ENSEMBLE SAMPLING WITH OPTIMISTIC REFRESHES FOR BEHAVIOR FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Online task inference with behavior foundation models (BFMs) aims to retrieve a high-performing policy for a target reward function through interaction with the environment, without updating the pretrained model. However, collecting reward-labeled data while achieving high returns at low computational cost remains challenging. In this paper, we propose Ensemble Sampling with Optimistic Refreshes for BFMs (ESOR-BFM), an online task inference method that combines ensemble sampling with optimistic recentering. When the regularized Gram matrix built from state features changes substantially, ESOR-BFM refreshes the ensemble perturbations and recenters the ensemble around an optimistic task embedding estimate. ESOR-BFM samples ensemble members for policy selection between refreshes and performs optimistic search only at refresh times, amortizing the search cost over multiple interactions. We theoretically establish a sublinear regret guarantee for an episodic variant of ESOR-BFM with well-trained BFMs, matching the state-of-the-art regret rate for ensemble sampling in stochastic linear contextual bandits. Extensive experiments on ExORL show that ESOR-BFM consistently outperforms existing online task inference methods and achieves the highest mean returns across settings with stationary rewards, selective reward labeling, and inaccurate successor feature estimates.

1 INTRODUCTION

Zero-shot reinforcement learning (RL) (Touati et al., 2023) aims to learn a family of policies from reward-free data, enabling agents to solve downstream tasks specified by reward functions without test-time planning or policy optimization. Behavior foundation models (BFMs) (Touati & Ollivier, 2021; Park et al., 2024; Agarwal et al., 2025; Tirinzoni et al., 2025; Jajoo et al., 2026; Bagatella et al., 2026) have emerged as a prominent framework for zero-shot RL by pretraining diverse task-conditioned policies. Many BFM methods use Universal Successor Features (USFs; Ma et al. 2020), which predict the expected discounted sum of state features under each policy by conditioning on its task embedding. When the reward is linear in these features, policy evaluation reduces to an inner product between the policy's successor features (SFs) and the reward weight (Barreto et al., 2017).

In USF-based BFMs, each policy is learned to maximize the expected return under the reward function specified by its task embedding. After pretraining, the estimated reward weight is used as a task embedding to select the corresponding policy. Offline task inference typically obtains this estimate by regressing rewards on the learned state features using a fixed dataset of state-reward pairs (Touati et al., 2023; Park et al., 2024). However, a dataset labeled with rewards may be unavailable or costly to obtain in practice (Rupf et al., 2026). As an alternative, Rupf et al. (2026) study *online task inference*, in which an agent adaptively collects state-reward pairs during deployment without requiring access to the pretraining dataset or updates to the pretrained model parameters. Under the linear reward assumption, each policy's expected return is linear in the unknown reward weight, with its SFs playing a role analogous to arm features in linear contextual bandits (Rupf et al., 2026). Exploiting this connection, they propose OpTI-BFM and OpTI-BFM-TS, which use upper confidence bounds (UCB) (Abbasi-Yadkori et al., 2011) and Thompson sampling (TS) (Agrawal & Goyal, 2013), respectively, to balance exploration and exploitation when selecting task embeddings.

OpTI-BFM explicitly maximizes a UCB objective at each interaction step, requiring SF evaluations for multiple candidate task embeddings. OpTI-BFM-TS avoids this optimization through posterior

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

sampling, but can yield lower returns than OpTI-BFM under the same interaction budget (Rupf et al., 2026). These approaches motivate the following question: *Can we improve online task inference at a lower computational cost than OpTI-BFM by incorporating the results of occasional optimistic search into randomized task embedding selection over multiple interactions?*

In this work, we propose Ensemble Sampling with Optimistic Refreshes for Behavior Foundation Models (ESOR-BFM), an online task inference method that combines ensemble sampling (ES) with optimistic recentering. ES (Lu & Van Roy, 2017) is a randomized exploration method that maintains a finite ensemble of perturbed models and uses a uniformly sampled member to select actions. Each member is updated incrementally using new observations and random noise sampled independently across members. In our setting, each member is represented by a task embedding estimate formed by adding a member-specific perturbation to a linear regression estimate shared by all ensemble members. Inspired by Hwang & Oh (2026), we refresh the ensemble by reinitializing each member's perturbation when the regularized Gram matrix of observed state features grows beyond a prescribed multiplicative threshold. At each refresh, ESOR-BFM uses SFs and the current confidence ellipsoid to construct an optimistic task embedding estimate and recenter the ensemble at this estimate rather than at the linear regression estimate. This recentering incorporates the optimistic estimate into the ensemble members, allowing it to influence subsequent policy selection. At the same time, member-specific perturbations enable exploration through diverse task embedding estimates initialized around the optimistic estimate. ESOR-BFM can maintain the effect of optimistic recentering across interactions while retaining ES-style randomized exploration. Performing optimistic search only at refresh times amortizes its computational cost over multiple interactions. Our main contributions are summarized as follows:

- We propose ESOR-BFM, an ES algorithm for online task inference in BFMs that incorporates optimistic recentering at refresh times. For an episodic variant under the stated assumptions, we prove an expected cumulative regret bound of $\widetilde{\mathcal{O}}(d^{3/2}\sqrt{n})$ over n episodes in feature dimension d (Section 4.2), matching the standard linear-ES dependence on d and n. To the best of our knowledge, this is the first theoretical and empirical study of ES with optimistic refresh for online task inference in BFMs.
- Extensive numerical experiments on the ExORL benchmark demonstrate that ESOR-BFM outperforms existing online task inference methods. We further demonstrate that using data collected by ESOR-BFM for downstream task inference enables the selection of policies with higher returns (Section 5.1).
- We show that ESOR-BFM achieves competitive task performance with fewer reward labels (Section 5.2) and is less sensitive than OpTI-BFM to SF estimation errors (Section 5.3). These results highlight the practicality of ESOR-BFM for online task inference with costly reward feedback and imperfect pretrained models.
- Our empirical analysis shows that ESOR-BFM achieves higher mean returns and lower per-step runtime than OpTI-BFM, while optimistic recentering introduces only modest overhead to the underlying ES procedure (Section 5.4).

2 Related Work

Task Inference for Behavior Foundation Models. Building on USFs (Borsa et al., 2019; Ma et al., 2020), SF-based BFMs learn parameterized policies with a variety of representation learning approaches, including modeling successor measures (Touati & Ollivier, 2021; Agarwal et al., 2025), using spectral state representations (Wu et al., 2019; Touati et al., 2023), learning Hilbert representations (Park et al., 2024), and predicting latent dynamics (Jajoo et al., 2026; Bagatella et al., 2026). Offline task inference typically estimates a task embedding from a fixed dataset labeled with the target reward (Touati et al., 2023; Park et al., 2024). Starting from embeddings obtained through offline task inference, Sikchi et al. (2025) improve policy performance by searching over task embeddings during online interaction. Our work on online task inference is most closely related to Rupf et al. (2026), which infer task embeddings from online reward observations without requiring an initial labeled dataset or updating the pretrained BFM. These methods apply established UCB and TS exploration principles to online task inference: OpTI-BFM selects task embeddings by maximizing a UCB objective, while OpTI-BFM-TS samples them from a Gaussian posterior over task embeddings. In contrast, ESOR-BFM combines ES with optimistic recentering at refresh time.

Ensemble Sampling. ES is a randomized exploration framework that maintains a finite collection of perturbed models and selects one member at random for decision making. Unlike methods such as TS that repeatedly draw fresh randomized estimates, ES maintains a finite ensemble that is incrementally updated and reused across interactions. This structure is compatible with practical ensemble implementations, including those used for exploration in deep RL and online recommendation systems (Osband et al., 2016; Lu et al., 2018; Osband et al., 2018; 2019; Zhu & Van Roy, 2023; Zhou et al., 2025). These works progressively strengthened the theoretical understanding of ES while reducing the number of ensemble members required for efficient exploration. Motivated by this practical appeal, a growing line of work has developed theoretical guarantees for ES in stochastic linear bandits, addressing the adaptive dependence induced by persistent perturbations (Qin et al., 2022; Janz et al., 2024; Lee & Oh, 2024; Janz et al., 2026). Most recently, Hwang & Oh (2026) introduced an epoch refresh mechanism that reinitializes ensemble perturbations when the regularized Gram matrix changes substantially. By restricting persistent perturbations to epochs with controlled growth of the regularized Gram matrix, this mechanism substantially reduces the sufficient ensemble size while retaining provable exploration guarantees. Our work builds on this refresh mechanism but uses it differently: rather than reconstructing the ensemble around the current regression estimate, ESOR-BFM recenters the refreshed ensemble around an optimistic reward estimate derived from the BFM's SFs and reward uncertainty. The resulting optimistic guidance is then reused through persistent ensemble updates between refreshes.

3 PRELIMINARIES

Notations. For a positive integer n, let $[n] := \{1, \dots, n\}$, and let $\mathrm{Unif}(\mathcal{X})$ denote the uniform distribution over a finite set $\mathcal{X}$. For $x \in \mathbb{R}^d$ and a positive semidefinite matrix A, define $\|x\|_A := \sqrt{x^\top A x}$. For symmetric matrices A and B, we write $A \preceq B$ if $B - A$ is positive semidefinite.

Markov decision process. We consider a reward-free Markov decision process (MDP) $\mathcal{M}$ defined by the tuple $(\mathcal{S}, \mathcal{A}, P, d_0, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the state transition probability kernel, $d_0 \in \Delta(\mathcal{S})$ is the initial state distribution, and $\gamma \in [0, 1)$ is the discount factor. We denote by $\Delta(\mathcal{X})$ the set of probability distributions over $\mathcal{X}$. A policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ maps each state to a probability distribution over actions. Under a policy π, the interaction begins with $s_0 \sim d_0$ and generates a state-action sequence $(s_t, a_t)_{t \geq 0}$, where $a_t \sim \pi(\cdot \mid s_t)$ and $s_{t+1} \sim P(\cdot \mid s_t, a_t)$. For a downstream reward function $r : \mathcal{S} \to \mathbb{R}$, we define the state-action value function (Q-function) under policy π as $Q_r^\pi(s, a) := \mathbb{E}_{\mathcal{M}}^\pi \left[\sum_{t=0}^\infty \gamma^t r(s_t) \mid s_0 = s,\ a_0 = a\right]$.

Successor Features. Let $\phi : \mathcal{S} \to \mathbb{R}^d$ be a feature map. For a policy π, successor features (SFs; Barreto et al., 2017) are defined as

$$\psi^\pi(s, a) := \mathbb{E}_{\mathcal{M}}^\pi \left[\sum_{t=0}^\infty \gamma^t \phi(s_t) \,\middle|\, s_0 = s,\ a_0 = a \right].$$

Consider the class of linear reward functions $r_z(s) := \phi(s)^\top z$, parameterized by $z \in \mathbb{R}^d$. For any linear reward in this class, the Q-function under policy π factorizes as

$$Q_{r_z}^\pi(s, a) = \mathbb{E}_{\mathcal{M}}^\pi \left[\sum_{t=0}^\infty \gamma^t r_z(s_t) \,\middle|\, s_0 = s,\ a_0 = a \right] = \psi^\pi(s, a)^\top z.$$

Similarly, we define the state-level SFs as $\psi^\pi(s) := \mathbb{E}_{a \sim \pi(\cdot|s)}[\psi^\pi(s, a)]$, so that $V_{r_z}^\pi(s) = \psi^\pi(s)^\top z$.

Behavior Foundation Models. Building on the ability of SFs to evaluate policies across multiple downstream reward functions, a common class of Behavior Foundation Models (BFMs) learns a parameterized policy family $\{\pi_z\}_{z \in \mathcal{Z}}$ for rewards in the linear span of the feature map ϕ (Borsa et al., 2019; Touati & Ollivier, 2021; Park et al., 2024; Agarwal et al., 2025; Bagatella et al., 2026). Each $z \in \mathcal{Z} \subseteq \mathbb{R}^d$ serves as a task embedding that specifies a reward $r_z(s) = \phi(s)^\top z$ and indexes a corresponding policy π_z trained to maximize returns under r_z. In the idealized setting where π_z is optimal for r_z, it satisfies

$$\mathrm{supp}\left(\pi_z(\cdot \mid s)\right) \subseteq \arg\max_{a \in \mathcal{A}} \psi^{\pi_z}(s, a)^\top z \qquad \text{for all } s \in \mathcal{S}.$$

In practice, we use a USF approximator $\psi(s,a,z) \approx \psi^{\pi_z}(s,a)$ and its state-level counterpart $\psi(s,z) := \mathbb{E}_{a\sim\pi_z(\cdot|s)}[\psi(s,a,z)] \approx \psi^{\pi_z}(s)$ (Ma et al., 2020).

Task Inference. Given a pre-trained BFM and a target reward function r, *offline task inference* estimates a task embedding $\hat{z}_r \in \mathcal{Z}$ by projecting r onto the linear reward class induced by ϕ using a fixed task-inference dataset $\mathcal{D} = \{s_i\}_{i=1}^N$:

$$\hat{z}_r \in \arg\min_{z\in\mathcal{Z}} \mathbb{E}_{s\sim\mathcal{D}} \left[\left(r(s) - \phi(s)^\top z \right)^2 \right]. \tag{1}$$

Online task inference (Rupf et al., 2026) infers a task embedding from reward observations collected sequentially through interaction with $\mathcal{M}$.

For our theoretical analysis, we consider n episodes of horizon H and a linear mean reward $r(s) = \phi(s)^\top z_r$ with unknown $z_r \in \mathcal{Z}$. We use the corresponding H-step SFs under Assumption 1, which also specifies the reward-noise model. Each episode starts from $s_{k,0} \sim d_0$. At step t, the agent selects $z_{k,t} \in \mathcal{Z}$, executes $a_{k,t} \sim \pi_{z_{k,t}}(\cdot \mid s_{k,t})$, and observes reward $r_{k,t}$ and next state $s_{k,t+1}$. Let $G_k := \sum_{t=0}^{H-1} \gamma^t r_{k,t}$ denote the episode return, and let $G_k^\star$ denote the return obtained by executing π_{z_r} throughout the episode. The objective is to minimize the expected cumulative regret

$$R_n := \mathbb{E}\left[\sum_{k=0}^{n-1} (G_k^\star - G_k) \right].$$

4 Ensemble Sampling with Optimistic Refreshes

4.1 Algorithm

We describe our optimistically refreshed ensemble sampling algorithm for BFMs in Algorithm 1. The algorithm maintains m perturbed estimates of the target task embedding z_r together with a ridge regression estimate $\hat{z}_{kH+t}$ based on the state-reward pairs observed so far. At each step, one ensemble member is sampled uniformly at random to construct $\widetilde{z}_{k,t}$, which is then used to select the task embedding $z_{k,t}$ and the corresponding policy $\pi_{z_{k,t}}$. The ensemble is refreshed at initialization and whenever the regularized feature Gram matrix changes sufficiently (Hwang & Oh, 2026). At each refresh, the perturbed estimates are recentered around an optimistic estimate $w_{k,t}$ obtained from a UCB criterion. This allows the algorithm to combine UCB-based optimism at refresh times with randomized ensemble-based exploration between refreshes.

