GIRO: ROBUST REINFORCEMENT LEARNING WITH GENERATIVE MODEL INDUCED UNCERTAINTY SET

Anonymous authors
Paper under double-blind review

ABSTRACT

Offline model-based reinforcement learning uses learned transition dynamics to generate training experience, but policies optimized against an estimated model may exploit its errors and fail under distribution shift. We introduce GIRO, a distributionally robust RL method that defines an uncertainty set of generative transition models through a training-loss budget. Because shared model parameters couple transitions across state–action pairs, the resulting set is generally non-rectangular, making robust policy evaluation challenging. GIRO alternates adversarial model updates with robust policy improvement. For robust policy evaluation, it uses a primal-only update that decreases policy value when the loss constraint is satisfied and reduces constraint violation otherwise. Under stated assumptions, we bound robust evaluation suboptimality and constraint violation and establish a stationarity guarantee for policy improvement. Across nine D4RL task–dataset combinations and eight test perturbations, GIRO attains the highest aggregate worst-case return and lowest aggregate across-perturbation variance among the compared methods.

1 INTRODUCTION

Reinforcement learning (RL) (Sutton et al., 1998) has achieved strong empirical performance across a wide range of sequential decision-making problems, but its dependence on environment interaction can make training expensive, slow, and unsafe. A common strategy for improving data efficiency is to replace direct interaction with simulated experience, which can reduce reliance on costly or risky real-world interactions, support planning through simulated trajectories, and represent stochastic or multimodal transitions. Rather than relying on a hand-designed simulator, recent methods learn generative environment models (Ha & Schmidhuber, 2018) that predict future states conditioned on the agent's actions and can be used to generate synthetic trajectories for policy optimization.

Generative dynamics models, e.g., encoder-decoder architectures (Hafner et al., 2020; 2023; Micheli et al., 2022; Wang et al., 2025), and diffusion-based world models (Jiang et al., 2025; Ding et al., 2024; Alonso et al., 2024; Li et al., 2025), have shown promise in modeling complex, high-dimensional transition distributions. These advances suggest that learned simulators can provide increasingly realistic training environments for downstream control. However, policies optimized within a learned simulator may exploit model errors and consequently transfer poorly to the real environment.

Distributionally robust RL (Ou & Bi, 2025) offers a way to address this problem by optimizing a policy against the worst-case transition dynamics in an uncertainty set, rather than against a single learned simulator. Accounting for plausible model errors can discourage policies from relying on inaccurate simulated transitions. Applying this approach to generative dynamics models, however, raises two challenges. The uncertainty set must reflect what the offline data tell us about the model while allowing meaningful variation in its predicted transitions. Moreover, shared generative-model parameters jointly determine transitions across state–action pairs, generally yielding a non-rectangular uncertainty set for which standard robust dynamic programming methods do not directly apply.

Many classical robust MDP formulations construct an uncertainty set by placing a divergence or distance ball around the estimated transition distribution at each state–action pair (Shi et al., 2023; Panaganti & Kalathil, 2022; Clement & Kroer, 2021; Wang et al., 2024; 2023b; Ho et al., 2026;

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.

Ma et al., 2022). This approach is natural in tabular settings, where conditional distributions can be estimated from transition counts, and rectangularity enables Bellman-style optimization. In continuous or high-dimensional spaces, however, estimating and calibrating a separate uncertainty set for every state–action pair is difficult. Moreover, rectangular sets allow these distributions to vary independently, potentially admitting combinations of transitions that cannot be generated by any single model with shared parameters.

These limitations motivate us to define an uncertainty set through the parameters of a generative transition model. Building on the loss-budget construction of Wen & Yang (2026), we include models whose training loss on the offline data falls below a prescribed threshold, constraining adversarial dynamics by their aggregate fit to observed transitions. This does not require a separate uncertainty radius at every state–action pair, and the construction retains the dependence among transitions induced by shared parameters. Moreover, within this uncertainty set, generative models can produce next states beyond those observed in the finite dataset, allowing the worst-case search to consider transitions that are absent from the observed samples. Parameterizing the set also turns the search over transition kernels into an optimization over model parameters, while shared parameters preserve dependencies across state–action pairs. This coupling makes the uncertainty set generally non-rectangular, creating the main challenge for robust policy evaluation and improvement.

