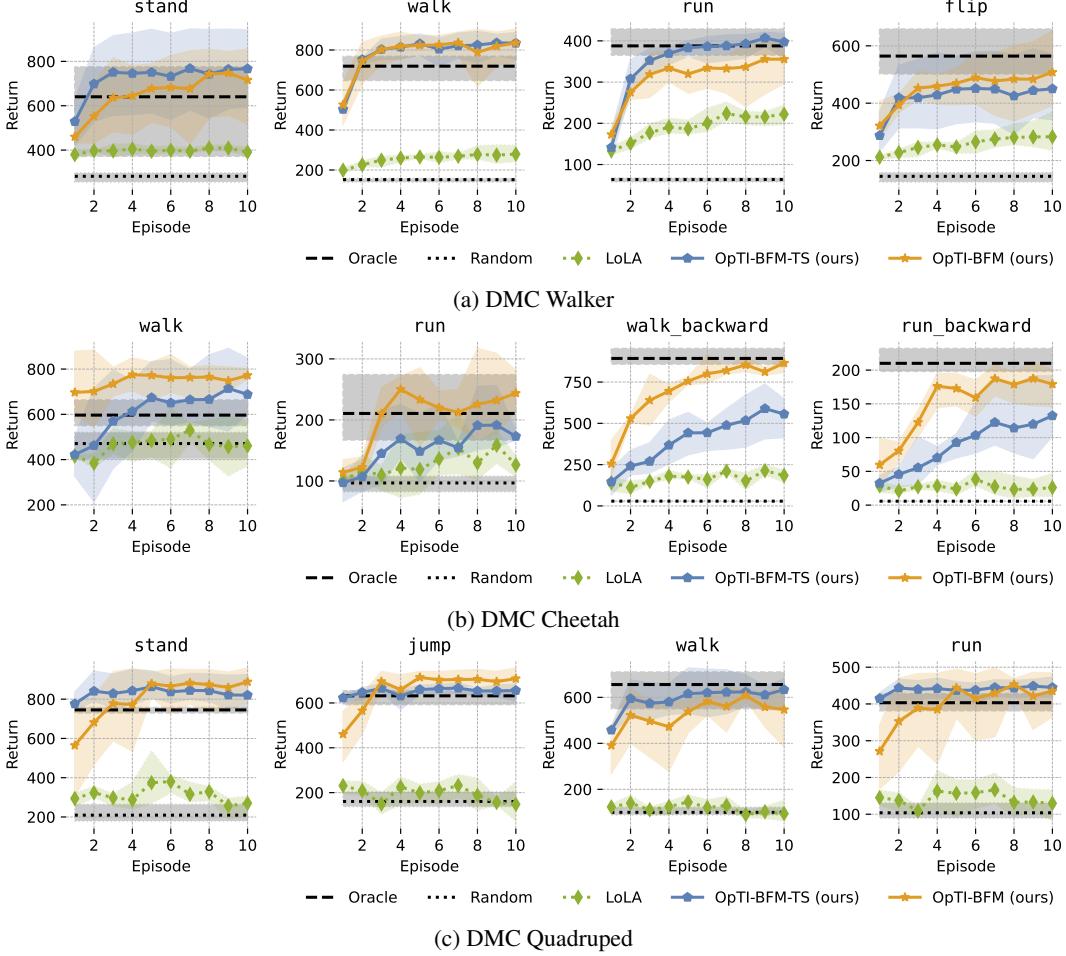


Figure 14: Episode Return over 10 episodes (10k steps) of interaction in DMC.

## D TRAINING PROTOCOL

## D.1 FORWARD BACKWARD FRAMEWORK

We use the forward backward (FB) framework (Touati & Ollivier, 2021) as the base BFM through our experiments. To fully explain FB, we quickly introduce the successor measure (Dayan, 1993; Blier et al., 2021)  $M^\pi$ ; it is defined as

$$M^\pi(s_0, a_0, X) = \sum_{t \geq 0} \gamma^t \Pr[s_t \in X | s_0, a_0, \pi]. \quad (127)$$

and can be thought of as the discounted state occupancy of a policy  $\pi$  when starting in  $s_0, a_0$ . FB then learns the low-rank decomposition of the successor measure density w.r.t. the empirical dataset measure  $\mathcal{D}_{\text{train}}(ds)$ :

$$M^{\pi_z}(s, a, ds^+) = F(s, a, z)^\top B(s^+) \mathcal{D}_{\text{train}}(ds^+). \quad (128)$$