The Gram matrix captures the uncertainty geometry induced by the observed state features. Let τ denote the most recent refresh time. After the initial refresh, a subsequent refresh is triggered whenever

$$V_{kH+t} \not\preceq 2V_\tau, \tag{2}$$

which limits the relative growth of the Gram matrix within each refresh epoch. At each refresh, we adopt the UCB criterion used by OpTI-BFM (Rupf et al., 2026), based on standard linear bandit confidence bounds (Dani et al., 2008; Abbasi-Yadkori et al., 2011), to select

$$u_{k,t} \in \arg\max_{z\in\mathcal{Z}} \left\{ \langle \psi(s_{k,t},z), \hat{z}_{kH+t} \rangle + \beta_{kH+t} \left\| \psi(s_{k,t},z) \right\|_{V_{kH+t}^{-1}} \right\}. \tag{3}$$

For the selected embedding $u_{k,t}$, we compute an optimistic estimate $w_{k,t}$ of the target task embedding by maximizing $\langle \psi(s_{k,t},u_{k,t}), w \rangle$ over the confidence ellipsoid $\mathcal{C}_{kH+t} := \{w \in \mathbb{R}^d : \|w - \hat{z}_{kH+t}\|_{V_{kH+t}} \le \beta_{kH+t}\}$. The closed-form solution is given in Line 8. Thus, $u_{k,t}$ determines the successor-feature direction, while $w_{k,t}$ attains the corresponding UCB score within the confidence ellipsoid.

At each refresh, we center the perturbed estimates at the UCB-derived estimate $w_{k,t}$ rather than at the ridge estimate $\hat{z}_{kH+t}$. Specifically, for each $j \in [m]$, we independently sample $g_{k,t}^j \sim \mathcal{N}(0, I_d)$ and initialize the perturbation state q_{kH+t}^j so that

$$\hat{z}_{kH+t} + \nu\beta_{kH+t} V_{kH+t}^{-1} q_{kH+t}^j = w_{k,t} + \nu\beta_{kH+t} V_{kH+t}^{-1/2} g_{k,t}^j. \tag{4}$$

Algorithm 1 Ensemble Sampling with Optimistic Refreshes for BFMs (ESOR-BFM)

Require: Pre-trained BFM $(\phi, \{\pi_z\}_{z\in\mathcal{Z}}, \{\psi(\cdot, z)\}_{z\in\mathcal{Z}})$, number of episodes n, horizon H, ensemble size m, regularization $\lambda > 0$, positive confidence radii $\{\beta_t\}_{t\geq 0}$, inflation parameter $\nu > 0$

Initialize $V_0 \leftarrow \lambda I_d$, $b_0 \leftarrow 0$, $\hat{z}_0 \leftarrow 0$, $\tau \leftarrow 0$
for $k = 0, \ldots, n-1$ **do**
  Observe the initial state $s_{k,0} \sim d_0$
  for $t = 0, \ldots, H-1$ **do**
    if $kH + t = 0$ **or** $V_{kH+t} \not\preceq 2V_\tau$ **then** ▷ *Refresh ensemble optimistically*
      Set $\tau \leftarrow kH + t$
      Compute $u_{k,t} \in \arg\max_{z\in\mathcal{Z}} \left\{ \langle \psi(s_{k,t}, z), \hat{z}_{kH+t} \rangle + \beta_{kH+t} \left\| \psi(s_{k,t}, z) \right\|_{V_{kH+t}^{-1}} \right\}$
      Set $w_{k,t} \leftarrow \begin{cases} \hat{z}_{kH+t} + \beta_{kH+t} \dfrac{V_{kH+t}^{-1} \psi(s_{k,t}, u_{k,t})}{\left\| \psi(s_{k,t}, u_{k,t}) \right\|_{V_{kH+t}^{-1}}}, & \text{if } \left\| \psi(s_{k,t}, u_{k,t}) \right\|_{V_{kH+t}^{-1}} > 0, \\ \hat{z}_{kH+t}, & \text{otherwise.} \end{cases}$
      For each $j \in [m]$, sample $g_{k,t}^j \sim \mathcal{N}(0, I_d)$ and set

$$q_{kH+t}^j \leftarrow \frac{1}{\nu\beta_{kH+t}} V_{kH+t}(w_{k,t} - \hat{z}_{kH+t}) + V_{kH+t}^{1/2} g_{k,t}^j$$

    Sample $J_{k,t} \sim \mathrm{Unif}([m])$ and set $\widetilde{z}_{k,t} \leftarrow \hat{z}_{kH+t} + \nu\beta_{kH+t} V_{kH+t}^{-1} q_{kH+t}^{J_{k,t}}$
    Select $z_{k,t} \in \arg\max_{z\in\mathcal{Z}} \langle \psi(s_{k,t}, z), \widetilde{z}_{k,t} \rangle$
    Execute $a_{k,t} \sim \pi_{z_{k,t}}(\cdot \mid s_{k,t})$, and observe $r_{k,t}$ and $s_{k,t+1}$
    Update $V_{kH+t+1} \leftarrow V_{kH+t} + \phi(s_{k,t})\phi(s_{k,t})^\top$
    Update $b_{kH+t+1} \leftarrow b_{kH+t} + \phi(s_{k,t}) r_{k,t}$ and $\hat{z}_{kH+t+1} \leftarrow V_{kH+t+1}^{-1} b_{kH+t+1}$
    For each $j \in [m]$, sample $\xi_{k,t}^j \sim \mathcal{N}(0, 1)$ and update $q_{kH+t+1}^j \leftarrow q_{kH+t}^j + \phi(s_{k,t}) \xi_{k,t}^j$

At each step, the algorithm samples $J_{k,t} \sim \mathrm{Unif}([m])$ and constructs the perturbed estimate $\widetilde{z}_{k,t} = \hat{z}_{kH+t} + \nu\beta_{kH+t} V_{kH+t}^{-1} q_{kH+t}^{J_{k,t}}$. It then selects

$$z_{k,t} \in \arg\max_{z\in\mathcal{Z}} \langle \psi(s_{k,t}, z), \widetilde{z}_{k,t} \rangle \tag{5}$$

and executes $a_{k,t} \sim \pi_{z_{k,t}}(\cdot \mid s_{k,t})$. After each interaction, the ridge estimate is updated using the new state-reward pair, and the perturbation states are incrementally updated using the observed state features and independent Gaussian noise.

Discussion of Optimistic Recentering. Let $\mathcal{F}_{k,t}$ denote the information available at a refresh after computing $u_{k,t}$ and $w_{k,t}$ but before drawing the fresh Gaussian perturbations. The mean predicted return of the UCB-selected policy then satisfies

$$\begin{aligned} \mathbb{E}[\langle \psi(s_{k,t}, u_{k,t}), \widetilde{z}_{k,t} \rangle \mid \mathcal{F}_{k,t}] &= \langle \psi(s_{k,t}, u_{k,t}), w_{k,t} \rangle \\ &= \langle \psi(s_{k,t}, u_{k,t}), \hat{z}_{kH+t} \rangle + \beta_{kH+t} \left\| \psi(s_{k,t}, u_{k,t}) \right\|_{V_{kH+t}^{-1}}, \end{aligned} \tag{6}$$

where the expectation is over the fresh Gaussian perturbations and uniform member selection. For the same regression statistics and perturbation scale, shifting the ensemble center from $\hat{z}_{kH+t}$ to $w_{k,t}$ (i) raises the mean predicted return of the UCB-selected policy by its confidence bonus, (ii) preserves the Gaussian perturbation covariance, maintaining diverse perturbed estimates around the optimistic center, and (iii) allows guidance from the most recent UCB search to be reused across subsequent randomized decisions. Between refreshes, the perturbation states retain the common recentering component, whose contribution to the perturbed estimates evolves with the current inverse Gram matrix and confidence radius as new observations arrive.

4.2 Main Results

Theorem 1 (Regret Guarantee, Informal)**.** *Under some regularity assumptions, in an episodic MDP, if Algorithm 1 (ESOR-BFM) only updates the task embedding $z_{k,t}$ at the start of each episode, it incurs an expected regret of $R_n \leq \widetilde{\mathcal{O}}(d^{3/2}\sqrt{n})$.*

Discussion of Theorem 1. Theorem 1 shows that the episodic variant of ESOR-BFM achieves an expected cumulative regret of $\widetilde{\mathcal{O}}(d^{3/2}\sqrt{n})$, matching the standard dependence of linear ES on the feature dimension d and the number of episodes n (Janz et al., 2026; Hwang & Oh, 2026). The main technical challenge introduced by optimistic recentering is the common, history dependent shift shared by all ensemble members, whereas standard ES analyses consider perturbations centered at the regression estimate. We separate this shift from the persistent centered perturbations and show that its self-normalized magnitude remains bounded within each refresh epoch, thereby preserving the concentration and directional anti concentration properties required for constant probability optimism. At each refresh, the resulting UCB-derived center shifts the ensemble toward a reward coefficient that is optimistic for the selected successor-feature direction, while Gaussian perturbations maintain randomized diversity around this center. Thus, the analysis shows that this optimistic guidance can be incorporated while retaining the regret guarantee of ES. We evaluate its empirical contribution through the ES-BFM ablation, which uses the same refresh criterion and ensemble updates as practical ESOR-BFM but centers the refreshed ensemble at the regression estimate.

4.3 Practical Implementation

For our experiments, we simplify the refresh and policy selection procedures in Algorithm 1 to reduce computation. We initialize the Gaussian ensemble at the regression estimate and apply optimistic recentering at subsequent refreshes. We replace the matrix comparison in Equation (2) with a determinant growth criterion (Abbasi-Yadkori et al., 2011):

$$\log\det V_{kH+t} - \log\det V_{\tau} > \log(1+\alpha_{\mathrm{ER}}),$$

where $\alpha_{\mathrm{ER}} > 0$ is a refresh threshold. The log determinant can be updated incrementally, allowing refresh decisions through a scalar comparison. At each subsequent refresh, we approximate the maximization in Equation (3) by selecting the candidate $u_{k,t}$ with the highest UCB score from a finite set of task embeddings sampled from the confidence ellipsoid $\mathcal{C}_{kH+t}$. We replace the optimistic estimate computation in Line 8 of Algorithm 1 with a direct shift toward the selected candidate:

$$w_{k,t} = \hat{z}_{kH+t} + \rho\left(u_{k,t} - \hat{z}_{kH+t}\right),$$

where $\rho \geq 0$ controls the strength of recentering. We refresh the ensemble using this center in Equation (4). This practical rule uses the UCB-selected candidate to guide recentering, while the exact identity in Equation (6) applies to the theoretical center defined in Line 8. For policy selection, we exploit the BFM pretraining objective, which encourages each policy π_z to maximize returns under the corresponding linear reward $r_z(s) = \phi(s)^\top z$ (Rupf et al., 2026). We therefore approximate the per-step maximization in Equation (5) by directly conditioning the policy on the sampled perturbed task embedding estimate $\widetilde{z}_{k,t}$, normalized to the pretraining sphere as described in Appendix B.2. Consequently, SF evaluations are required only for optimistic search at refresh times.

5 Experiments

Our experiments address the following questions:

(i) Compared with existing online task inference methods, does ESOR-BFM achieve higher online returns and collect more informative data for downstream task inference across diverse environments?

(ii) How does ESOR-BFM trade off task performance against the cost of reward labeling when labels are selectively requested using a threshold on the D-gap (Kiefer & Wolfowitz, 1960; Lattimore & Szepesvári, 2020)?

(iii) Does ESOR-BFM maintain strong performance despite inaccurate successor feature estimates?

(iv) How does the computational cost of ESOR-BFM compare with that of existing methods, and what is its runtime overhead relative to ES-BFM?

Evaluation protocol. We run all experiments with 5 random seeds and report the mean performance. Error bars and shaded regions indicate one standard deviation across seeds.

Environments and datasets. We evaluate online task inference in five environments: `Walker`, `Cheetah`, `Quadruped`, `Jaco Arm`, and `PointMass Maze`, each with four tasks defined by distinct reward functions. For reward-free pretraining, we use offline data from ExORL (Yarats et al., 2022). Further details on the datasets and tasks are provided in Appendix C.1.

Pretrained BFM. We pretrain a BFM using the forward-backward (FB) framework (Touati & Ollivier, 2021). Following the protocol of Touati et al. (2023), we use offline ExORL datasets collected with random network distillation (RND) (Burda et al., 2019; Yarats et al., 2022), containing 10M transitions per environment except for `Jaco Arm`, which contains approximately 6.36M transitions. Further details on pretraining are provided in Appendix B.2.

Baselines. We compare ESOR-BFM with the following baselines for online task inference and adaptation: (i) "Random" selects a task embedding uniformly at random from $\mathcal{Z}$ at each step, (ii) LoLA (Sikchi et al., 2025) is a policy search method originally proposed for fast adaptation, (iii) OpTI-BFM (Rupf et al., 2026) selects task embeddings using UCB, (iv) OpTI-BFM-TS (Rupf et al., 2026) selects task embeddings using Thompson sampling, and (v) ES-BFM adapts linear ensemble sampling with epoch refresh (Hwang & Oh, 2026) to the task embedding space. ES-BFM uses the same refresh criterion and ensemble updates as practical ESOR-BFM, but centers the refreshed ensemble at the regression estimate without an optimistic shift.