To solve the resulting robust RL problem, we introduce Generative-model-Induced Robust Policy Optimization (GIRO), which alternates between robust policy evaluation and robust policy improvement. Robust policy evaluation is challenging because the loss-defined uncertainty set is generally non-rectangular and does not admit a simple projection in transition-kernel space (Wang et al., 2023a; Li et al., 2026). We address this challenge by deriving model gradients using diffusion trajectories or VAE state–latent pairs, and by adapting a primal-only update for the adversarial model. The update decreases policy value when the model satisfies the loss budget and switches to reducing the loss when the budget is violated, without introducing dual variables or projecting transition kernels (Xu et al., 2021; Lan & Zhou, 2020). We establish bounds on the evaluation gap and constraint violation, followed by an approximate stationarity guarantee for policy improvement.

Finally, we evaluate GIRO on a range of D4RL benchmark tasks (Fu et al., 2020), using both conditional VAE and diffusion models trained on the corresponding offline datasets. We evaluate the trained policies across eight disturbance settings, including additive state and action noise and changes to body mass and friction parameters. Our results show that GIRO achieves stronger worst-case performance under transition disturbances and maintains more consistent performance across different disturbance types.

2 Related Work

Distributionally Robust RL. In distributionally robust RL, one seeks to optimize the worst-case return over an uncertainty set of transition kernels (Ou & Bi, 2025). Uncertainty sets are commonly distinguished by their coupling structure, e.g., s-rectangular, (s, a)-rectangular, d-rectangular and non-rectangular, and local geometry, e.g., total variation, chi-square, Kullback–Leibler (KL), or Wasserstein distance (Shi et al., 2023; Panaganti & Kalathil, 2022; Clement & Kroer, 2021; Wang et al., 2024; 2023b; Ho et al., 2026; Ma et al., 2022).

Non-rectangular uncertainty sets couple multiple states or trajectories, so the Bellman equation generally fails, and convergence becomes more challenging. Recent methods for non-rectangular settings (Wang & Si, 2026; Wang et al., 2023a; Li et al., 2026) are not directly applicable to our problem because they assume that the transition model P_ξ is affine in ξ and require projecting the model parameter ξ onto an explicit uncertainty set. Our uncertainty set is defined by the generative model parameters and its training loss, so it does not allow such a projection. Parametric RMDPs are closely related to our work, as our uncertainty set is also parameterized. However, existing formulations typically assume them to be rectangular, e.g., (Black et al., 2022), whereas ours is non-rectangular and couples transitions across state–action pairs.

RL using Generative World Models. Model-based RL is often more sample-efficient than direct policy learning, as it first learns a generative model for the transition dynamics and then optimizes the policy using that Agarwal et al. (2020); Li et al. (2024). A widely used class of world models is latent world models, which typically follow an encoder–decoder architecture coupled with a la-

tent dynamics model (Hafner et al., 2020; 2023; Micheli et al., 2022; Wang et al., 2025; Li et al., 2025). More recently, diffusion-based world models have also been explored for generating imagined states (Jiang et al., 2025; Ding et al., 2024; Alonso et al., 2024). Policies are then trained on synthetic data produced by these generative models. However, more expressive generative models do not by themselves ensure that policies trained on synthetic trajectories are robust to errors in the learned dynamics; during optimization, a policy may exploit those errors and perform poorly in the real environment. In contrast, we explicitly account for potential errors in the learned dynamics by optimizing policies against worst-case transition models within a training-loss-constrained uncertainty set.

3 PRELIMINARIES

MDPs and Offline RL. We consider a Markov decision process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathsf{P}, R, \gamma)$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ denotes the action space, $\mathsf{P} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition kernel, and $\mathsf{P}(\cdot|s, a)$ denotes the probability distribution of the next state after taking action a in state s. The function $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ denotes the reward function with $R(s, a) \in [0, 1]$, $\rho \in \Delta(\mathcal{S})$ is the initial state distribution, and $\gamma \in [0, 1)$ is the discount factor. A *stationary* policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ maps each state to a distribution over actions. In this work, we denote by $d^{\mathsf{P}}_{\pi,s}$ the occupancy measure of policy π starting from initial state s under transition kernel P. For an initial state distribution $\rho \in \Delta(\mathcal{S})$, we define $d^{\mathsf{P}}_{\pi,\rho} = \mathbb{E}_{s\sim\rho}\big[d^{\mathsf{P}}_{\pi,s}\big]$. We focus on the infinite-horizon discounted setting, in which both the transition kernel and the reward function are stationary. The objective in standard RL is to find a policy π that maximizes the expected discounted reward: $\max_\pi V^{\mathsf{P}}_\pi(s) := \mathbb{E}_{\pi,\mathsf{P}}[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \,|\, s_0 = s]$. The value function under the initial state distribution ρ is given by $V^{\mathsf{P}}_\pi(\rho) = \mathbb{E}_{s\sim\rho}\big[V^{\mathsf{P}}_\pi(s)\big]$.