where the policy should satisfy

$$\pi_z(a|s) > 0 \implies a \in \arg \max_a F(s, a, z)^\top z. \quad (129)$$

The successor measure and successor features are closely related, since we have that

$$\psi^\pi(s, a) = \int M^\pi(s, a, ds^+) \phi(s^+). \quad (130)$$

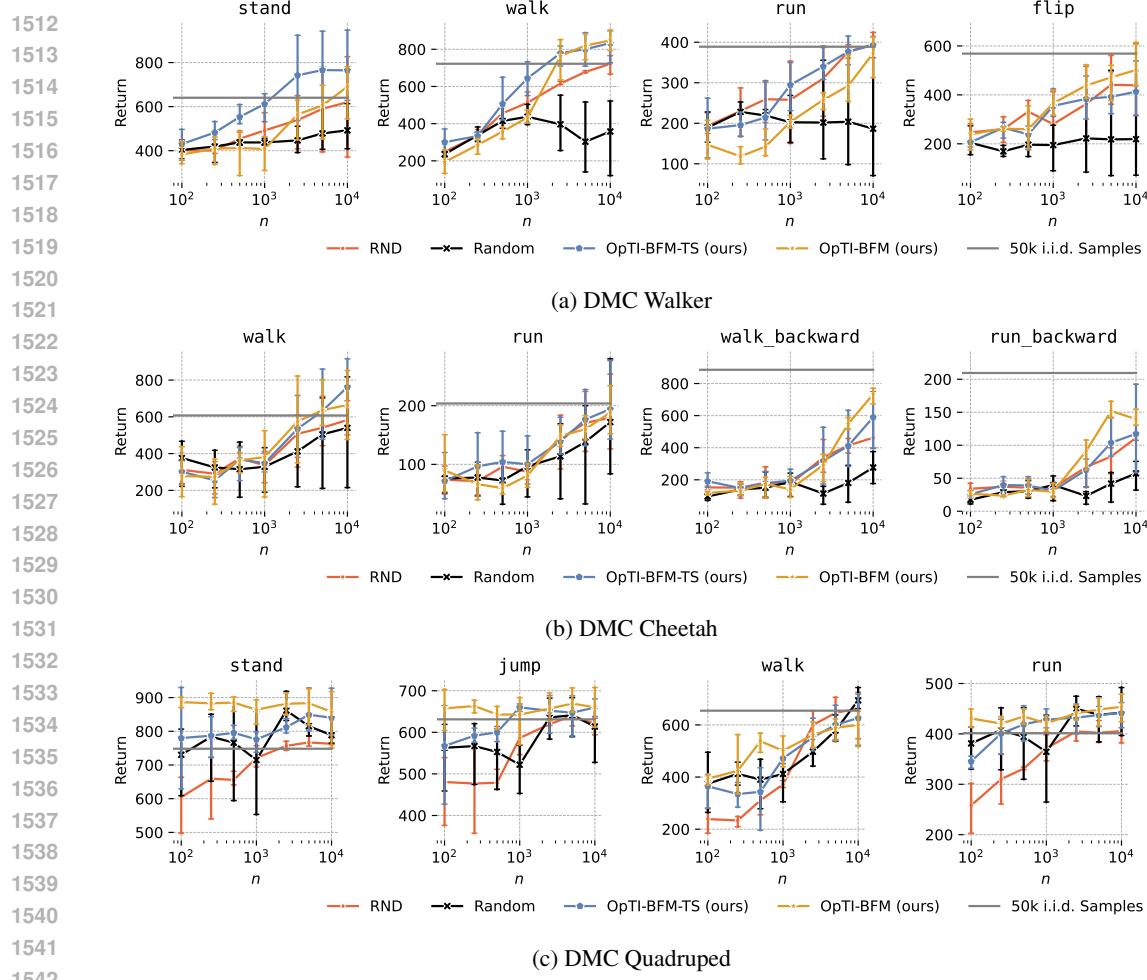


Figure 15: Per task results of Fig. 3. We show absolute performance here and include the Oracle performance (gray line).