5.1 Results under Stationary Rewards

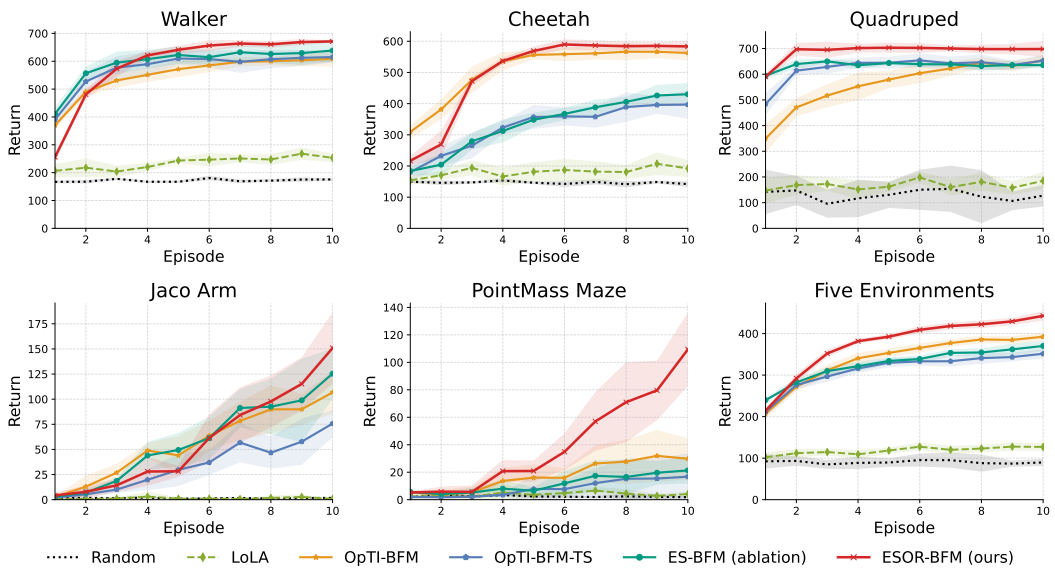


Figure 1: Online task inference performance under stationary rewards. The final panel reports the mean return across the five environments. ESOR-BFM achieves the highest mean final return in all five environments after 10 episodes of interaction.

Question (i): We evaluate online task inference over 10 episodes with a fixed target reward function and reward feedback available at every environment step. Figure 1 demonstrates that ESOR-BFM achieves the highest mean final return in five environments. The learning curves also indicate faster adaptation in several environments. In `Walker` and `Quadruped`, ESOR-BFM reaches high mean returns earlier than OpTI-BFM. In `Cheetah` and `PointMass Maze`, OpTI-BFM-TS shows slower adaptation and a lower mean final return than ESOR-BFM. ES-BFM has a similar or higher mean final return than OpTI-BFM-TS across five environments. ESOR-BFM further improves the mean final return over ES-BFM in each environment, suggesting that optimistic recentering contributes to the gains beyond ensemble sampling and epoch refresh.

We next evaluate how well the data collected by each method support downstream task inference. For each interaction budget, we infer task embeddings from the collected pairs of states and re-

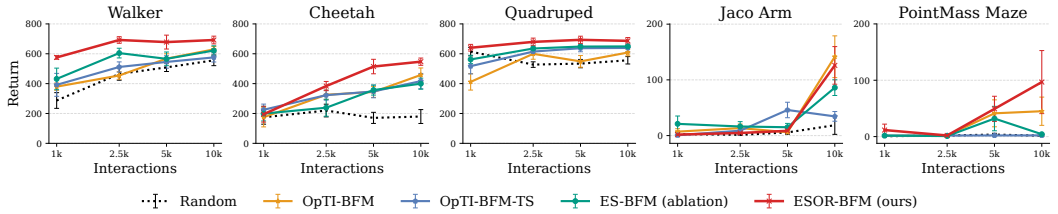


Figure 2: Downstream task inference performance using data collected by each method. At each interaction budget, we infer task embeddings using the same offline inference procedure and report episode returns averaged across target tasks.

wards using the same estimator in Equation (1). We then evaluate the policy conditioned on each inferred embedding under a common protocol and report episode returns averaged across target tasks. Figure 2 shows that, in `Walker` and `Quadruped`, data collected by ESOR-BFM over 2.5k interactions yield higher mean returns than data collected by any baseline over 10k interactions. At 10k interactions, data collected by ESOR-BFM also yield the highest mean return in `Cheetah` and `PointMass Maze`, although variability is substantial in the latter. The comparison with ES-BFM further suggests that optimistic recentering improves the usefulness of collected data for downstream task inference. In `Jaco Arm`, however, data collected by OpTI-BFM yield a higher mean return than those collected by ESOR-BFM at the same interaction budget.

5.2 Results under Selective Reward Labeling

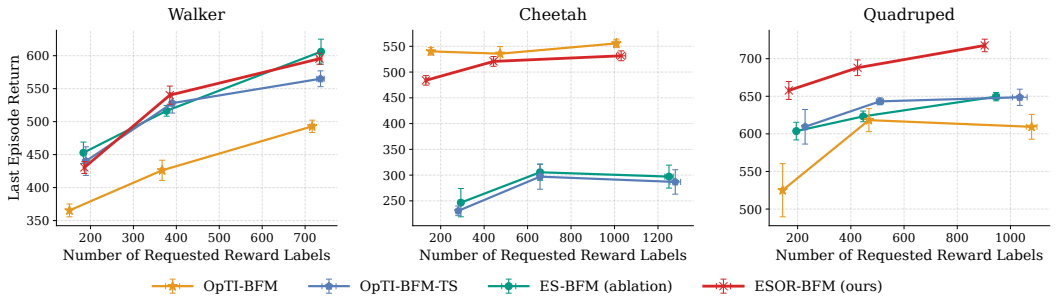


Figure 3: Task performance under selective reward labeling. Curves plot mean final return against the number of requested reward labels for D-gap thresholds $\kappa \in \{0.1, 0.3, 1.0\}$.

Question (ii): We next relax the assumption that reward feedback is available at every environment step and allow the agent to selectively request the reward label associated with each transition. Let t index transitions across episodes, and let ϕ_t denote the feature vector used for reward regression at transition t. V_{t-1} consists of the regularization term and contributions from transitions whose reward labels have been acquired. During online adaptation, we incrementally update the regression sufficient statistics and stored ensemble perturbations only upon reward-label acquisition. Following D-optimal design (Kiefer & Wolfowitz, 1960; Lattimore & Szepesvári, 2020), we define the D-gap as follows:

$$\Delta(\phi_t) = \log\det\left(V_{t-1} + \phi_t\phi_t^\top\right) - \log\det V_{t-1} = \log\left(1 + \|\phi_t\|^2_{V_{t-1}^{-1}}\right). \tag{7}$$

A reward label is requested only when $\Delta(\phi_t) > \kappa$. We evaluate $\kappa \in \{0.1, 0.3, 1.0\}$. Figure 3 plots mean final return against the number of requested reward labels. In these experiments, increasing κ from 0.1 to 1.0 reduces the number of requested labels. Over this threshold range, ESOR-BFM shows a modest decrease in mean final return in `Cheetah` and `Quadruped`, but a larger decrease in `Walker`. In `Quadruped`, ESOR-BFM achieves a mean final return of approximately 660 using 0.17k labels ($\kappa = 1.0$), exceeding the mean final returns achieved by the baselines using 0.95k–1.1k labels. ESOR-BFM remains competitive with OpTI-BFM-TS and ES-BFM in `Walker`. In `Cheetah`, it achieves higher mean final returns than both methods, but lower mean final returns than OpTI-BFM.

5.3 Results under Perturbed SF Estimates

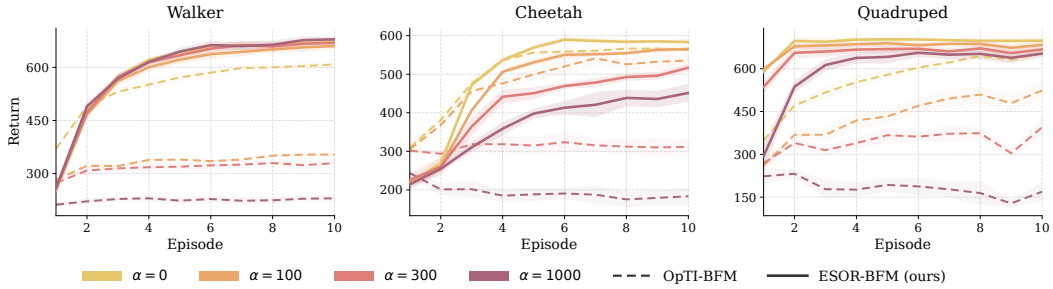


Figure 4: Online task inference performance under perturbed SF estimates. Curves show episode returns averaged across target tasks for each perturbation magnitude α.

Question (iii): Following the experimental framework of Rupf et al. (2026), we evaluate the robustness of ESOR-BFM to inaccurate SF estimates by perturbing the predicted SFs as follows:

$$\widetilde{\psi}(s, z) = \psi(s, z) + \alpha g_\theta(z), \tag{8}$$

where g_θ is a fixed random network whose outputs have unit Euclidean norm. We set $\alpha \in \{0, 100, 300, 1000\}$. We perturb only the SF predictions used during online task inference, keeping the pretrained policy, feature map, and reward regression procedure fixed. We compare our practical implementation of ESOR-BFM, which uses SF estimates only at refresh events, with OpTI-BFM, which uses them at every interaction step. (Further details are provided in Appendix C.2.) Figure 4 shows that the mean final return of ESOR-BFM is less sensitive to SF perturbations than that of OpTI-BFM in all three environments. At $\alpha = 1000$, OpTI-BFM's mean final return remains close to or below its mean return in the first episode, whereas ESOR-BFM continues to improve over episodes. At this perturbation level, ESOR-BFM's mean final return remains close to that obtained without perturbation in `Walker` and `Quadruped`, although the decrease is larger in `Cheetah`.

5.4 Runtime Analysis

Table 1: Agent runtime on DMC `Quadruped`. Time per step is reported as mean $\pm$ standard error across 5 seeds. N_{cand} denotes the number of candidate task embeddings used for optimistic selection, and m denotes the ensemble size.

	OpTI-BFM	**OpTI-BFM-TS**	**ES-BFM** (ablation)	**ESOR-BFM** (ours)
Configuration	$N_{\text{cand}} = 128$	–	$m = 128$	$N_{\text{cand}} = 128,\ m = 128$
Time per step (ms)	4.468 ± 0.048	2.987 ± 0.024	3.736 ± 0.096	4.087 ± 0.009
Frequency (Hz)	224	335	268	245

Question (iv): Finally, we compare the mean runtime per interaction step of ESOR-BFM and the online task inference baselines in `Quadruped`. We exclude the first episode from timing to avoid JIT compilation overhead. Table 1 reports the results. ES-BFM and ESOR-BFM have mean runtimes per interaction step of 3.736 and 4.087 ms, respectively. ESOR-BFM incurs an average overhead of 0.351 ms per interaction step relative to ES-BFM. ESOR-BFM is also faster than OpTI-BFM, which requires 4.468 ms per step. OpTI-BFM-TS has the lowest mean runtime among the compared methods, while ESOR-BFM yields a higher mean final return than both OpTI-BFM-TS and ES-BFM in `Quadruped`, as shown in Figure 1.

6 Conclusion

We proposed ESOR-BFM, which combines ensemble sampling with optimistic recentering for efficient online task inference in BFMs. By maintaining optimistic guidance between refreshes, ESOR-BFM supports randomized exploration while amortizing the cost of optimistic search. For an episodic variant under the stated assumptions, we established a regret bound of $\widetilde{\mathcal{O}}(d^{3/2}\sqrt{n})$. Experiments on ExORL demonstrated improved task performance and label efficiency, reduced sensitivity to SF estimation errors, and lower per-step runtime than OpTI-BFM.

AI USE STATEMENT

In this work, we used generative AI tools for code assistance, running experiments, plot generation, and grammar correction. We did not use generative AI tools for idea generation, hypothesis development, experimental design, theoretical or conceptual development, formulating or proving mathematical claims, or interpreting results. We developed the core codebase ourselves and manually vetted and reviewed all AI-assisted code. We also verified that all plots faithfully and accurately represent our results. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work presents an ensemble-sampling method with optimistic recentering for efficient online task inference in BFMs. All experiments are conducted in simulated environments. Detailed experimental settings and implementation details are provided in Appendix C. It does not involve human subjects, personal data, or deployment in real-world decision-making systems. We are not aware of any specific ethical or societal risks arising directly from the proposed method beyond the general potential for more efficient machine learning systems to be used in applications with societal impacts.

REPRODUCIBILITY STATEMENT

Section 4 specifies the proposed algorithm and its practical implementation, while Appendix A provides the assumptions, formal theorem statement, and complete proof of the regret guarantee. Appendix B.2 details the BFM pretraining architecture and hyperparameters, and Appendices C.1-C.3 describe the datasets, evaluation protocol, experimental settings, baseline methods, and online task-inference hyperparameters. All datasets used in this work are publicly available and cited in the paper. Our source code is included in the supplementary material.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24:2312–2320, 2011.

Siddhant Agarwal, Harshit Sikchi, Peter Stone, and Amy Zhang. Proto successor measure: Representing the behavior space of an rl agent. In *International Conference on Machine Learning*, pp. 566–586. PMLR, 2025.

Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International conference on machine learning*, pp. 127–135. PMLR, 2013.

Marco Bagatella, Matteo Pirotta, Ahmed Touati, Alessandro Lazaric, and Andrea Tirinzoni. Td-jepa: Latent-predictive representations for zero-shot reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 36211–36244, 2026.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P Van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Léonard Blier, Corentin Tallec, and Yann Ollivier. Learning successor states and goal-dependent values: A mathematical viewpoint. *arXiv preprint arXiv:2101.07123*, 2021.

Diana Borsa, Andre Barreto, John Quan, Daniel J. Mankowitz, Hado van Hasselt, Remi Munos, David Silver, and Tom Schaul. Universal successor features approximators. In *International Conference on Learning Representations*, 2019.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019.

Varsha Dani, Thomas P. Hayes, and Sham M. Kakade. Stochastic linear optimization under bandit feedback. In *Proceedings of the 21st Annual Conference on Learning Theory*, pp. 355–366, 2008.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

Taehyun Hwang and Min-hwan Oh. Linear ensemble sampling with smaller ensembles. *arXiv preprint arXiv:2609.13954*, 2026.