In offline RL, the agent has access only to a fixed dataset of transitions collected from the underlying MDP, $\mathcal{D} = \{(s_i, a_i, s'_i, R_i)\}_{i=1}^{|\mathcal{D}|}$. The objective of offline RL is to learn a policy that maximizes the expected return in the true environment using only this offline dataset.

Distributionally Robust MDPs. In distributionally robust MDPs, the goal is to find a policy that is robust to uncertainty in the transition dynamics. When the transition kernel is uncertain, we represent the candidate kernels by an uncertainty set $\mathcal{P}$. The goal is to find a policy π that maximizes the return under the worst-case transition kernel within $\mathcal{P}$:

$$f = \max_\pi \min_{\mathsf{P}\in\mathcal{P}} V^{\mathsf{P}}_\pi(\rho). \tag{1}$$

The inner minimization problem $\min_{\mathsf{P}\in\mathcal{P}} V^{\mathsf{P}}_\pi(\rho)$ for a given π is usually referred to as robust policy evaluation. The structure of the uncertainty set $\mathcal{P}$ largely determines the difficulty of the robust RL problem. Uncertainty sets can be divided into two categories based on their coupling structure: rectangular and non-rectangular. Two commonly used rectangular uncertainty sets are (s, a)-rectangular and s-rectangular uncertainty sets, defined as follows:

$$(s,a)\text{-rectangular:} \quad \mathcal{P} = \prod_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathcal{P}_{s,a}, \qquad s\text{-rectangular:} \quad \mathcal{P} = \prod_{s\in\mathcal{S}} \mathcal{P}_s.$$

Here, $\mathcal{P}_{s,a} \subseteq \Delta(\mathcal{S})$ is the set of admissible next-state distributions for a given state-action pair, whereas $\mathcal{P}_s \subseteq \Delta(\mathcal{S})^{\mathcal{A}}$ is a set of action-indexed collections $(\mathsf{P}(\cdot \mid s, a))_{a\in\mathcal{A}}$. Thus, (s, a)-rectangularity allows transition distributions to vary independently across state-action pairs, while s-rectangularity allows coupling across actions at the same state but not across states. A non-rectangular uncertainty set $\mathcal{P} \subseteq \Delta(\mathcal{S})^{\mathcal{S}\times\mathcal{A}}$ is neither s-rectangular nor (s, a)-rectangular.

In this work, let ξ be the parameter of the generative model for the transition dynamics, and we will construct a parameter set $\Xi \subseteq \mathbb{R}^d$ such that the uncertainty set can be represented as $\mathcal{P} = \{\mathsf{P}_\xi : \xi \in \Xi\}$. For non-rectangular uncertainty sets, transition distributions cannot in general be minimized independently across states, and the standard robust Bellman decomposition need not apply. In this work, we restrict attention to stationary Markov policies, a commonly used policy class in practical RL.

4 GENERATIVE MODEL INDUCED UNCERTAINTY SETS CONSTRUCTION

To account for potential errors in the transition dynamics predicted by a generative model used as a simulator, we construct a model-induced uncertainty set for distributionally robust offline RL. Given an offline dataset, we train a generative transition model and a separate reward model. We then bound the transition model's empirical training loss to define an uncertainty set over its parameters and the corresponding transition kernels. We instantiate this construction with conditional diffusion models and conditional variational autoencoders (VAEs).

Conditional Diffusion Model. We approximate the transition kernel $\mathsf{P}(\cdot \mid s, a)$ with a conditional diffusion model that generates the next state s' from (s, a). For the analysis, we adopt the SDE formulation of Song et al. (2020) and let $s' \sim \mathsf{P}(\cdot \mid s, a)$ denote a clean next-state sample from the true dynamics.

Forward process. The forward diffusion process evolves a clean next-state sample s' into a sequence of progressively noisier states. Let x_k denote the state at diffusion time $k \in [0, K]$, with $x_0 = s'$, where k indexes the continuous diffusion process. The forward dynamics are described by the stochastic differential equation

$$dx_k = b(x_k, k)\, dk + r(k)\, dw_k, \tag{2}$$

where $b(\cdot, k)$ is the drift coefficient, $r(k)$ is the diffusion coefficient, and w_k is a standard Brownian motion. As k approaches K, the distribution of x_k approaches a simple prior distribution, typically a standard Gaussian.

Let $P_k(x_k \mid s, a)$ denote the marginal of x_k, $P_{k|0}(x_k \mid s', s, a)$ its conditional given s', and $P(x_{0:K} \mid s, a)$ the induced trajectory distribution. Throughout, P denotes a transition kernel over the state space and P a distribution induced by a generative model (diffusion or VAE).