Following Touati & Ollivier (2021) (Theorem 13), we thus have for FB

$$\int B(s^+) M_z^\pi(s, a, ds^+) = \int B(s^+) B(s^+)^\top \mathcal{D}_{\text{train}}(ds^+) F(s, a, z) \quad (131)$$

$$= \mathbb{E}_{s \sim \mathcal{D}_{\text{train}}} [B(s^+) B(s^+)^\top] F(s, a, z) \quad (132)$$

$$= (\text{Cov}_{\mathcal{D}_{\text{train}}} B) F(s, a, z) \quad (133)$$

so  $F(s, a, z)$  are SFs of features  $\phi(s) = (\text{Cov}_{\mathcal{D}_{\text{train}}} B)^{-1} B(s)$ . Task inference for FB with labeled dataset  $\mathcal{D}$  then effectively becomes

$$z_r = (\text{Cov}_{\mathcal{D}} \phi)^{-1} \mathbb{E}[\phi(s) r(s)] \quad (134)$$

$$= (\text{Cov}_{\mathcal{D}} ((\text{Cov}_{\mathcal{D}_{\text{train}}} B)^{-1} B))^{-1} (\text{Cov}_{\mathcal{D}_{\text{train}}} B)^{-1} \mathbb{E}[B(s) r(s)] \quad (135)$$

$$= (\text{Cov}_{\mathcal{D}_{\text{train}}} B) (\text{Cov}_{\mathcal{D}} B)^{-1} (\text{Cov}_{\mathcal{D}_{\text{train}}} B) (\text{Cov}_{\mathcal{D}_{\text{train}}} B)^{-1} \mathbb{E}[B(s) r(s)] \quad (136)$$

$$= (\text{Cov}_{\mathcal{D}_{\text{train}}} B) (\text{Cov}_{\mathcal{D}} B)^{-1} \mathbb{E}[B(s) r(s)] \quad (137)$$

which is consistent with Touati & Ollivier (2021) (Proposition 15). In practice we pre-compute  $(\text{Cov}_{\mathcal{D}_{\text{train}}} B)$  with 50k samples from  $\mathcal{D}_{\text{train}}$  after training.

## D.2 PRE-TRAINING

As is the standard in zero-shot RL benchmarks (Touati et al., 2023; Agarwal et al., 2025b), we train our FB model using an offline dataset collected with RND (Burda et al., 2018; Yarats et al., 2022) consisting of 10M transitions.