Pranaya Jajoo, Harshit Sikchi, Siddhant Agarwal, Amy Zhang, Scott Niekum, and Martha White. Regularized latent dynamics prediction is a strong baseline for behavioral foundation models. In *The Fourteenth International Conference on Learning Representations*, 2026.

David Janz, Alexander Litvak, and Csaba Szepesvari. Ensemble sampling for linear bandits: small ensembles suffice. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

David Janz, Arya Akhavan, and Csaba Szepesvári. Sharp analysis of linear ensemble sampling. In Steve Hanneke and Tor Lattimore (eds.), *Proceedings of Thirty Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 3716–3750. PMLR, 29 Jun–03 Jul 2026. URL https://proceedings.mlr.press/v336/janz26a.html.

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

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6980.

Tor Lattimore and Csaba Szepesvári. Bandit algorithms. *Cambridge University*, 355, 2020.

Harin Lee and Min-hwan Oh. Improved regret of linear ensemble sampling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Xiuyuan Lu and Benjamin Van Roy. Ensemble sampling. *Advances in neural information processing systems*, 30, 2017.

Xiuyuan Lu, Zheng Wen, and Branislav Kveton. Efficient online recommendation via low-rank ensemble sampling. In *Proceedings of the 12th ACM conference on recommender systems*, pp. 460–464, 2018.

Chen Ma, Dylan R Ashley, Junfeng Wen, and Yoshua Bengio. Universal successor features for transfer reinforcement learning. *arXiv preprint arXiv:2001.04025*, 2020.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29, 2016.

Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

Ian Osband, Benjamin Van Roy, Daniel J Russo, Zheng Wen, et al. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 20(124):1–62, 2019.

Seohong Park, Tobias Kreiman, and Sergey Levine. Foundation policies with hilbert representations. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 39737–39761, 2024.

Chao Qin, Zheng Wen, Xiuyuan Lu, and Benjamin Van Roy. An analysis of ensemble sampling. *Advances in Neural Information Processing Systems*, 35:21602–21614, 2022.

Thomas Rupf, Marco Bagatella, Marin Vlastelica, and Andreas Krause. Optimistic task inference for behavior foundation models. In *International Conference on Learning Representations*, volume 2026, pp. 64938–64971, 2026.

Harshit Sikchi, Andrea Tirinzoni, Ahmed Touati, Yingchen Xu, Anssi Kanervisto, Scott Niekum, Amy Zhang, Alessandro Lazaric, and Matteo Pirotta. Fast adaptation with behavioral foundation models. In *Reinforcement Learning Conference*, 2025.

Andrea Tirinzoni, Ahmed Touati, Jesse Farebrother, Mateusz Guzek, Anssi Kanervisto, Yingchen Xu, Alessandro Lazaric, and Matteo Pirotta. Zero-shot whole-body humanoid control via behavioral foundation models. In *International Conference on Learning Representations*, volume 2025, pp. 21693–21748, 2025.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. *Advances in Neural Information Processing Systems*, 34:13–23, 2021.

Ahmed Touati, Jérémy Rapin, and Yann Ollivier. Does zero-shot reinforcement learning exist? In *The Eleventh International Conference on Learning Representations*, 2023.

Yifan Wu, George Tucker, and Ofir Nachum. The laplacian in RL: Learning representations with efficient approximations. In *International Conference on Learning Representations*, 2019.

Denis Yarats, David Brandfonbrener, Hao Liu, Michael Laskin, Pieter Abbeel, Alessandro Lazaric, and Lerrel Pinto. Don't change the algorithm, change the data: Exploratory data for offline reinforcement learning. *arXiv preprint arXiv:2201.13425*, 2022.

Jie Zhou, Botao Hao, Zheng Wen, Jingfei Zhang, and Will Wei Sun. Stochastic low-rank tensor bandits for multi-dimensional online decision making. *Journal of the American Statistical Association*, 120(549):198–211, 2025.

Zheqing Zhu and Benjamin Van Roy. Deep exploration for recommendation systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 963–970, 2023.

CONTENTS OF APPENDIX

A Regret Guarantees for ESOR-BFM **14**

A.1 Formal Statement of Theorem 1 14

A.2 Proof of Main Theorem 16

A.2.1 Confidence of the reward embedding 16

A.2.2 Persistent ensemble within refresh epochs 16

A.2.3 Optimistic refresh preserves anti-concentration 17

A.2.4 One-episode regret 17

A.2.5 Cumulative regret 18

A.3 Proofs of Technical Lemmas 19

A.3.1 Proof of Lemma 2 19

A.3.2 Proof of Lemma 3 20

A.3.3 Proof of Lemma 4 20

A.3.4 Proof of Lemma 5 21

A.4 Auxiliary Lemmas 22

B Pretraining Protocol for BFMs **22**

B.1 Forward-Backward Representations 22

B.2 Datasets and Implementation Details 23

C Additional Experimental Details **24**

C.1 ExORL Benchmarks 24

C.2 Experimental Results 26

C.3 Online Task Inference Hyperparameters 29

A REGRET GUARANTEES FOR ESOR-BFM

We give the formal statement and proof of Theorem 1 for the episodic variant of ESOR-BFM, in which the selected task embedding is kept fixed within each episode. The analysis builds on recent theoretical advances in ensemble sampling for linear bandits (Janz et al., 2026; Hwang & Oh, 2026) together with the reward-level-to-return-level comparison technique of Rupf et al. (2026). Unlike standard linear ensemble sampling (Hwang & Oh, 2026), however, ESOR-BFM refreshes the ensemble around an optimistic parameter, introducing a common shift across ensemble members. Our additional analysis shows that this shift preserves the anti-concentration required for constant-probability optimism.

We organize the appendix into three parts. First, we specify the episodic algorithm, assumptions, regret criterion, and the formal statement of Theorem 1. Second, we establish the main regret bound using the required confidence, ensemble, and reward-level-to-return-level arguments. Third, we provide the proofs of the auxiliary technical lemmas.

A.1 FORMAL STATEMENT OF THEOREM 1

Episodic Update Rule. We begin by introducing the episodic version of ESOR-BFM analyzed in this section. We consider an episodic interaction protocol with horizon H, and index episodes by $k \in [n]$ and steps within each episode by $t \in \{0, \dots, H-1\}$. At the beginning of episode k, the learner selects a task embedding $z_k \in \mathcal{Z}$ and executes the corresponding BFM policy π_{z_k} for the entire episode.

Let $\phi_{k,t} := \phi(s_{k,t})$, and let the unknown test-time reward be parameterized by $z_r \in \mathbb{R}^d$. From the reward-level observations collected up to and including episode k, we define

$$\Sigma_k = \lambda I_d + \sum_{\ell=1}^{k} \sum_{t=0}^{H-1} \phi_{\ell,t} \phi_{\ell,t}^\top , \quad b_k = \sum_{l=1}^{k} \sum_{t=0}^{H-1} \phi_{l,t} r_{l,t} , \quad \hat{z}_k = \Sigma_k^{-1} b_k .$$

Accordingly, the decision at episode k is based on Σ_{k-1} and $\hat{z}_{k-1}$, with $\Sigma_0 = \lambda I_d$ and $\hat{z}_0 = 0$.

The episodic version of ESOR-BFM maintains a persistent ensemble of perturbations and refreshes the ensemble whenever the accumulated information has changed sufficiently relative to the Gram matrix at the most recent refresh. At a refresh, the ensemble is re-centered around an optimistic parameter in the current confidence set; between refreshes, the perturbations evolve persistently using the newly observed reward-level features. The complete procedure is summarized in Algorithm 2.

Assumptions. We introduce the assumptions used in our analysis, which are also adopted in the episodic regret analysis of Rupf et al. (2026).

Assumption 1. *The following conditions hold.*

(A1) ***Perfect USF.*** *For every $(s, a, z) \in \mathcal{S} \times \mathcal{A} \times \mathcal{Z}$,*

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

and

$$\pi_z(a \mid s) > 0 \quad \Longrightarrow \quad a \in \arg\max_{a' \in \mathcal{A}} \psi(s, a', z)^\top z.$$

We write $\psi(s, z) := \mathbb{E}_{a \sim \pi_z(\cdot|s)}[\psi(s, a, z)]$.

(A2) ***Linear rewards.*** *The test-time reward r lies in the span of the features ϕ up to i.i.d. mean-zero 1-sub-Gaussian noise $\eta_{k,t}$, i.e.,*

$$r_{k,t} = \langle \phi(s_{k,t}), z_r \rangle + \eta_{k,t}$$

for some $z_r \in \mathcal{Z}$.

(A3) ***Optimization oracle.*** *All optimization problems in Algorithm 2 are solved exactly.*

(A4) ***Boundedness.*** *For every $s \in \mathcal{S}$, $\|\phi(s)\|_2 \le 1$, and $\|z_r\|_2 \le S$ for a constant $S > 0$.*

Algorithm 2 Episodic variant of ESOR-BFM

Require: Pre-trained BFM $(\phi, \{\pi_z\}_{z\in\mathcal{Z}}, \{\psi(\cdot, z)\}_{z\in\mathcal{Z}})$; number of episodes n; horizon H; ensemble size m; regularization $\lambda > 0$; confidence radii $\{\beta_k\}_{k\geq 1}$; inflation $\nu > 0$

Initialize $\Sigma_0 \leftarrow \lambda I_d$, $b_0 \leftarrow 0$, $\hat{z}_0 \leftarrow 0$, $\tau \leftarrow 0$

for $k = 1, \ldots, n$ **do**

  Observe $s_{k,0} \sim d_0$

  if $k = 1$ **or** $\Sigma_{k-1} \not\preceq 2\Sigma_\tau$ **then** ▷ Optimistic refresh

    $\tau \leftarrow k - 1$

    $u_k \in \arg\max_{z\in\mathcal{Z}} \left\{ \langle \psi(s_{k,0}, z), \hat{z}_{k-1} \rangle + \beta_k \|\psi(s_{k,0}, z)\|_{\Sigma_{k-1}^{-1}} \right\}$

    $w_k \leftarrow \begin{cases} \hat{z}_{k-1} + \beta_k \dfrac{\Sigma_{k-1}^{-1}\psi(s_{k,0}, u_k)}{\|\psi(s_{k,0}, u_k)\|_{\Sigma_{k-1}^{-1}}}, & \|\psi(s_{k,0}, u_k)\|_{\Sigma_{k-1}^{-1}} > 0, \\ \hat{z}_{k-1}, & \text{otherwise} \end{cases}$

    For each $j \in [m]$, sample $g_k^j \sim \mathcal{N}(0, I_d)$ and set $q_{k-1}^j \leftarrow \dfrac{\Sigma_{k-1}(w_k - \hat{z}_{k-1})}{\nu\beta_k} + \Sigma_{k-1}^{1/2} g_k^j$

  Sample $J_k \sim \mathrm{Unif}([m])$ and set $\widetilde{z}_k \leftarrow \hat{z}_{k-1} + \nu\beta_k \Sigma_{k-1}^{-1} q_{k-1}^{J_k}$

  $z_k \in \arg\max_{z\in\mathcal{Z}} \langle \psi(s_{k,0}, z), \widetilde{z}_k \rangle$

  for $t = 0, \ldots, H-1$ **do**

    Sample $a_{k,t} \sim \pi_{z_k}(\cdot \mid s_{k,t})$ and observe $r_{k,t}, s_{k,t+1}$

  $\Sigma_k \leftarrow \Sigma_{k-1} + \sum_{t=0}^{H-1} \phi_{k,t}\phi_{k,t}^\top, \quad b_k \leftarrow b_{k-1} + \sum_{t=0}^{H-1} \phi_{k,t} r_{k,t}, \quad \hat{z}_k \leftarrow \Sigma_k^{-1} b_k$

  For each $j \in [m]$, sample $\{\xi_{k,t}^j\}_{t=0}^{H-1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$ and set $q_k^j \leftarrow q_{k-1}^j + \sum_{t=0}^{H-1} \phi_{k,t}\xi_{k,t}^j$

Regret. For episode k, let

$$G_{k,z} := \mathbb{E}\left[\sum_{t=0}^{H-1} \gamma^t r_{k,t} \,\middle|\, s_{k,0}, \pi_z \right]$$

denote the expected discounted return of π_z. We define the cumulative expected regret over n episodes as

$$R_n := \mathbb{E}\left[\sum_{k=1}^{n} (G_{k,z_r} - G_{k,z_k}) \right].$$

We are now ready to state the formal version of Theorem 1. The theorem provides a theoretical justification for the exploration mechanism of ESOR-BFM, complementing its empirical performance with a sublinear cumulative regret guarantee.

Theorem 2 (Formal statement of Theorem 1)**.** *Suppose Assumption 1 holds. Fix $\delta \in (0, 1/4)$. Set the input parameters of Algorithm 2 as follows:*

$$\lambda = 80\,, \quad \nu = 80\,, \quad \beta_k = \sqrt{\lambda}S + \sqrt{d\log\left(1 + \frac{(k-1)H}{d\lambda}\right) + 2\log\frac{1}{\delta}}\,,$$

$$m := \left\lceil C\left[d\log\left(d + \log\frac{2\bar{R}}{\delta}\right) + \log\frac{2\bar{R}}{\delta} \right] \right\rceil,$$

where $C > 0$ is a sufficiently large universal constant and

$$\bar{R} := \left\lceil 2 + d\log_2\left(1 + \frac{nH}{d\lambda}\right) \right\rceil.$$

Then the cumulative expected regret satisfies

$$R_n \leq 30\alpha_H(\nu\Gamma_\delta + 2)\beta_n \sqrt{2nd\log\left(1 + \frac{nD_H^2}{d\lambda}\right)} + 4\delta nSD_H\,,$$

where

$$\Gamma_\delta := \sqrt{d} + \sqrt{2\log\frac{8m\bar{R}}{\delta}} + \sqrt{d\log 2 + 2\log\frac{8m\bar{R}}{\delta}},$$
$$D_H := \sum_{t=0}^{H-1}\gamma^t = \frac{1-\gamma^H}{1-\gamma}, \qquad c_H := \sum_{t=0}^{H-1}\gamma^{2t} = \frac{1-\gamma^{2H}}{1-\gamma^2},$$
$$\alpha_H := \max\left\{\frac{D_H}{\sqrt{\lambda}}, \sqrt{c_H}\right\}.$$

In particular, for $n \geq 5$, choosing $\delta = 1/n$ gives

$$R_n = \widetilde{\mathcal{O}}\left(d^{3/2}\sqrt{n}\right).$$

A.2 Proof of Main Theorem

We first introduce the technical ingredients. Proofs of the lemmas are given in Appendix A.3.