Reverse process. By starting from samples of the prior distribution at time K and reversing the diffusion process, we can obtain samples from the conditional next-state distribution at diffusion time 0. A result by Anderson (Anderson, 1982; Song et al., 2020) states that the reverse of a diffusion process is also a diffusion process, running backward in time and governed by the reverse-time stochastic differential equation

$$dx_k = \big(b(x_k, k) - r^2(k)\nabla_{x_k} \log P_k(x_k \mid s, a)\big)\, dk + r(k)\, d\bar{w}_k, \tag{3}$$

where $\bar{w}_k$ is a Brownian motion when time flows backward from K to 0. If the marginal score $\nabla_{x_k} \log P_k(x_k \mid s, a)$ were known for all k, we could simulate the reverse process to generate samples from the conditional next-state distribution. In practice, we approximate the unknown marginal score with a neural score model $u_\xi(x_k, k, s, a)$, parameterized by $\xi \in \mathbb{R}^d$. We train the score model using the standard denoising score-matching objective, which replaces the unknown marginal score with the tractable conditional score of the forward diffusion process:

$$J(\xi) = \frac{1}{2} \int_0^K \mathbb{E}_{\mathcal{D},\mathsf{P}} \left[\mathbb{E}_{x_k \sim P_{k|0}(\cdot|s',s,a)} \left[r^2(k) \left\| \nabla_{x_k} \log P_{k|0}(x_k \mid s', s, a) - u_\xi(x_k, k, s, a) \right\|^2 \right] \right] dk.$$

After training, the learned reverse diffusion process induces a conditional generative transition kernel, denoted by $\mathsf{P}_\xi(\cdot \mid s, a)$, which serves as the learned generative model for the transition kernel.

The denoising score-matching objective provides an empirical measure of how well the diffusion model fits the offline dataset. We then define the diffusion-based uncertainty set as

$$\Xi := \{\xi : J(\xi) \le \tau\}, \qquad \mathcal{P} := \{\mathsf{P}_\xi : \xi \in \Xi\}, \tag{4}$$

where τ is a prescribed tolerance level and P_ξ denotes the transition kernel induced by the diffusion model with parameters ξ.

Conditional VAE. We next consider a conditional variational autoencoder (VAE) as the generative model for transition dynamics. A conditional VAE introduces an auxiliary latent variable z and parameterizes this transition distribution as

$$\mathsf{P}_\xi(s' \mid s, a) = \int P_\xi(s' \mid s, a, z')\, P_\xi(z' \mid s, a)\, dz'. \tag{5}$$

Here, $P_\xi(z' \mid s, a)$ is the conditional prior, or latent transition predictor, and $P_\xi(s' \mid s, a, z)$ is the decoder. For notational simplicity, we use ξ to denote the parameters of both components.

We introduce an encoder $q_\zeta(z' \mid s, a, s')$ that maps a real next-state sample s' to the latent space. The model is trained by minimizing the negative conditional ELBO:

$$J_{\mathrm{VAE}}(\xi, \zeta) = \mathbb{E}_{\substack{(s,a)\sim\mathcal{D},\ s'\sim\mathsf{P}(\cdot|s,a) \\ z'\sim q_\zeta(\cdot|s,a,s')}} \left[-\log p_\xi(s' \mid s, a, z') + \log \frac{q_\zeta(z' \mid s, a, s')}{p_\xi(z' \mid s, a)} \right]. \tag{6}$$

This objective trains the decoder to predict the next state from (s, a, z'), while regularizing the encoder toward the conditional prior. To define the uncertainty set for the conditional VAE world model, we first train the model on the offline transition dataset. Let $(\xi_0, \zeta_0) \in \arg\min_{\xi,\zeta} J_{\mathrm{VAE}}(\xi, \zeta)$. After this pretraining step, we fix the encoder q_{ζ_0} and define an uncertainty set over the generative parameters ξ:

$$\Xi_{\mathrm{VAE}} = \{\xi : J_{\mathrm{VAE}}(\xi, \zeta_0) \le \tau\}, \mathcal{P}_{\mathrm{VAE}} = \{\mathsf{P}_\xi : \xi \in \Xi_{\mathrm{VAE}}\}. \tag{7}$$

We keep the encoder fixed when defining the uncertainty set because it is used only during pretraining and does not participate in the generative sampling process used for subsequent policy optimization.

5 Robust Algorithm and Convergence Results

In this section, we introduce our design of robust RL with a generative model-induced uncertainty set and its convergence analysis. Specifically, we will solve the robust RL problem in eq. (1) with diffusion model induced uncertainty set in eq. (4) and VAE induced uncertainty set in eq. (7).