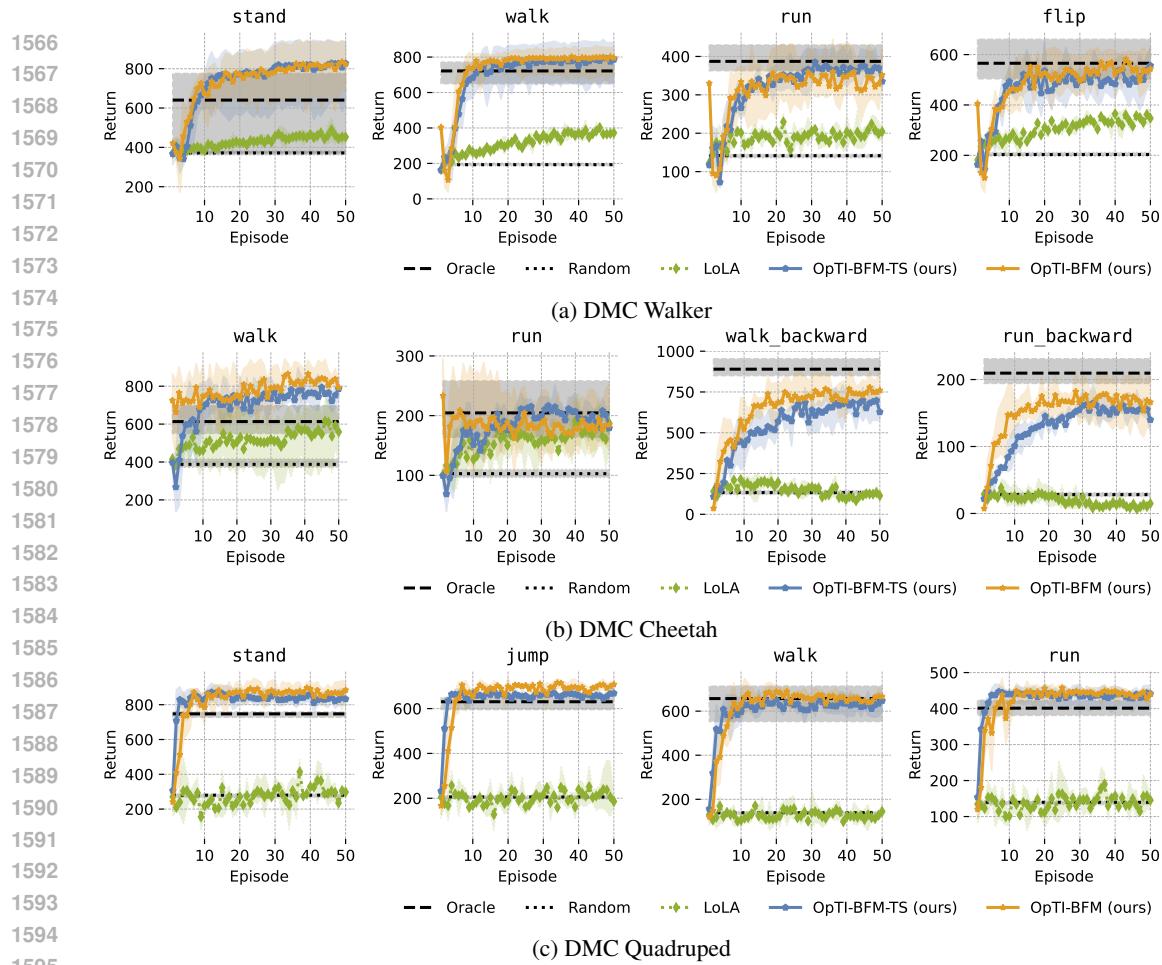


Figure 16: Episode Return over 50 episodes (50k steps) of interaction in DMC when only allowing switching of the task embedding at the start of episodes. Note that LoLA resamples its task embedding every 50-250 steps (hyper-parameter).

FB is trained using 3 losses. Firstly, the successor measure loss (Blier et al., 2021)

$$\mathcal{L}_{FB}(z) = \mathbb{E}_{\substack{s, a, s' \sim \mathcal{D}_{\text{train}} \\ a' \sim \pi_z(\cdot | s')}} \mathbb{E}_{s^+ \sim \mathcal{D}_{\text{train}}} \left[ -2F(s, a, z)^\top B(s') \right] \quad (138)$$

$$+ (F(s, a, z)^\top B(s^+) - \gamma \bar{F}(s', a', z)^\top \bar{B}(s^+))^2 \right], \quad (139)$$

where  $\bar{F}$ ,  $\bar{B}$  are target networks, secondly, an orthogonality regularizing loss on  $B$

$$\mathcal{L}_{\text{ortho}} = \mathbb{E}_{\substack{s \sim \mathcal{D}_{\text{train}} \\ s' \sim \mathcal{D}_{\text{train}}}} \left[ -2 \|B(s)\|_2^2 + B(s)^\top B(s') \right], \quad (140)$$

and thirdly a DDPG-style loss for  $\pi$ :

$$\mathcal{L}_\pi(z) = \mathbb{E}_{s \sim \mathcal{D}_{\text{train}}} \mathbb{E}_{a \sim \pi_z(\cdot | s)} [-F(s, a, z)^\top z], \quad (141)$$

using the reparameterization trick for Gaussian policies.

### D.3 ARCHITECTURE AND HYPER-PARAMETERS

We choose to follow the implementation details of the most recent work on FB (Tirinzoni et al., 2025), specifically their implementation for DMC in the released code-base. The  $B$ -network is a 3-layer MLP. The  $F$  is an ensemble of size 2 with each two 2-layer MLPs to encode the arguments  $(s, a)$  and  $(s, z)$  that are then concatenated and fed into another 2-layer MLP.  $\pi$  uses two 2-layer MLPs to