A.2.1 Confidence of the reward embedding

Lemma 1 (Theorem 1 of Rupf et al. (2026))**.** *Under Assumption 1, let*

$$\mathcal{C}_k := \left\{w \in \mathbb{R}^d : \|w - \hat{z}_{k-1}\|_{\Sigma_{k-1}} \leq \beta_k\right\},$$

where β_k is defined in Theorem 2. Then

$$\mathbb{P}\big(\forall k \geq 1, z_r \in \mathcal{C}_k\big) \geq 1 - \delta.$$

Lemma 1 ensures that, with high probability, the confidence set $\mathcal{C}_k$ contains the true reward embedding z_r at every episode. This allows us to use $\mathcal{C}_k$ as a valid uncertainty set over reward embeddings when analyzing the exploration behavior of ESOR-BFM.

A.2.2 Persistent ensemble within refresh epochs

We next characterize how the ensemble evolves between successive optimistic refreshes. Let τ be the refresh index maintained by Algorithm 2, so that the current refresh epoch begins at episode $\tau+1$. For episode k in this epoch, define the refresh shift and the corresponding centered perturbation by

$$h_\tau := \frac{\Sigma_\tau(w_{\tau+1} - \hat{z}_\tau)}{\nu\beta_{\tau+1}}, \qquad \bar{q}^j_{k-1} := q^j_{k-1} - h_\tau.$$

Unrolling the ensemble updates gives

$$\bar{q}^j_{k-1} = \Sigma_\tau^{1/2} g^j_{\tau+1} + \sum_{\ell=\tau+1}^{k-1}\sum_{t=0}^{H-1}\phi_{\ell,t}\xi^j_{\ell,t}. \tag{9}$$

Thus, after removing the fixed optimistic shift, each ensemble member retains the random perturbation sampled at the beginning of the refresh epoch and updates it incrementally as new reward observations are collected. In particular, it starts from $\Sigma_\tau^{1/2} g^j_{\tau+1}$ and subsequently accumulates the Gaussian increments $\phi_{\ell,t}\xi^j_{\ell,t}$, rather than being independently resampled at every episode. Each increment is conditionally Gaussian given the preceding interaction history, since $\phi_{\ell,t}$ is determined before $\xi^j_{\ell,t}$ is sampled. This persistence allows the ensemble to retain directional diversity across episodes while adapting to newly collected data.

Our next result formalizes two properties of the centered ensemble that will be used in the regret analysis. First, every centered perturbation remains uniformly controlled in self-normalized norm. Second, in every direction, a constant fraction of ensemble members maintains a sufficiently large positive projection. The first property controls the magnitude of the sampled reward parameters, while the second provides the anti-concentration needed for constant-probability optimism.

Lemma 2 (Uniform control of centered ensemble perturbations). *Fix $\delta \in (0,1)$. Let Γ_δ and $\bar{R}$ be defined in Theorem 2, and set*

$$N_\delta := \left(240\sqrt{2}\,\Gamma_\delta\right)^d.$$

If

$$m \geq 40 \log\left(\frac{4\lceil 250 \log 2\rceil \bar{R} N_\delta}{\delta}\right), \tag{10}$$

then, with probability at least $1-\delta$, the following two properties hold simultaneously:

$$\max_{k\in[n]} \max_{j\in[m]} \|\bar{q}_{k-1}^j\|_{\Sigma_{k-1}^{-1}} \leq \Gamma_\delta, \tag{11}$$

$$\min_{k\in[n]} \inf_{u\neq 0} \frac{1}{m}\sum_{j=1}^m \mathbb{1}\left\{\frac{\langle u, \bar{q}_{k-1}^j\rangle}{\|u\|_{\Sigma_{k-1}}} \geq \frac{1}{40}\right\} \geq \frac{1}{10}. \tag{12}$$

A.2.3 Optimistic refresh preserves anti-concentration

The algorithm uses $q_{k-1}^j = \bar{q}_{k-1}^j + h_\tau$, not the centered perturbation alone. We therefore control how much the common refresh shift can reduce a normalized projection.

Lemma 3 (Self-normalized bound on the refresh shift). *For every episode k in the refresh epoch beginning at $\tau+1$,*

$$\|h_\tau\|_{\Sigma_{k-1}^{-1}} \leq \frac{1}{\nu}.$$

Consequently, for every $u \in \mathbb{R}^d \setminus \{0\}$,

$$\frac{\langle u, h_\tau\rangle}{\|u\|_{\Sigma_{k-1}}} \geq -\frac{1}{\nu}.$$

This bound is deterministic: it follows from the construction of $w_{\tau+1}$ and the monotonicity of Σ_k. Combining this bound of Lemma 3 with the centered anti-concentration guarantee in Lemma 2 shows that the shift can reduce the normalized projection of each ensemble member by at most $1/\nu$.

Lemma 4 (Uniform anti-concentration of shifted ensemble perturbations). *Suppose that Lemma 2 holds and set $\nu = 80$. Then*

$$\min_{k\in[n]} \inf_{u\neq 0} \frac{1}{m}\sum_{j=1}^m \mathbb{1}\left\{\frac{\langle u, q_{k-1}^j\rangle}{\|u\|_{\Sigma_{k-1}}} \geq \frac{1}{80}\right\} \geq \frac{1}{10}.$$

Lemma 4 shows that the optimistic refresh preserves the anti-concentration of the perturbations actually used by Algorithm 2. In particular, the common shift introduced at a refresh is small enough that every ensemble member satisfying the centered exceedance condition also satisfies the shifted one. Hence, a constant fraction of the shifted ensemble retains a sufficiently large positive projection in every direction, which is the key ingredient for establishing constant-probability optimism of the sampled ensemble member.

A.2.4 One-episode regret

We next show how stochastic optimism and concentration of the ensemble reward parameters translate into a regret bound for a single episode.

Lemma 5 (One-episode regret under stochastic optimism). *Let $\mathcal{A}_k$ contain all information available immediately before sampling J_k. Note that conditional on $\mathcal{A}_k$, $J_k \sim \mathrm{Unif}([m])$. For each $j \in [m]$, define*

$$\widetilde{z}_k^j := \hat{z}_{k-1} + \nu\beta_k \Sigma_{k-1}^{-1} q_{k-1}^j \quad \textit{and} \quad z_k^j \in \operatorname*{argmax}_{z\in\mathcal{Z}} \left\langle \psi(s_{k,0}, z), \widetilde{z}_k^j \right\rangle.$$

Define the conditional probability

$$p_k := \mathbb{P}\left(\langle \psi(s_{k,0}, z_k), \widetilde{z}_k\rangle \geq \langle \psi(s_{k,0}, z_r), z_r\rangle \,|\, \mathcal{A}_k\right).$$

Suppose that $p_k > 0$ and that, for some $B_k > 0$,

$$\max_{j\in[m]} \|\widetilde{z}_k^j - z_r\|_{\Sigma_{k-1}} \le B_k\,.$$

Then

$$\mathbb{E}\left[G_{k,z_r} - G_{k,z_k} \,|\, \mathcal{A}_k\right] \le \frac{3B_k}{p_k}\mathbb{E}\left[\|\psi(s_{k,0},z_k)\|_{\Sigma_{k-1}^{-1}} \,\middle|\, \mathcal{A}_k\right]\,.$$

A.2.5 CUMULATIVE REGRET

We are now ready to prove Theorem 2.

Proof of Theorem 2. Let $\mathcal{E}$ be the intersection of the uniform confidence and centered-ensemble events defined in Lemma 1 and 2. Lemmas 1 and 2 give $\mathbb{P}(\mathcal{E}^c) \le 2\delta$. Write $e_k := G_{k,z_r} - G_{k,z_k}$. On $\mathcal{E}$, Lemma 5 and $\beta_k \le \beta_n$ imply

$$\mathbb{E}[e_k \mid \mathcal{A}_k] \le 30(\nu\Gamma_\delta + 2)\beta_n\mathbb{E}\left[\|\psi(s_{k,0},z_k)\|_{\Sigma_{k-1}^{-1}} \,\middle|\, \mathcal{A}_k\right]. \tag{13}$$

Define the empirical successor features and their Gram matrices by

$$\widetilde{\psi}_k := \sum_{t=0}^{H-1}\gamma^t\phi_{k,t}, \qquad W_k := \lambda I_d + \sum_{\ell=1}^{k}\widetilde{\psi}_\ell\widetilde{\psi}_\ell^\top, \qquad W_0 = \lambda I_d.$$

By (A1) and episode-wise commitment to π_{z_k},

$$\psi(s_{k,0},z_k) = \mathbb{E}[\widetilde{\psi}_k \mid \mathcal{A}_k, J_k].$$

Since Σ_{k-1} is $\mathcal{A}_k$-measurable, conditional Jensen and then the tower property give

$$\begin{aligned}\|\psi(s_{k,0},z_k)\|_{\Sigma_{k-1}^{-1}} &\le \mathbb{E}[\|\widetilde{\psi}_k\|_{\Sigma_{k-1}^{-1}} \mid \mathcal{A}_k, J_k],\\ \mathbb{E}[\|\psi(s_{k,0},z_k)\|_{\Sigma_{k-1}^{-1}} \mid \mathcal{A}_k] &\le \mathbb{E}[\|\widetilde{\psi}_k\|_{\Sigma_{k-1}^{-1}} \mid \mathcal{A}_k].\end{aligned} \tag{14}$$

The second line averages the first over J_k conditional on $\mathcal{A}_k$.

Propositions 2–4 of Rupf et al. (2026), in our notation, give

$$\widetilde{\psi}_k\widetilde{\psi}_k^\top \preceq c_H\sum_{t=0}^{H-1}\phi_{k,t}\phi_{k,t}^\top, \qquad \Sigma_k \succeq c_H^{-1}W_k, \qquad \|v\|_{\Sigma_k^{-1}} \le \sqrt{c_H}\|v\|_{W_k^{-1}} \quad (v\in\mathbb{R}^d).$$

These comparisons also hold at $k = 0$, since $c_H \ge 1$. Moreover, $\|\widetilde{\psi}_k\|_2 \le D_H$ and $\Sigma_{k-1} \succeq \lambda I_d$, so, pathwise,

$$\|\widetilde{\psi}_k\|_{\Sigma_{k-1}^{-1}} \le \min\left\{\frac{D_H}{\sqrt{\lambda}}, \sqrt{c_H}\|\widetilde{\psi}_k\|_{W_{k-1}^{-1}}\right\} \le \alpha_H\min\{1, \|\widetilde{\psi}_k\|_{W_{k-1}^{-1}}\}.$$

Combining this with (14) yields

$$\mathbb{E}[\|\psi(s_{k,0},z_k)\|_{\Sigma_{k-1}^{-1}} \mid \mathcal{A}_k] \le \alpha_H\mathbb{E}\left[\min\{1, \|\widetilde{\psi}_k\|_{W_{k-1}^{-1}}\} \,\middle|\, \mathcal{A}_k\right]. \tag{15}$$

We account for failed confidence or ensemble bounds without conditioning the interaction on the entire-horizon event $\mathcal{E}$. By (A2)–(A4), $|G_{k,z}| \le SD_H$ for every candidate and the oracle, and hence $\mathbb{E}[e_k \mid \mathcal{A}_k] \le 2SD_H$ everywhere. Using the tower property under the original probability law,

$$\begin{aligned}R_n &= \sum_{k=1}^{n}\mathbb{E}\left[\mathbb{1}\{\mathcal{E}\}\mathbb{E}[e_k \mid \mathcal{A}_k]\right] + \sum_{k=1}^{n}\mathbb{E}\left[\mathbb{1}\{\mathcal{E}^c\}\mathbb{E}[e_k \mid \mathcal{A}_k]\right]\\ &\le 30\alpha_H(\nu\Gamma_\delta+2)\beta_n\sum_{k=1}^{n}\mathbb{E}\left[\mathbb{1}\{\mathcal{E}\}\mathbb{E}\left[\min\{1,\|\widetilde{\psi}_k\|_{W_{k-1}^{-1}}\} \,\middle|\, \mathcal{A}_k\right]\right] + 2nSD_H\mathbb{P}(\mathcal{E}^c)\\ &\le 30\alpha_H(\nu\Gamma_\delta+2)\beta_n\mathbb{E}\left[\sum_{k=1}^{n}\min\{1,\|\widetilde{\psi}_k\|_{W_{k-1}^{-1}}\}\right] + 4\delta nSD_H\,.\end{aligned} \tag{16}$$

Finally, the elliptical potential lemma (Lemma 9) and Cauchy–Schwarz give,

$$\sum_{k=1}^{n} \min\{1, \|\widetilde{\psi}_k\|_{W_{k-1}^{-1}}\} \le \sqrt{n \sum_{k=1}^{n} \min\{1, \|\widetilde{\psi}_k\|^2_{W_{k-1}^{-1}}\}}$$
$$\le \sqrt{2n \log \frac{\det W_n}{\lambda^d}} \le \sqrt{2nd \log\left(1 + \frac{nD_H^2}{d\lambda}\right)},$$

where the last inequality follows from the determinant–trace inequality (Lemma 8) and $\|\widetilde{\psi}_k\|_2 \le D_H$. Substitution into (16) proves the claimed regret bound.

For $n \ge 5$, take $\delta = 1/n$. Under the ensemble-size choice of $m = \Theta(d \log d + d \log\log n + \log n)$, we have $\Gamma_\delta = \widetilde{\mathcal{O}}(\sqrt{d})$ and $\beta_n = \widetilde{\mathcal{O}}(\sqrt{d})$ for fixed H, γ, λ, S. The failure contribution is $4SD_H$, so $R_n = \widetilde{\mathcal{O}}(d^{3/2}\sqrt{n})$. □

A.3 Proofs of Technical Lemmas

A.3.1 Proof of Lemma 2

We use the following two results of Hwang & Oh (2026), restated for predictable Gaussian perturbations rather than bandit actions. Their proofs depend on the process and Gram-matrix conditions below, not on the reward function or the action-selection objective.