5.1 Algorithm Design

To solve the problem in eq. (1), we take a robust policy gradient approach (Wang et al., 2023a; Li et al., 2022; Wang et al., 2023a; Sun et al., 2024; Li et al., 2026). Let the policy be parameterized by θ: π_θ. The robust objective in eq. (1) is generally nondifferentiable. As shown in Lemma 2.2 of Li et al. (2022), its Fréchet supergradient can be obtained by evaluating the policy gradient under its worst-case transition kernel:

$$h(\theta) = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi_\theta,\rho}^{\mathsf{P}_{\xi^\star}},\ s'\sim\mathsf{P}_{\xi^\star}(\cdot|s,a)} \left[\nabla_\theta \log \pi_\theta(a|s) \left(R(s,a) + \gamma V_{\pi_\theta}^{\mathsf{P}_{\xi^\star}}(s') \right) \right]. \tag{8}$$

where $\mathsf{P}_{\xi^*} \in \arg\min_{\mathsf{P}_\xi\in\mathcal{P}} V_{\pi_\theta}^{\mathsf{P}_\xi}(\rho)$ denotes the worst-case transition kernel in the uncertainty set given policy π_θ. Thus, solving eq. (1) requires identifying the worst-case transition kernel within the generative-model-induced uncertainty set and evaluating its value function.

We adopt an adversarial training approach that updates the generative model's parameters using gradients of the robust value function w.r.t. the generative model parameter ξ. To ensure that the adversarially updated generative model remains within the prescribed uncertainty set, we use a primal-only method (Xu et al., 2021; Lan & Zhou, 2020): when the uncertainty constraint is satisfied, the generative model is updated to decrease the policy value; when the constraint is violated, the update is redirected toward reducing the constraint violation. After robust policy evaluation, we perform robust policy optimization using eq. (8). We then derive the gradient with respect to the transition-kernel parameter ξ in the following proposition.

Proposition 1 (Model gradient)**.** *The gradient of $V_\pi^{P_\xi}(\rho)$ with respect to the transition-kernel parameter ξ is given by*

$$\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\rho) = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_\xi}} \left[\mathbb{E}_{s'\sim\mathsf{P}_\xi(\cdot|s,a)} \left[\nabla_\xi \log \mathsf{P}_\xi(s' \mid s, a) G_\pi^{\mathsf{P}_\xi}(s,a,s') \right] \right], \tag{9}$$

where $G_\pi^{\mathsf{P}_\xi}(s,a,s') = \gamma V_\pi^{\mathsf{P}_\xi}(s') + R(s,a) - V_\pi^{\mathsf{P}_\xi}(s)$.

The proof of Proposition 1 is provided in Appendix A.2. Since $P_\xi(s' \mid s, a)$ is not directly accessible for either diffusion or VAE-based models, we further provide the corresponding model gradients for

diffusion models:

$$\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\rho) = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_\xi}}\left[\mathbb{E}_{x_{0:K}\sim P_\xi(\cdot|s,a)}\left[\nabla_\xi \log P_\xi(x_{0:K} \mid s,a)\, G_\pi^{\mathsf{P}_\xi}(s,a,x_0)\right]\right]. \quad (10)$$

An analogous gradient holds for conditional VAE models, and we defer its derivation to Appendix A.2. We are now ready to present our Algorithm 1. The overall algorithm follows a double loop structure scheme: **Inner loop**: worst-case transition kernel identification and evaluation of the corresponding value function, which is essentially a robust policy evaluation problem; and **Outer Loop**: robust policy improvement.

For the inner loop, in order to find the worst-case transition kernel, a naive approach is to run gradient descent using eq. (9). But the key challenge is that we also need to enforce the constraint that the resulting generative model is still in the uncertainty set, which is non-rectangular. Existing methods achieve this through a projection method, e.g., (Wang & Si, 2026; Wang et al., 2023a; Li et al., 2026), which, however, requires a linear affine structure of the uncertainty set, and therefore cannot be applied to our problem.

In this paper, we develop a primal-only method. More specifically, given the predefined loss threshold τ and tolerance η, if $J(\xi) \geq \tau + \eta$, i.e., the model ξ is outside of the uncertainty set, we update the model in the direction of $\nabla_\xi J(\xi)$ to reduce the loss; otherwise, we use the approximate-natural-gradient direction (will be defined later in Section 5.2.2). To estimate the model gradient, we use Monte Carlo and sample from the current policy and generative model and use the learned value function to estimate the gradient.

For the outer loop, we perform a robust policy gradient $\theta \leftarrow \theta + \beta h(\theta)$ with respect to the value function induced by the worst-case transition kernel in the inner loop, where $h(\theta)$ is defined in eq. (8) and β is the outer loop learning rate.