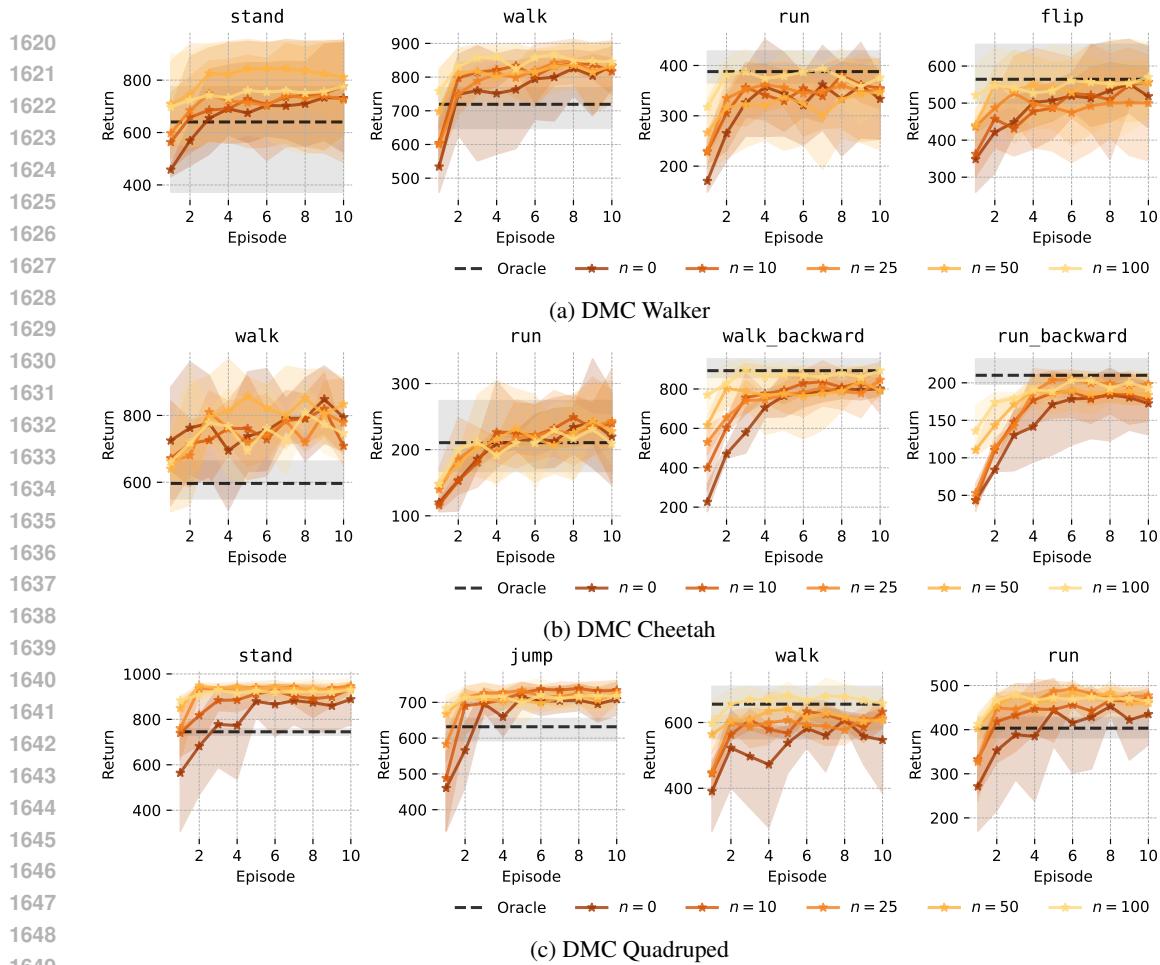


Figure 17: Return of OpTI-BFM when warm-starting with  $n$  i.i.d. labeled states from the pre-training dataset.

encode the arguments  $s$  and  $(s, z)$  and then also another 2-layer MLP to map from the concatenated layers to a mean  $\mu$ . The policy is then a truncated Gaussian with fixed standard deviation of  $\sigma = 0.2$  that truncates at  $1.5\sigma$ . The rest of the pre-training hyper-parameters are listed in Table 2 below. The task embedding space of FB is the  $d$ -dimensional hyper-sphere since the optimal policy  $\pi_z$  is invariant to the scale of the reward function. Note that OpTI-BFM cannot make this simplification because it tries to estimate the hidden parameter of the reward function and not the latent that is plugged into the policy. We can recover the latter easily with an L2 normalization.

## E EXPERIMENT PROTOCOL

We evaluate our methods on a common zero-shot RL benchmark (Touati et al., 2023; Agarwal et al., 2025b). The benchmark consists of the Walker, Cheetah, and Quadruped environments from DMC (Tassa et al., 2018) with four tasks (reward functions) each. Note that each task has randomized initial states, so when evaluating single episode performance, e.g. Fig. 15, we report the mean over 20 episodes. And when evaluating task inference performance over multiple episode, e.g., Figs. 2 and 16, we report the mean over 10 trials. All error-bars and shaded regions denote min-max-intervals over three training seeds around mean performance.

### E.1 CUSTOM WALKER VELOCITY TASKS

The custom tasks we implement for DMC Walker Fig. 5 consist of only the velocity tracking components of the `stand`, `walk`, and `run` tasks. Further note that, by default, higher velocity is always allowed: a running policy will also perform rather well in standing. For this reason, we modify