Lemma 6 (Lemma 8 of Hwang & Oh (2026))**.** *Consider at most $\bar{R}$ refresh epochs, where $\bar{R}$ is deterministic. At the start of epoch r, let $V_{r,0} \succ 0$ be measurable with respect to the pre-refresh history. For local times $t \ge 0$, define*

$$V_{r,t} := V_{r,0} + \sum_{s=1}^{t} x_{r,s} x_{r,s}^\top, \qquad S_{r,t}^j := V_{r,0}^{1/2} g_r^j + \sum_{s=1}^{t} x_{r,s} \xi_{r,s}^j.$$

Here $g_r^j \sim \mathcal{N}(0, I_d)$ are fresh independent draws at the refresh. Before each update, $x_{r,s}$ is determined, and $(\xi_{r,s}^j)_{j=1}^m \sim \mathcal{N}(0, I_m)$ is sampled independently of all information then available. Suppose that, for every local time under consideration,

$$V_{r,0} \preceq V_{r,t} \preceq (1+\alpha) V_{r,0}, \qquad \alpha > 0.$$

For $\delta' \in (0,1)$, define

$$\Gamma(\delta', \alpha) := \sqrt{d} + \sqrt{2 \log \frac{4m\bar{R}}{\delta'}} + \sqrt{d \log(1+\alpha) + 2 \log \frac{4m\bar{R}}{\delta'}}.$$

With probability at least $1 - \delta'$, simultaneously for all epochs, local times under consideration, and $j \in [m]$,

$$\|S_{r,t}^j\|_{V_{r,t}^{-1}} \le \Gamma(\delta', \alpha).$$

Lemma 7 (Lemma 2 of Hwang & Oh (2026))**.** *Consider the process and conditions of Lemma 6. Fix $\nu' = 40$ and $\delta' \in (0,1)$, and define*

$$\kappa_\alpha := \lceil 250 \log(1+\alpha) \rceil, \qquad N_{\delta',\alpha} := \left(6\sqrt{1+\alpha}\, \nu' \Gamma(\delta', \alpha)\right)^d.$$

If $m \ge 40 \log(2\kappa_\alpha \bar{R} N_{\delta',\alpha} / \delta')$, then, with probability at least $1 - \delta'$, simultaneously for all epochs and local times under consideration,

$$\inf_{u \ne 0} \frac{1}{m} \sum_{j=1}^{m} \mathbb{1}\left\{ \frac{\langle u, S_{r,t}^j \rangle}{\|u\|_{V_{r,t}}} \ge \frac{1}{\nu'} \right\} \ge \frac{1}{10}.$$

Proof of Lemma 2. Fix a refresh epoch beginning at episode $\tau + 1$. For every episode k in this epoch, the centered ensemble satisfies

$$\bar{q}_{k-1}^j = \Sigma_\tau^{1/2} g_{\tau+1}^j + \sum_{\ell=\tau+1}^{k-1} \sum_{t=0}^{H-1} \phi_{\ell,t} \xi_{\ell,t}^j.$$

Since $\phi_{\ell,t}$ is observed before $\xi^j_{\ell,t} \sim \mathcal{N}(0,1)$ is sampled, each $\phi_{\ell,t}\xi^j_{\ell,t}$ is a predictable Gaussian increment.

For the r-th refresh epoch, identify the process in Lemmas 6 and 7 by $V_{r,0} = \Sigma_\tau$ and $g^j_r = g^j_{\tau+1}$. For the reward-level observation (ℓ, t), let $i = (\ell - \tau - 1)H + t + 1$ and set $x_{r,i} = \phi_{\ell,t}$ and $\xi^j_{r,i} = \xi^j_{\ell,t}$.

Let $i_k := (k - 1 - \tau)H$ denote the number of reward-level updates accumulated before episode k. Then, at the beginning of episode k,

$$S^j_{r,i_k} = \bar{q}^j_{k-1}, \qquad V_{r,i_k} = \Sigma_{k-1}.$$

For every decision made using the current ensemble, the refresh rule ensures

$$\Sigma_\tau \preceq \Sigma_{k-1} \preceq 2\Sigma_\tau.$$

Since the Gram matrix is updated by positive-semidefinite increments, the same bound holds for every intermediate matrix contributing to such a decision boundary. Hence the covariance-growth condition in Lemma 6 holds with $\alpha = 1$. Moreover, the number of refresh epochs is at most $\bar{R}$ (Lemma 4 in Hwang & Oh (2026)).

Apply Lemma 6 with $\alpha = 1$ and $\delta' = \delta/2$. Since $\Gamma(\delta/2, 1) = \Gamma_\delta$, with probability at least $1 - \delta/2$,

$$\max_{k\in[n]} \max_{j\in[m]} \|\bar{q}^j_{k-1}\|_{\Sigma^{-1}_{k-1}} \le \Gamma_\delta.$$

Next, apply Lemma 7 with the same α and δ', and with threshold parameter $\nu' = 40$. Then $\kappa_1 = \lceil 250 \log 2 \rceil$ and $N_{\delta/2,1} = (240\sqrt{2}\,\Gamma_\delta)^d = N_\delta$. Thus its ensemble-size requirement becomes

$$m \ge 40 \log\left(\frac{4\lceil 250 \log 2\rceil \bar{R} N_\delta}{\delta}\right),$$

which is exactly Eq. (10). Therefore, with probability at least $1 - \delta/2$,

$$\min_{k\in[n]} \inf_{u\neq 0} \frac{1}{m} \sum_{j=1}^{m} \mathbb{1}\left\{\frac{\langle u, \bar{q}^j_{k-1}\rangle}{\|u\|_{\Sigma_{k-1}}} \ge \frac{1}{40}\right\} \ge \frac{1}{10}.$$

A union bound proves Eqs. (11)–(12) simultaneously with probability at least $1 - \delta$. □

A.3.2 Proof of Lemma 3

Proof of Lemma 3. At the refresh beginning episode $\tau + 1$, the definition of $w_{\tau+1}$ gives

$$\|w_{\tau+1} - \hat{z}_\tau\|_{\Sigma_\tau} \le \beta_{\tau+1}.$$

Indeed, the norm equals $\beta_{\tau+1}$ when the successor feature in the update is nonzero, and equals zero otherwise. Hence

$$\|h_\tau\|_{\Sigma_\tau^{-1}} = \frac{\|w_{\tau+1} - \hat{z}_\tau\|_{\Sigma_\tau}}{\nu\beta_{\tau+1}} \le \frac{1}{\nu}.$$

For every episode k in this epoch, $\Sigma_{k-1} \succeq \Sigma_\tau$, so

$$\|h_\tau\|_{\Sigma^{-1}_{k-1}} \le \|h_\tau\|_{\Sigma_\tau^{-1}} \le \frac{1}{\nu}.$$

Cauchy–Schwarz then gives, for $u \neq 0$,

$$\frac{\langle u, h_\tau\rangle}{\|u\|_{\Sigma_{k-1}}} \ge -\|h_\tau\|_{\Sigma^{-1}_{k-1}} \ge -\frac{1}{\nu}.$$

□

A.3.3 Proof of Lemma 4

Proof of Lemma 4. Fix $k \in [n]$ and $u \neq 0$. By Eq. (12) in Lemma 2, at least a $1/10$ fraction of members satisfies $\langle u, \bar{q}^j_{k-1}\rangle / \|u\|_{\Sigma_{k-1}} \ge 1/40$. For each such member, Lemma 3 and $\nu = 80$ give

$$\frac{\langle u, q^j_{k-1}\rangle}{\|u\|_{\Sigma_{k-1}}} = \frac{\langle u, \bar{q}^j_{k-1}\rangle}{\|u\|_{\Sigma_{k-1}}} + \frac{\langle u, h_\tau\rangle}{\|u\|_{\Sigma_{k-1}}} \ge \frac{1}{40} - \frac{1}{80} = \frac{1}{80}.$$

The fraction is therefore preserved simultaneously for every k and u. □

A.3.4 PROOF OF LEMMA 5

Proof of Lemma 5. For each $j \in [m]$, define

$$I_k^j := \mathbb{1}\left\{\langle \psi(s_{k,0}, z_k^j), \widetilde{z}_k^j \rangle \geq \langle \psi(s_{k,0}, z_r), z_r \rangle\right\}.$$

Since J_k is uniform conditional on $\mathcal{A}_k$,

$$p_k = \frac{1}{m}\sum_{j=1}^{m} I_k^j.$$

For $j \in [m]$, let

$$\Delta_k^j := \langle \psi(s_{k,0}, z_r), z_r \rangle - \langle \psi(s_{k,0}, z_k^j), z_r \rangle.$$

For any ℓ such that $I_k^\ell = 1$, by definition,

$$\langle \psi(s_{k,0}, z_r), z_r \rangle \leq \left\langle \psi(s_{k,0}, z_k^\ell), \widetilde{z}_k^\ell \right\rangle.$$

Using the optimality of z_k^j for $\widetilde{z}_k^j$, we have

$$\begin{aligned}
\Delta_k^j &\leq \langle \psi(s_{k,0}, z_k^\ell), \widetilde{z}_k^\ell \rangle - \langle \psi(s_{k,0}, z_k^j), z_r \rangle \\
&= \langle \psi(s_{k,0}, z_k^\ell), \widetilde{z}_k^\ell - \widetilde{z}_k^j \rangle + \langle \psi(s_{k,0}, z_k^\ell), \widetilde{z}_k^j \rangle - \langle \psi(s_{k,0}, z_k^j), z_r \rangle \\
&\leq \langle \psi(s_{k,0}, z_k^\ell), \widetilde{z}_k^\ell - \widetilde{z}_k^j \rangle + \langle \psi(s_{k,0}, z_k^j), \widetilde{z}_k^j \rangle - \langle \psi(s_{k,0}, z_k^j), z_r \rangle \\
&= \langle \psi(s_{k,0}, z_k^\ell), \widetilde{z}_k^\ell - \widetilde{z}_k^j \rangle + \langle \psi(s_{k,0}, z_k^j), \widetilde{z}_k^j - z_r \rangle.
\end{aligned}$$

By Cauchy–Schwarz and the assumed concentration bound $\max_{j\in[m]} \|\widetilde{z}_k^j - z_r\|_{\Sigma_{k-1}} \leq B_k$, we have

$$\Delta_k^j \leq 2B_k \|\psi(s_{k,0}, z_k^\ell)\|_{\Sigma_{k-1}^{-1}} + B_k \|\psi(s_{k,0}, z_k^j)\|_{\Sigma_{k-1}^{-1}}. \tag{17}$$

Here we used

$$\|\widetilde{z}_k^\ell - \widetilde{z}_k^j\|_{\Sigma_{k-1}} \leq \|\widetilde{z}_k^\ell - z_r\|_{\Sigma_{k-1}} + \|z_r - \widetilde{z}_k^j\|_{\Sigma_{k-1}} \leq 2B_k.$$

Since Eq. (17) holds for every optimistic member ℓ, i.e., every ℓ such that $I_k^\ell = 1$, we may average its right-hand side over these members while leaving Δ_k^j unchanged. Since

$$\sum_{\ell=1}^{m} I_k^\ell = m p_k,$$

we obtain

$$\begin{aligned}
\Delta_k^j &\leq \frac{2B_k}{m p_k} \sum_{\ell=1}^{m} I_k^\ell \|\psi(s_{k,0}, z_k^\ell)\|_{\Sigma_{k-1}^{-1}} + B_k \|\psi(s_{k,0}, z_k^j)\|_{\Sigma_{k-1}^{-1}} \\
&\leq \frac{2B_k}{p_k} \frac{1}{m} \sum_{\ell=1}^{m} \|\psi(s_{k,0}, z_k^\ell)\|_{\Sigma_{k-1}^{-1}} + B_k \|\psi(s_{k,0}, z_k^j)\|_{\Sigma_{k-1}^{-1}}.
\end{aligned}$$

Taking the conditional expectation over $J_k \sim \mathrm{Unif}([m])$ given $\mathcal{A}_k$ yields

$$\begin{aligned}
\mathbb{E}\left[\Delta_k^{J_k} \,\middle|\, \mathcal{A}_k\right] &\leq \left(\frac{2B_k}{p_k} + B_k\right) \mathbb{E}\left[\|\psi(s_{k,0}, z_k)\|_{\Sigma_{k-1}^{-1}} \,\middle|\, \mathcal{A}_k\right] \\
&\leq \frac{3B_k}{p_k} \mathbb{E}\left[\|\psi(s_{k,0}, z_k)\|_{\Sigma_{k-1}^{-1}} \,\middle|\, \mathcal{A}_k\right],
\end{aligned}$$

where the last inequality uses $p_k \leq 1$.

Finally, by the linear reward model and the definition of successor features,

$$\mathbb{E}\left[G_{k,z_r} - G_{k,z_k} \,|\, \mathcal{A}_k\right] = \mathbb{E}\left[\Delta_k^{J_k} \,\middle|\, \mathcal{A}_k\right].$$

This proves the claim. □

A.4 Auxiliary Lemmas

Lemma 8 (Lemma 10 in Abbasi-Yadkori et al. (2011)). *Suppose $X_1, \dots, X_t \in \mathbb{R}^d$ and for any $1 \le s \le t$, $\|X_s\|_2 \le L$, Let $\mathbf{V}_t = \lambda \mathbf{I}_d + \sum_{s=1}^{t} X_s X_s^\top$ for some $\lambda > 0$. Then,*

$$\det(\mathbf{V}_t) \le (\lambda + tL^2/d)^d \,.$$

Lemma 9 (Lemma 11 in Abbasi-Yadkori et al. (2011)). *Let $\{X_t\}_{t=1}^{\infty}$ be a sequence in $\mathbb{R}^d$, V a $d \times d$ positive matrix and define $\bar{V}_t = V + \sum_{s=1}^{t} X_s X_s^\top$. If $\|X_t\|_2 \le L$ for all t, then*

$$\sum_{t=1}^{n} \min\left\{1, \|X_t\|^2_{\bar{V}_{t-1}^{-1}}\right\} \le 2 \log\left(\frac{\det \bar{V}_n}{\det V}\right) .$$

B Pretraining Protocol for BFMs

In this appendix, we first describe the forward-backward framework and its pretraining objectives. We then detail the pretraining datasets, network architectures, and hyperparameters.