Algorithm 1 Robust RL with Generative Model Induced Uncertainty Set

1: **Input:** Offline dataset $\mathcal{D}$, generative model parameter ξ_0, initial policy parameter θ_0.
2: **for** $t = 0, 1, \ldots, T-1$ **do**
3: Find the worst-case model ξ_t using Algorithm 2.
4: Update θ_{t+1} using the robust policy gradient in eq. (8).
5: **end for**

Algorithm 2 Robust Policy Evaluation

1: **Initialize:** diffusion model parameters ξ_t; policy π_{θ_t}.
2: Set $\xi_t^0 \leftarrow \xi_t$
3: **for** $i = 0, 1, \ldots, N-1$ **do**
4: Compute the uncertainty loss $J(\xi_t^i)$.
5: **if** $J(\xi_t^i) \leq \tau + \eta$ **then**
6: Sample trajectories $\mathcal{D}_i = \{(s_0, a_0, s_1, \ldots)\}$ from $\mathsf{P}_{\xi_t^i}$ under policy π_{θ_t}.
7: Train the critic $V_{\pi_{\theta_t}}^{\mathsf{P}_{\xi_t^i}}$ with $\mathcal{D}_i$.
8: Update the model by a natural-gradient step as in eq. (14)
9: **else**
10: Update the model parameters to reduce the constraint violation: $\xi_t^{i+1} \leftarrow \xi_t^i - \alpha \nabla_\xi J(\xi_t^i)$.
11: **end if**
12: **end for**

5.2 Convergence Results

In this section, we present convergence results. We show that the inner robust policy evaluation is guaranteed to find the worst-case parameter in Ξ under mild conditions, and that the outer policy optimization loop converges to a stationary point. The lack of global optimality in the outer loop primarily stems from the non-convexity of the policy network; global convergence might be possible if stricter assumptions are enforced. Detailed proofs are provided in Appendices A.2 and A.3. We first establish several useful auxiliary lemmas.

5.2.1 TRANSLATION OF THE GENERATIVE MODEL INDUCED UNCERTAINTY SET

In this section, we show that the uncertainty sets induced by diffusion models and conditional VAEs admit equivalent KL-based characterizations. Specifically, the diffusion-based uncertainty set can be expressed as a KL constraint over the induced diffusion-trajectory distributions, whereas the conditional VAE-based uncertainty set can be expressed as a KL constraint over the joint distribution of the latent variable and the generated sample.

Lemma 1. *The uncertainty set induced from the diffusion model can be equivalently written as*

$$\Xi = \left\{ \xi : \mathbb{E}_{(s,a)\sim\mathcal{D}} \Big[\mathrm{KL}(P(x_{0:K} \mid s,a) \,\|\, P_\xi(x_{0:K} \mid s,a)) \Big] \le \tau - C_1 + C_2 \right\}. \tag{11}$$

where C_1 and C_2 are ξ-independent constants. Similarly, the uncertainty set induced by the conditional VAE can be equivalently written as

$$\Xi_{\mathrm{VAE}} = \left\{ \xi : \mathbb{E}_{(s,a)\sim\mathcal{D}} \left[\mathrm{KL}\Big(q_{\zeta_0,P}(s', z \mid s,a) \,\Big\|\, P_\xi(s', z \mid s,a) \Big) \right] \le \tau' \right\}, \tag{12}$$

where $\tau' = \tau - \mathbb{E}_{(s,a)\sim\mathcal{D}} \Big[\mathcal{H}(\mathsf{P}(\cdot \mid s,a)) \Big]$.

The main difference between the generative uncertainty set and KL-based constraints is that the true model distribution appears on the left-hand side, yielding a reverse KL constraint. This makes our setting fundamentally different from both KL-regularized RL and KL-constrained RL.

Remark 1. *Lemma 1 is central to the inner-loop convergence analysis. It converts the generative model training loss, up to a constant independent of ξ, into a reverse-KL constraint. This allows us to analyze the two update rules using the same KL-based measure and combine their progress bounds.*

5.2.2 INNER LOOP CONVERGENCE RESULTS

To establish convergence of the inner loop, we make assumptions on the generative world model as in (Agarwal et al., 2021). We next state the key assumptions and define the constants appearing in the theorem below. The complete list of assumptions is deferred to the Appendix. Throughout this section, we suppress the time index t and treat π as fixed. Let ξ^* denote the worst-case transition-kernel parameter solving $\min_{\xi\in\Xi} V_\pi^{\mathsf{P}_\xi}(\rho)$, and let $\mathsf{P}^* = \mathsf{P}_{\xi^*}$.