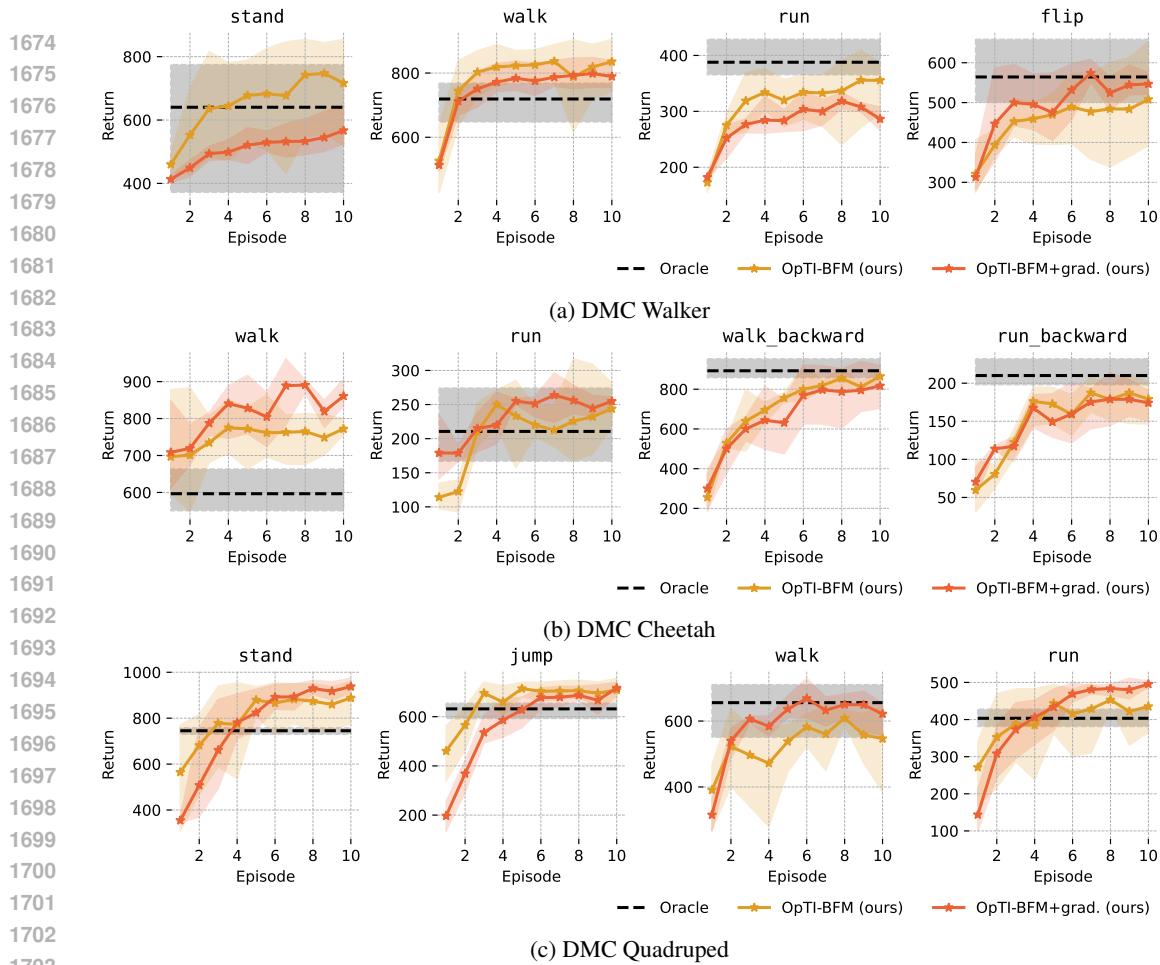


Figure 18: Return of OpTI-BFM and OpTI-BFM +grad. that use random-shooting and gradient ascent respectively to optimize the UCB objective. The two variants perform very similarly.

the velocity reward bonus so that it is 1 if the velocity matches the target exactly, and then linearly tapers to 0.5 when off by 2. From there, the reward then directly drops to 0.

## F INFERENCE HYPER-PARAMETERS

For each experiment and method we perform a grid-search over hyper-parameters. We then choose the hyper-parameters with highest overall cumulative return *per environment* for each method to report performance.

### F.1 OPTI-BFM

We found both OpTI-BFM and OpTI-BFM-TS to be very robust to the hyper-parameters we tested. For each method we consider a single hyper-parameter with three values each: For UCB, we test a fixed  $\beta_t = \beta \in \{1.0, 0.1, 0.001\}$ . For TS, we test  $\sigma \in \{0.1, 0.001, 0.0001\}$ . This means all other parameters were held constant. Specifically,  $\rho = 1$  and  $\lambda = 1$  if not reported otherwise (Fig. 5). The number of samples for the UCB optimization was  $n = 128$  throughout if not specified otherwise.