B.1 Forward-Backward Representations

We pretrain the BFM using forward-backward (FB) representations (Touati & Ollivier, 2021). FB builds on successor representations (Dayan, 1993) and their extension to successor measures (Blier et al., 2021). Following Touati et al. (2023) we define the successor measure as the expected discounted number of visits to a measurable set $X \subseteq \mathcal{S}$ after the initial action:

$$M^\pi(s_0, a_0, X) := \sum_{t=0}^{\infty} \gamma^t \Pr\left(s_{t+1} \in X \,|\, s_0,\ a_0,\ \pi\right). \tag{18}$$

Let $\mathcal{D}_{\text{train}}$ denote the unlabeled pretraining dataset and ρ its marginal state distribution. FB learns a forward representation $F : \mathcal{S} \times \mathcal{A} \times \mathbb{R}^d \to \mathbb{R}^d$ and a backward representation $B : \mathcal{S} \to \mathbb{R}^d$ with the desired relations

$$\begin{aligned} M^{\pi_z}(s, a, \mathrm{d}s^+) &\approx F(s, a, z)^\top B(s^+)\, \rho(\mathrm{d}s^+), \\ \operatorname{supp} \pi_z(\cdot \mid s) &\subseteq \arg\max_{a \in \mathcal{A}} F(s, a, z)^\top z. \end{aligned} \tag{19}$$

The inner product $F(s, a, z)^\top B(s^+)$ approximates the density of the successor measure with respect to ρ. The second relation requires the policy to assign probability only to greedy actions. The task embedding z indexes the policies represented by the model.

The representations F and B are parameterized by neural networks and jointly trained using the Bellman equation for successor measures (Touati & Ollivier, 2021; Touati et al., 2023):

$$M^{\pi_z}(s, a, X) = P(X \mid s, a) + \gamma \mathbb{E}_{\substack{s' \sim P(\cdot|s,a) \\ a' \sim \pi_z(\cdot|s')}} \left[M^{\pi_z}(s', a', X)\right]. \tag{20}$$

For each sampled task embedding z we draw a transition $(s, a, s') \sim \mathcal{D}_{\text{train}}$ and an independent state $s^+ \sim \rho$. We also sample $a' \sim \pi_z(\cdot \mid s')$ for the bootstrap target. The FB loss is

$$\begin{aligned} \mathcal{L}_{\text{FB}}(F, B; z) = \mathbb{E}_{\substack{(s,a,s') \sim \mathcal{D}_{\text{train}} \\ a' \sim \pi_z(\cdot|s') \\ s^+ \sim \rho}} \left[\left(F(s, a, z)^\top B(s^+) - \gamma \bar{F}(s', a', z)^\top \bar{B}(s^+)\right)^2\right] \\ - 2\mathbb{E}_{(s,a,s') \sim \mathcal{D}_{\text{train}}} \left[F(s, a, z)^\top B(s')\right], \end{aligned} \tag{21}$$

where $\bar{F}$ and $\bar{B}$ are target networks. The task embedding and the bootstrap target are treated as constants during representation updates.

We regularize the backward representation using a normalization loss that penalizes deviations of its second moment matrix from the identity (Touati et al., 2023). This loss can be written as

$$\begin{aligned} \mathcal{L}_{\text{norm}}(B) &= \left\|\mathbb{E}_{s \sim \rho}\left[B(s)B(s)^\top\right] - I_d\right\|^2_{\text{F}} \\ &= \mathbb{E}_{s, s^+ \overset{\text{iid}}{\sim} \rho}\left[\left(B(s)^\top B(s^+)\right)^2 - \|B(s)\|_2^2 - \|B(s^+)\|_2^2\right] + d. \end{aligned} \tag{22}$$

This expression can be estimated using independent state pairs and the constant d can be omitted during optimization.

In continuous action spaces we parameterize $\pi_z(\cdot \mid s) = \pi(\cdot \mid s, z)$ with a policy network. The network is trained to approximate the greedy condition in Equation (19) by minimizing

$$\mathcal{L}_\pi(z) := -\mathbb{E}_{s\sim\rho}\mathbb{E}_{a\sim\pi_z(\cdot|s)}\left[F(s,a,z)^\top z\right]. \tag{23}$$

During policy updates the representation parameters are held fixed. Gradients propagate through reparameterized action samples to the policy network. The same policy supplies the sampled actions a' in the bootstrap targets. Training averages the task-dependent objectives over sampled embeddings and includes the weighted normalization loss. For the ensemble implementation and the Q-value penalty setting, we follow Rupf et al. (2026).

After pretraining we freeze the learned networks. The FB factorization gives

$$\int_{\mathcal{S}} B(s^+)\, M^{\pi_z}(s,a,\mathrm{d}s^+) \approx \mathbb{E}_{s^+\sim\rho}\left[B(s^+)B(s^+)^\top\right] F(s,a,z). \tag{24}$$

When the second moment matrix is invertible this relation identifies the state features for task inference as (Touati & Ollivier, 2021; Rupf et al., 2026)

$$\phi(s) := \left(\mathbb{E}_{s^+\sim\rho}\left[B(s^+)B(s^+)^\top\right]\right)^{-1} B(s). \tag{25}$$

Following Rupf et al. (2026) we estimate the expectation over ρ once using 50k states sampled from $\mathcal{D}_{\text{train}}$ after pretraining. We use this estimate to construct ϕ and keep the resulting feature map fixed during downstream task inference. For offline task inference, let $\mathcal{D}_{\text{inf}}$ denote a dataset of states labeled with a downstream reward function r. The task embedding can be estimated by unregularized least squares. Assuming that the required matrices are invertible, we obtain

$$\begin{aligned}\widehat{z}_r &= \left(\mathbb{E}_{\mathcal{D}_{\text{inf}}}\left[\phi(s)\phi(s)^\top\right]\right)^{-1}\mathbb{E}_{\mathcal{D}_{\text{inf}}}\left[\phi(s)r(s)\right] \\ &= \mathbb{E}_{s^+\sim\rho}\left[B(s^+)B(s^+)^\top\right]\left(\mathbb{E}_{\mathcal{D}_{\text{inf}}}\left[B(s)B(s)^\top\right]\right)^{-1}\mathbb{E}_{\mathcal{D}_{\text{inf}}}\left[B(s)r(s)\right].\end{aligned} \tag{26}$$

Here, $\mathbb{E}_{\mathcal{D}_{\text{inf}}}$ denotes empirical averaging over the labeled inference dataset. The expectation over ρ is replaced by its precomputed estimate in implementation. For policy execution, we normalize the inferred embedding to the pretraining sphere and use it to condition the pretrained policy without additional policy optimization.

B.2 Datasets and Implementation Details

We follow the experimental setup of OpTI-BFM (Rupf et al., 2026). Specifically, we pretrain the FB model on offline datasets collected using random network distillation (RND) (Burda et al., 2019; Yarats et al., 2022). Dataset sizes are reported in Appendix C.1. This pretraining setup is also consistent with previous zero-shot RL studies (Touati et al., 2023; Agarwal et al., 2025).

We follow the FB architecture used in the DMC implementation released by Tirinzoni et al. (2025). We parameterize the backward representation B with a 3-layer MLP and the forward representation F with an ensemble of two networks. Each ensemble member separately encodes (s, a) and (s, z) using 2-layer MLPs, then passes their concatenated outputs through another 2-layer MLP to obtain the forward representation. The policy network follows a similar structure, using separate 2-layer MLPs to encode s and (s, z). Their outputs are concatenated and mapped to the action mean $\mu(s, z)$ by an additional 2-layer MLP. We use a truncated Gaussian policy with a fixed standard deviation parameter $\sigma = 0.2$ and noise truncated at $\pm 1.5\sigma$. During pretraining, we constrain task embeddings to a sphere in $\mathbb{R}^d$, since positive rescaling of the reward function does not change the optimal policy. We sample embeddings from an equal mixture of normalized backward representations of dataset states and random directions on this sphere. During task inference, we preserve the magnitude of the estimated reward coefficients and normalize the corresponding embeddings to the pretraining sphere when using them to condition the policy. The hyperparameters used for FB pretraining are summarized in Table 2.

Table 2: FB pretraining hyperparameters.

Hyperparameter	Value
Discount factor γ	0.98
Batch size	1024
Number of training steps	2×10^6
Optimizer	Adam (Kingma & Ba, 2015)
Learning rate	1e$-$4
Target network update factor	0.01
Weight of $\mathcal{L}_{\text{norm}}$	1.0
Q-value penalty	0.5
Fixed actor standard deviation σ	0.2
Actor noise clipping threshold	0.3
Task embedding sampling	50% $\mathcal{D}_{\text{train}}$ and 50% Random
Representation dimension d	50
B network output normalization	L2 normalization
B network hidden dimension	256
F network hidden dimension	1024
π network hidden dimension	1024
First-layer normalization and activation	LayerNorm + Tanh
Other hidden-layer activations	ReLU

C Additional Experimental Details

In this appendix, we first briefly introduce how cumulative rewards are calculated in the ExORL benchmark. We then describe our implementation and experimental details.

C.1 ExORL Benchmarks



Figure 5: The five ExORL domains used in our experiments: `Walker`, `Cheetah`, `Quadruped`, `Jaco Arm`, and `PointMass Maze`.

We evaluate our method on 20 tasks across five state-based continuous-control domains from the ExORL benchmark (Yarats et al., 2022). These domains cover locomotion, robotic reaching, and maze navigation, as illustrated in Figure 5. Each domain contains four tasks that share the same environment dynamics but differ in their reward functions:

- `Walker`: *stand*, *walk*, *run*, and *flip*.
- `Cheetah`: *walk*, *walk backward*, *run*, and *run backward*.
- `Quadruped`: *stand*, *walk*, *run*, and *jump*.
- `Jaco Arm`: *reach top left*, *reach top right*, *reach bottom left*, and *reach bottom right*.
- `PointMass Maze`: *reach top left*, *reach top right*, *reach bottom left*, and *reach bottom right*.

Figure 6 shows example trajectories from each domain. The action space is normalized to $[-1, 1]$ in each dimension, and the action repeat is set to 1.

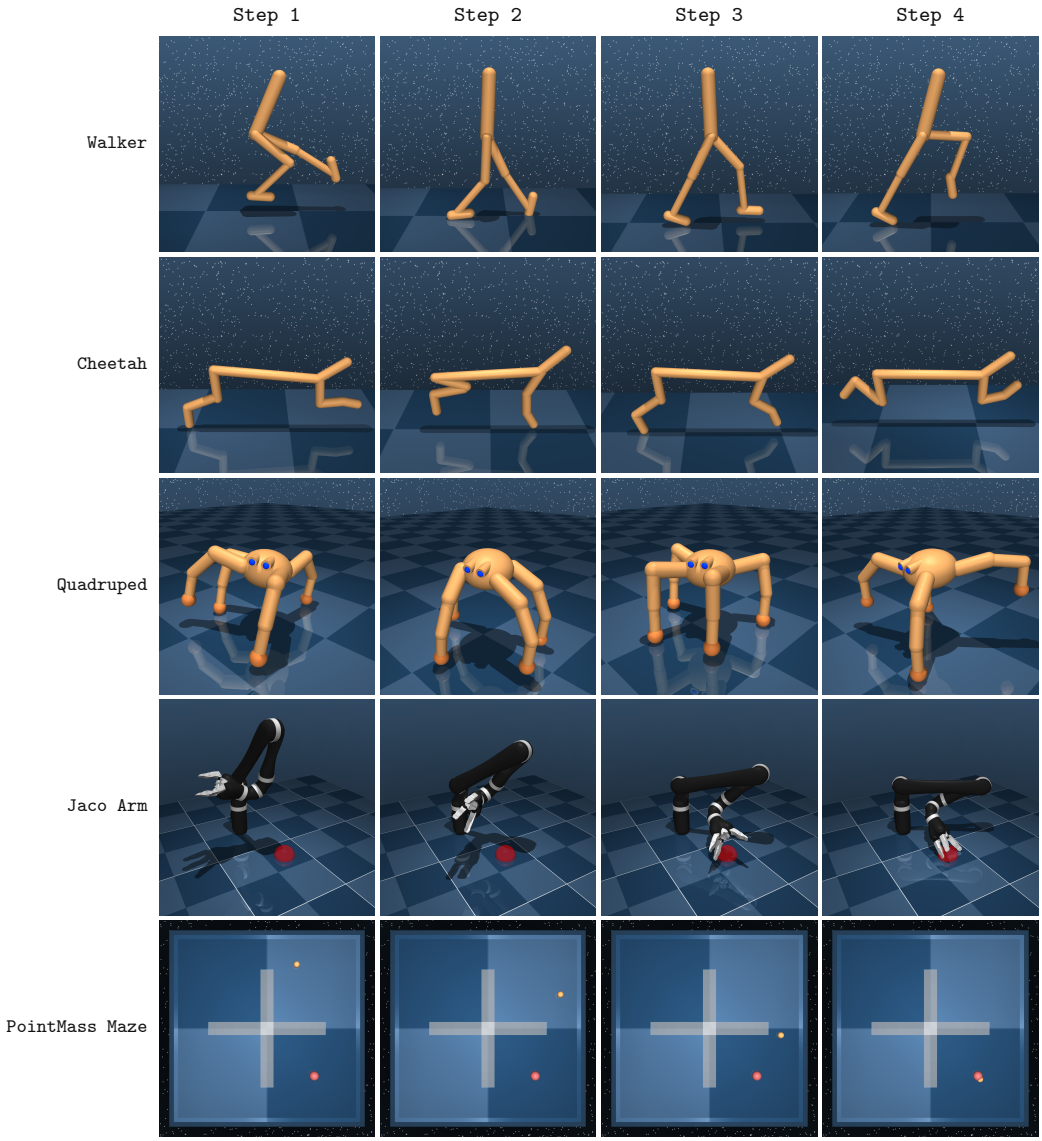


Figure 6: Example trajectories in the five ExORL domains. Each row corresponds to a domain, and the columns show four selected frames in temporal order.

We use the publicly released ExORL datasets collected by a RND agent (Burda et al., 2019; Yarats et al., 2022) using intrinsic exploration rewards without access to downstream task rewards. Each domain's dataset contains 10,000,000 transitions, except for `Jaco Arm`, which contains 6,359,250 transitions. Within each domain, all methods and downstream tasks share the same FB model checkpoint, obtained after 2×10^6 pretraining updates on the corresponding dataset. The model parameters remain fixed during evaluation, and online adaptation is performed through task inference.