Assumption 1. *1. The log-likelihood $\log P_\xi(x_{0:K}|s,a)$ is l-smooth.*
2. The KL divergence between the worst-case transition kernel and the initial model is bounded as

$$\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}} \left[\mathrm{KL}(P^*(x_{0:K} \mid s,a) \,\|\, P_{\xi_0}(x_{0:K} \mid s,a)) \right] \le B_{\mathrm{KL}}. \tag{13}$$

3. There exists a constant $\lambda > 0$ such that $\inf_{\xi: J(\xi)\ge\tau+\eta} \|\nabla_\xi J(\xi)\|^2 \ge \lambda\eta$.

Consider the diffusion model first. To analyze the inner-loop update, we define

$$w_i^* = \underset{\|w\|_2\le W}{\arg\min}\, \mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P_{\xi_i}}} \left[\mathbb{E}_{x_{0:K}\sim P_{\xi_i}(\cdot|s,a)} \Big[\big(w^\top \nabla_\xi \log P_{\xi_i}(x_{0:K} \mid s,a) - G(s,a,x_0) \big)^2 \Big] \right],$$

and when $J(\xi_i) \le \tau + \eta$, we update the generative model parameter as follows:

$$\xi_{i+1} = \xi_i - \alpha w_i^*, \tag{14}$$

where W is the boundedness constant used in the approximate-natural gradient analysis and appears in the final convergence bound. The update for the conditional VAE world model is analogous and is omitted for brevity.

We are now ready to state the convergence result for the inner loop. We note that the convergence results apply to both diffusion and VAE. Let ξ_{out} be selected uniformly at random from $\{\xi_i : J(\xi_i) \le \tau + \eta,\ i \in [0, N-1]\}$.

Theorem 1 (Convergence of robust policy evaluation)**.** *In Algorithm 2, let $N \ge \frac{2B_{\mathrm{KL}} l}{W^2}$ and choose*

$$\alpha = \sqrt{\frac{2B_{\mathrm{KL}}}{lW^2N}} \quad \textit{and} \quad \eta = 4\sqrt{\frac{2B_{\mathrm{KL}} lW^2}{\lambda^2 N}} + \frac{4\max\{\epsilon_1,\epsilon_2\}}{\lambda}.$$

Then

$$\mathbb{E}\Big[V_{\pi}^{P_{\xi_{out}}}(\rho) - V_{\pi}^{P^*}(\rho)\Big] \leq \sqrt{\frac{8B_{\mathrm{KL}}lW^2}{(1-\gamma)^2 N}} + \frac{2\max\{\epsilon_1, \epsilon_2\}}{1-\gamma}, \tag{15}$$

and

$$\mathbb{E}[J(\xi_{out})] - \tau \leq \eta. \tag{16}$$

Here, ϵ_1 captures world-model approximation and statistical errors, while ϵ_2 measures the distribution mismatch between the offline data and policy execution under the worst-case transition kernel; see Appendix A.2 for its definition.

As shown in Theorem 1, starting from the initial generative model parameter ξ_0, the inner robust policy evaluation procedure is guaranteed to identify the worst-case transition-kernel parameter up to small errors determined by the representational capacity of the generative model. The non-vanishing terms in eq. (15) and eq. (16) as $N \to \infty$ mainly arise from function approximation error and the distribution mismatch between the offline dataset and the occupancy measure induced by the worst-case transition kernel. Moreover, eq. (16) guarantees that the output world-model parameter ξ_{out} remains within the uncertainty set up to the same type of approximation and distribution-shift errors.

5.2.3 Outer Loop Convergence

In this section, we present the outer loop convergence results. Let $\Phi(\theta, \xi) = V_{\pi_\theta}^{P_\xi}(\rho)$ and $\Psi(\theta) = \min_{\xi \in \Xi} V_{\pi_\theta}^{P_\xi}(\rho)$. We assume that $\log \pi_\theta(a \mid s)$ is l_θ-smooth and that $\|\nabla_\theta \log \pi_\theta(a \mid s)\| \leq B_\Theta$ for all (s, a). Then, by Lemma 3.4 of (Zhang et al., 2020), $\Phi(\theta, \xi)$ is smooth with respect to θ, i.e., $\|\nabla_\theta \Phi(\theta_1, \xi) - \nabla_\theta \Phi(\theta_2, \xi)\| \leq L_\Theta \|\theta_1 - \theta_2\|$, where $L_\Theta = \frac{l_\theta}{(1-\gamma)^2} + \frac{(1+\gamma)B_\Theta^2}{(1-\gamma)^3}$. Ψ is L_π-Lipschitz continuous, where $L_\pi = \frac{B_\Theta}{(1-\gamma)^2}$. Then we define the Moreau envelope of Ψ as $\Psi_{\frac{1}{2L_\Theta}}(\theta) = \max_{\theta'} \Big\{ \Psi(\theta') - L_\Theta \|\theta' - \theta\|^2 \Big\}$.