### F.2 LOLA

For LoLA we consider a range of hyper-parameters to trade-off exploration, update frequency, update step-size, and variance in the gradient. We search all combinations of:

- horizon length and task embedding update rate {50, 100, 250} as in (Sikchi et al., 2025);

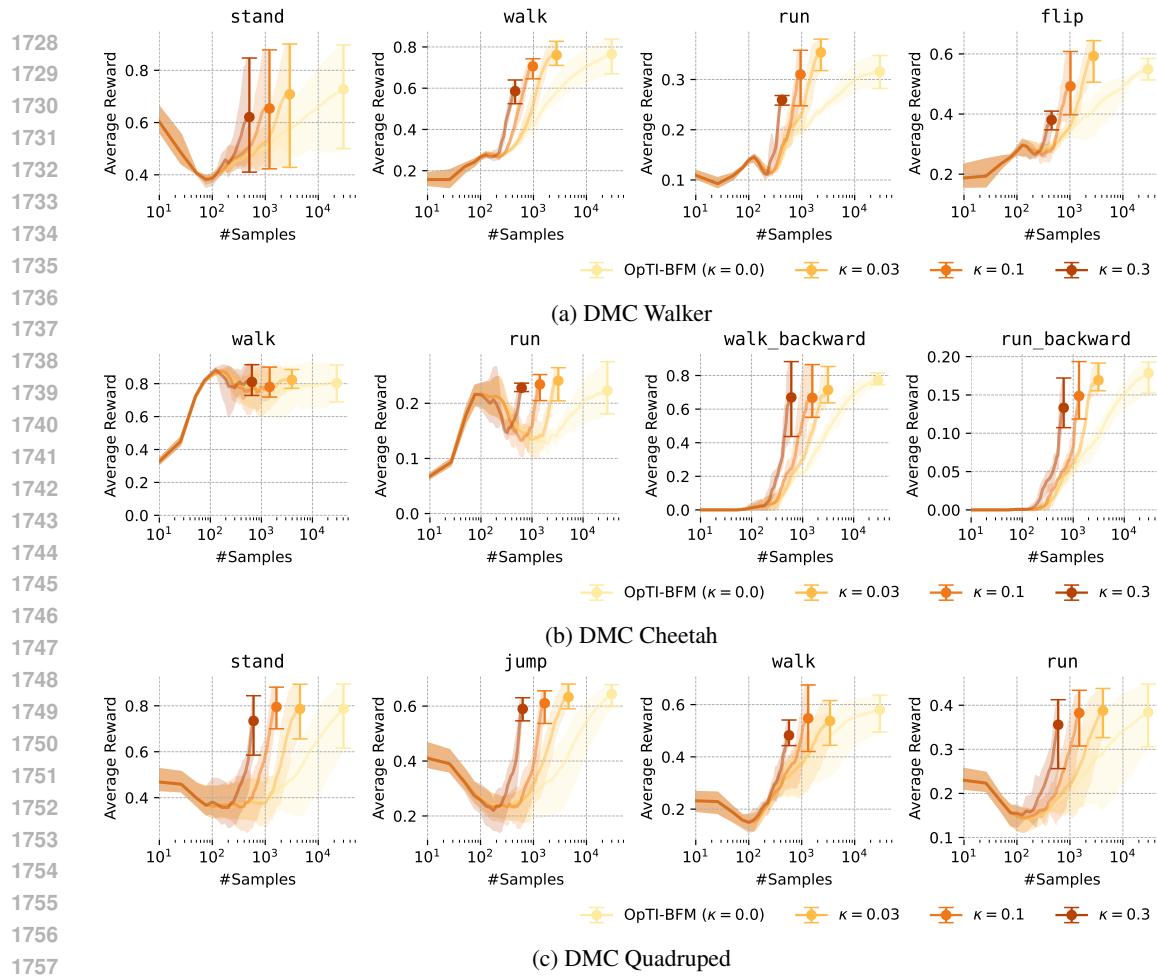


Figure 19: Average reward of OpTI-BFM at different numbers of requested reward labels (# Samples) for different information thresholds  $\kappa$ . We stop interaction after 30k environment steps.  $\kappa$  trades-off interaction cost with labeling cost. More than one order of magnitude less reward labels can still result in the same performance in easier tasks.

- learning rate  $\{0.1, 0.05\}$  as in (Sikchi et al., 2025);
- standard deviation of the Gaussian on the task embedding  $\{0.05, 0.1, 0.2\}$ .
- The batch size reported in Sikchi et al. (2025) is 5-10: we thus consider a replay buffer of 1000, which results in an effective batch size of 20, 10, and 4 for the respective update rates.