In our evaluation, each episode has a fixed horizon of $H = 1{,}000$ environment steps. For all tasks, the per-step reward lies in $[0, 1]$. We report the undiscounted episodic return without additional score normalization: $R_k = \sum_{t=0}^{H-1} r_{k,t}$, where $r_{k,t}$ denotes the reward at step t of episode k. Thus, $R_k \in [0, 1{,}000]$, with 1,000 representing a theoretical upper bound.

For each method and task, we evaluate 10 runs per seed using five evaluation seeds. At each episode index, we compute domain-level returns by averaging equally over the 10 runs and 4 tasks within each seed.

C.2 Experimental Results

r_{int} denotes the latent reselection interval in environment steps, m the ensemble size, and N_{cand} the number of candidates for optimistic selection. The prior precision is $P_0 = \lambda I$, and σ denotes the reward-model observation-noise standard deviation. The refresh threshold, perturbation inflation factor, and optimistic recentering strength are α_{ER}, ν, and ρ, respectively.

Stationary rewards. We evaluate online task inference on 20 tasks across five domains: `Walker`, `Cheetah`, `Quadruped`, `Jaco Arm`, and `PointMass Maze`. Within each domain, all methods use the same pretrained BFM checkpoint, and the network parameters remain fixed during adaptation. Reward feedback is available at every environment step. Figure 7 reports episodic returns for each task over 10 adaptation episodes.

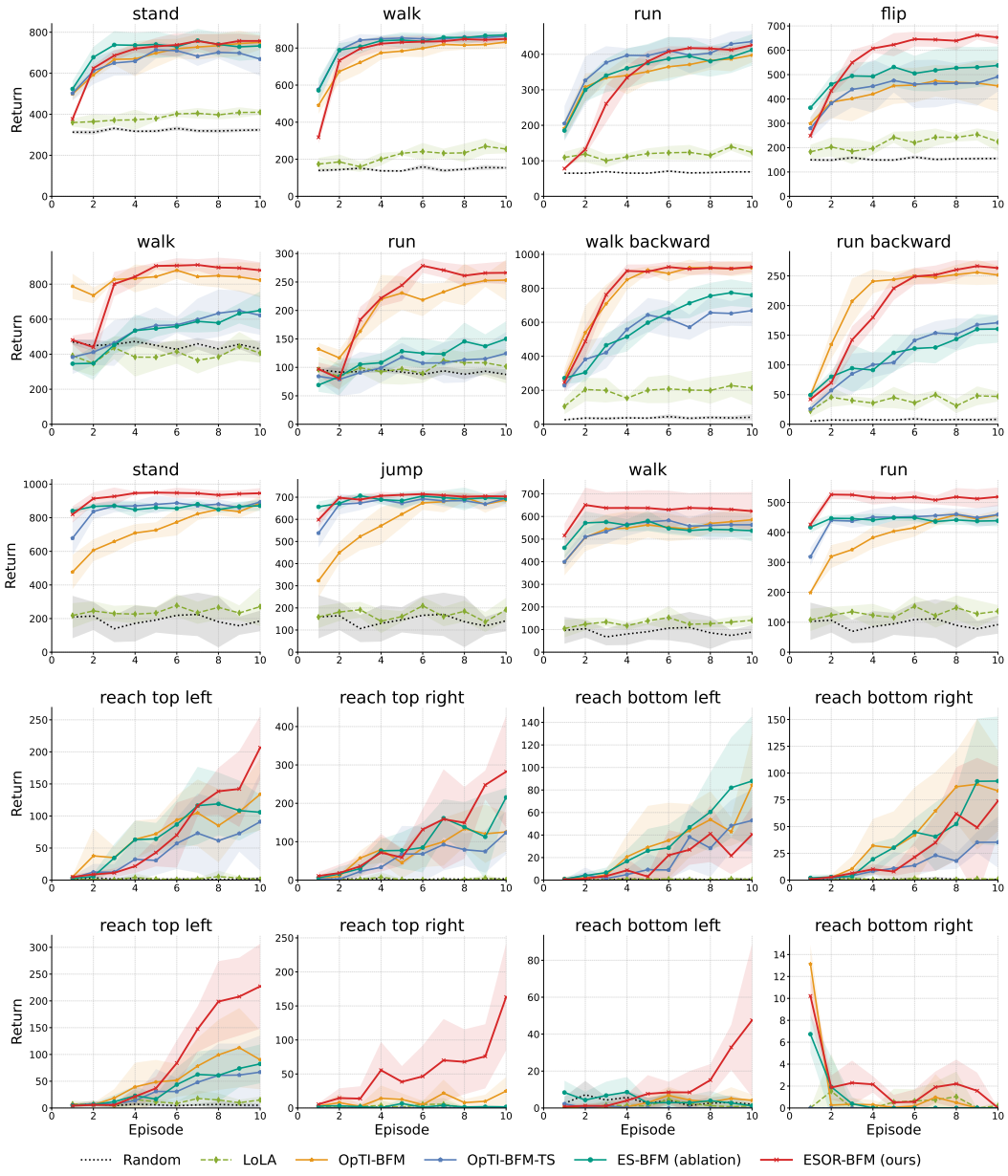


Figure 7: Task-wise learning curves under stationary rewards. Rows correspond to `Walker`, `Cheetah`, `Quadruped`, `Jaco Arm`, and `PointMass Maze`, from top to bottom. Each panel shows episodic returns for one downstream task over 10 episodes of online adaptation.

Table 3: Hyperparameters for each environment in stationary rewards.

	ES-BFM	ESOR-BFM		
Environment	α_{ER}	ν	α_{ER}	ρ
`Walker`	0.1	0.5	0.1	0.5
`Cheetah`	0.1	0.1	0.1	0.5
`Quadruped`	0.1	0.1	0.1	0.5
`Jaco Arm`	0.5	0.5	0.5	0.1
`PointMass Maze`	0.1	0.1	0.1	1.0

Hyperparameters for ES-BFM and ESOR-BFM are listed in Table 3. For baselines, we use the hyperparameter settings from the official OpTI-BFM implementation.

Selective reward labeling. We examine the tradeoff between task performance and reward-label acquisition in `Walker`, `Cheetah`, and `Quadruped`. Unlike the full-feedback setting, the agent selectively requests reward labels using the D-gap defined in Equation 7. The D-gap measures the increase in the log determinant of the regularized regression Gram matrix that would result from incorporating the observed feature vector. A reward label is acquired only when this quantity meets a threshold κ. We use a separate hyperparameter configuration for selective reward labeling, as reported in Table 4. For ESOR-BFM and ES-BFM, we set $\sigma = 0.1$ in `Walker` and $\sigma = 1.0$ in `Cheetah` and `Quadruped`. For baselines, we use the hyperparameter settings from the official OpTI-BFM implementation.

Table 4: Hyperparameters for each environment in selective reward labeling.

	ES-BFM	ESOR-BFM		
Environment	α_{ER}	ν	α_{ER}	ρ
`Walker`	1.0	0.5	1.0	0.05
`Cheetah`	0.1	0.1	0.1	0.01
`Quadruped`	0.5	0.5	0.5	0.01

Perturbed SF estimates. We evaluate robustness to inaccurate SF estimates in `Walker`, `Cheetah`, and `Quadruped`. During online task inference, we replace the predicted SFs with

$$\widetilde{\psi}(s, z) = \psi(s, z) + \alpha g_\theta(z),$$

where g_θ is a fixed random network whose outputs have unit Euclidean norm. The coefficient α controls the magnitude of the injected perturbation, with $\alpha = 0$ recovering the unperturbed predictions. The fixed network introduces a consistent distortion of the SF estimates across interactions.

We perturb only the SF predictions used during online task inference. The pretrained policy, state feature map, and reward regression procedure remain unchanged. We compare ESOR-BFM, which uses SF estimates for optimistic search at refresh events, with OpTI-BFM, which uses them at every interaction step. This comparison examines how these different uses of SF estimates affect sensitivity to prediction errors. We use the same hyperparameters as in the stationary experiments. Figure 8 reports episodic returns for each task over 10 episodes, with SF perturbation strengths $\alpha \in \{0, 100, 300, 1000\}$. We exclude OpTI-BFM-TS and ES-BFM from this comparison because neither method uses predicted SFs for online latent selection, and ES-BFM's refresh criterion depends only on posterior precision. With the feature map, reward regression, and pretrained policy held fixed, perturbing the SF predictions alone therefore leaves the behavior of both methods unchanged.

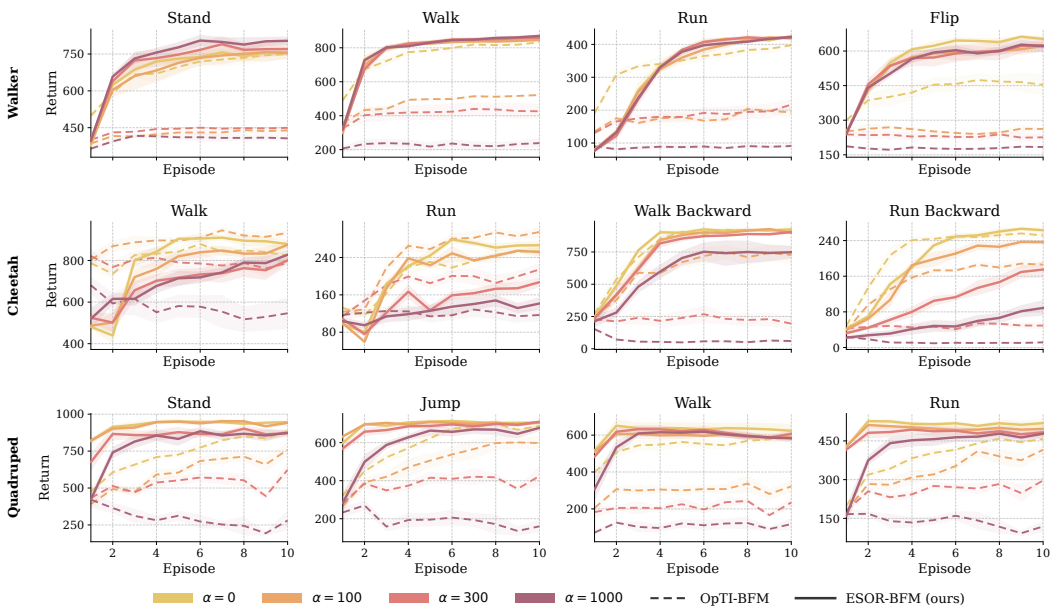


Figure 8: Task-wise learning curves under perturbed SF estimates. Rows correspond to `Walker`, `Cheetah`, and `Quadruped`, from top to bottom. Each panel shows episodic returns for one downstream task over 10 episodes. Solid and dashed lines denote ESOR-BFM and OpTI-BFM, respectively. Colors indicate the perturbation strength α.

Nonlinear rewards. We vary the residual component of the reward. Given an original reward function $r(s)$, we fit a linear approximation $\phi(s)^\top z_r$ by least squares and define the residual as $e(s) = r(s) - \phi(s)^\top z_r$. Whereas Rupf et al. (2026) approximate the residual using a separately trained network, we compute it directly from the environment reward and its fitted linear component. We then evaluate the modified rewards

$$r_\beta(s) = \phi(s)^\top z_r + \beta e(s) \tag{27}$$

for $\beta \in \{0.5, 1.0, 2.0\}$. Setting $\beta = 1.0$ recovers the original reward, while $\beta = 0.5$ attenuates the residual component and $\beta = 2.0$ amplifies it.

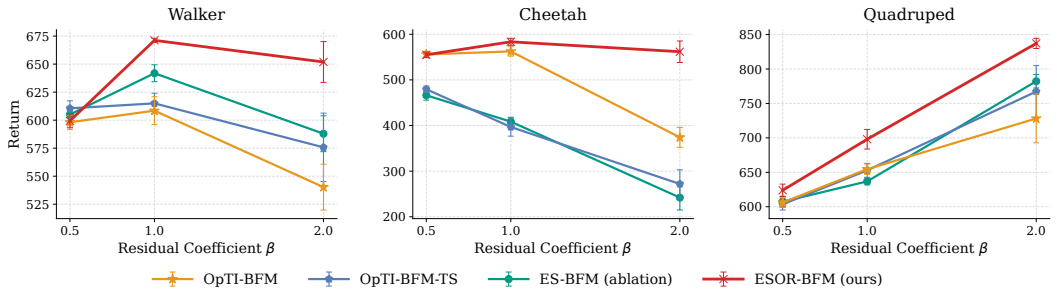


Figure 9: Mean final returns under the modified rewards r_β, averaged across target tasks for each residual coefficient β.

Figure 9 shows that ESOR-BFM achieves the highest mean final return in all three environments for both the original reward ($\beta = 1.0$) and the amplified residual setting ($\beta = 2.0$). In `Quadruped`, it also has the highest mean final return at $\beta = 0.5$. At $\beta = 2.0$, its advantage over the strongest baseline is particularly pronounced in `Cheetah`.

C.3 Online Task Inference Hyperparameters

Table 5 summarizes the hyperparameters used for stationary online task inference. The latent reselection interval r_{int} specifies the number of environment steps between successive task embedding selections. The ensemble size m applies to ES-BFM and ESOR-BFM, while the number of UCB candidates N_{cand} applies to OpTI-BFM and ESOR-BFM.

Posterior-based methods use unit prior precision $\lambda = 1$. In particular, OpTI-BFM-TS initializes its posterior directly as $\mathcal{N}(0, I)$. For OpTI-BFM and ESOR-BFM, β denotes a constant UCB coefficient, and their covariance-ellipsoid candidate samplers use a Mahalanobis radius of 2β. In ES-BFM, $\beta = 1$ instead serves as an ensemble perturbation multiplier. Environment- and setting-specific values of the refresh threshold α_{ER}, perturbation inflation factor ν, and recentering strength ρ are reported separately.

Table 5: Hyperparameters for stationary online task inference.

Hyperparameter	**Symbol**	**Value**
Number of adaptation episodes	n	10
Episode horizon	H	1,000
Task embedding dimension	d	50
Latent reselection interval	r_{int}	1
Prior precision	λ	1
Ensemble size	m	128
Number of UCB candidates	N_{cand}	128
Pretraining sphere radius	R_z	$\sqrt{d} = \sqrt{50}$