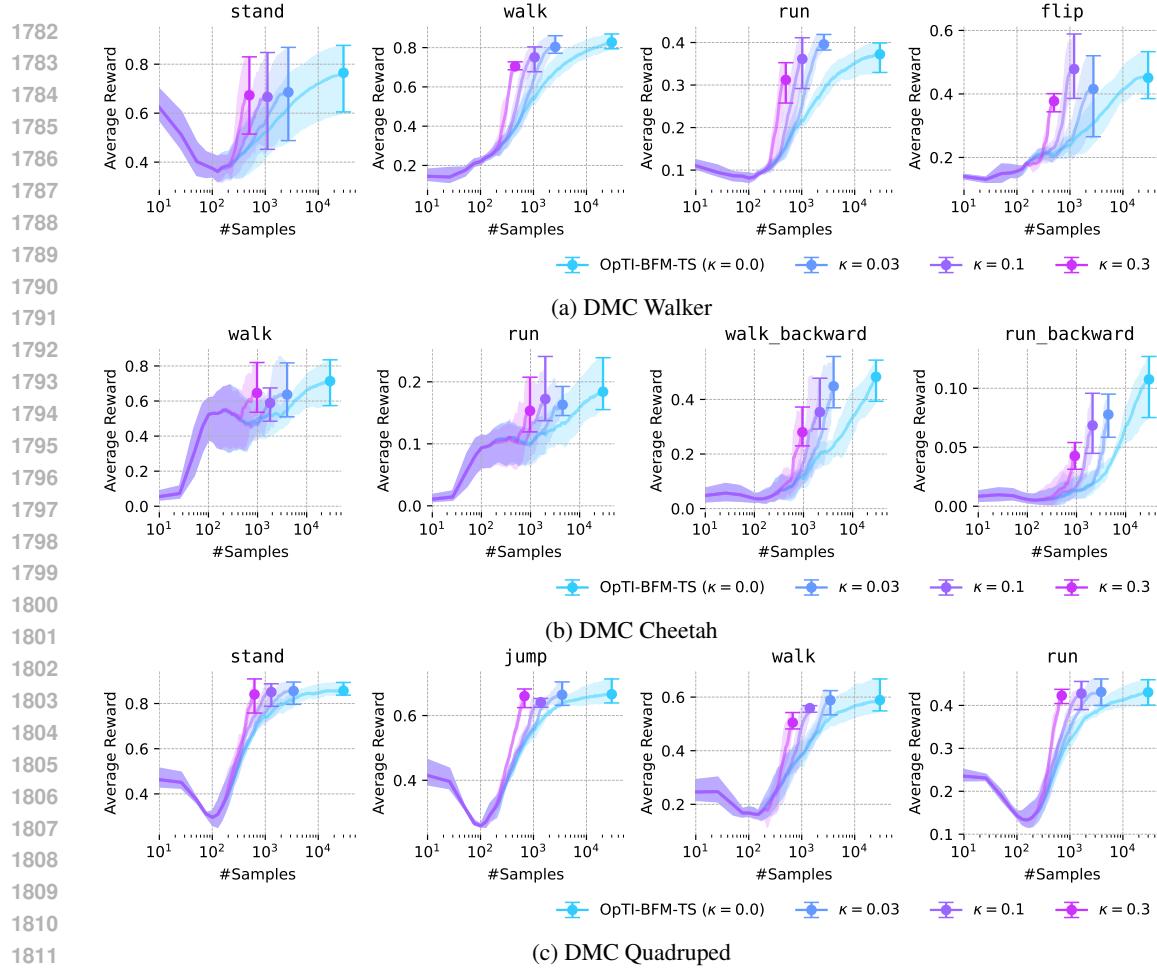


Figure 20: Average reward of OpTI-BFM-TS at different numbers of requested reward labels (# Samples) for different information thresholds  $\kappa$ .

Table 2: FB Hyper-Parameters.

Name	Value
discount $\gamma$	0.98
batch size	1024
# training steps	2M
optimizer	adam
learning rate	1e-4
target network update factor	0.01
weight of $\mathcal{L}_{\text{ortho}}$	1.0
$Q$ -value penalty	0.5
fixed actor standard deviation	0.2
actor sample noise clipping	0.3
$z$ sampling	50% $\mathcal{D}_{\text{train}}$ and 50% Random
dimension $d$	50
$B$ network final activation	L2 normalization
$B$ network hidden dimension	256
$F$ network hidden dimension	1024
$\pi$ network hidden dimension	1024
all networks first activation	Layernorm + Tanh
all other activations	relu