Theorem 2 (Convergence of robust policy improvement)**.** *In algorithm 1, the average squared norm of the gradient of the Moreau envelope satisfies*

$$\frac{1}{T}\sum_{t=0}^{T-1} \Big\| \nabla_\theta \Psi_{\frac{1}{2L_\Theta}}(\theta_t) \Big\|^2 \leq 4\sqrt{\frac{L_\pi^3 L_\Theta \|\theta_T - \theta_0\|}{T}} + \frac{4L_\Theta}{T}\sum_{t=0}^{T-1} \epsilon_t. \tag{17}$$

Here, ϵ_t denotes the suboptimality gap between the value under the model parameter ξ_t and the worst-case model P_t^*:

$$\epsilon_t = V_{\pi_t}^{P_{\xi_t}}(\rho) - V_{\pi_t}^{P_t^*}(\rho). \tag{18}$$

The parameter ξ_t is obtained by Algorithm 2, whose convergence guarantee is given in eq. (15)*.*

Theorem 2 provides a stationarity guarantee through the Moreau envelope of Ψ. This formulation is necessary because $\Psi(\theta)$ is generally nondifferentiable (Wang et al., 2023a). Moreover, since the inner-loop error ϵ_t contains terms that do not vanish as $N \to \infty$, the final convergence bound includes a non-vanishing residual term.

6 Experiments

We evaluate the robustness of our proposed algorithm on standard offline RL benchmarks under controlled perturbations. We consider eight perturbation settings that introduce noise into the state or action space or modify the agent's body mass or environmental friction.

For the MuJoCo experiments, we use the D4RL (Fu et al., 2020) offline datasets for three tasks—HalfCheetah, Hopper, and Walker2d—under three dataset regimes: medium, medium-replay, and medium-expert. All results are reported over three random seeds. We compare our algorithm with the model-free methods CQL (Kumar et al., 2020) and IQL (Kostrikov et al., 2021), as well as the model-based methods RAMBO (Rigter et al., 2022), MOPO (Yu et al., 2020), and MOBILE (Sun

et al., 2023). Model-Based SAC denotes a standard SAC agent trained on synthetic transitions generated by the CVAE world model, whereas our algorithm uses the same SAC backbone with robust policy evaluation. For all other methods, we evaluate the released checkpoints from OfflineRLKit (Yihao Sun, 2023).

We use two complementary robustness metrics. Table 1 reports the worst-case return for each method and task, defined as the lowest mean return across the eight perturbation settings. Table 2 reports the variance of the mean returns across these settings, where a lower value indicates more consistent performance under perturbations. Further details on the experimental protocol and perturbation configurations are provided in Section B in the appendix.

Aggregated across the nine task–dataset combinations, our algorithm achieves the highest worst-case return and the lowest disturbance variance. These results demonstrate that our algorithm provides stronger and more consistent robustness to environmental perturbations than the considered baselines. Per-perturbation results and additional experiments using diffusion-based world models are provided in Section C in the appendix.

Table 1: Worst-case normalized D4RL scores over eight disturbance settings (mean $\pm$ population standard deviation across three seeds). The final row reports the sum of the worst-case means across all nine tasks.

Dataset	Model-based SAC	CQL	MOPO	IQL	RAMBO	Mobile	**GIRO**
HC-M	66.2 ± 1.8	47.7 ± 0.2	69.9 ± 5.6	46.3 ± 0.4	76.2 ± 0.8	70.7 ± 5.3	64.6 ± 1.3
HC-MR	32.8 ± 12.1	42.0 ± 0.4	55.5 ± 1.2	39.5 ± 2.1	60.0 ± 0.4	56.3 ± 1.4	47.1 ± 1.1
HC-ME	62.9 ± 7.0	68.0 ± 4.4	45.8 ± 5.1	65.9 ± 1.4	68.8 ± 6.9	62.8 ± 6.1	68.8 ± 4.4
Hopper-M	40.4 ± 29.5	48.3 ± 0.9	30.9 ± 11.7	57.1 ± 3.8	53.2 ± 17.7	82.4 ± 29.0	97.9 ± 1.4
Hopper-MR	35.9 ± 2.5	70.0 ± 28.1	83.5 ± 19.2	65.9 ± 9.7	73.4 ± 20.8	80.1 ± 13.5	95.9 ± 0.8
Hopper-ME	63.4 ± 14.4	75.8 ± 2.0	63.9 ± 42.1	56.8 ± 10.9	96.2 ± 4.9	78.7 ± 4.4	78.3 ± 4.6
Walker2d-M	68.8 ± 9.8	82.0 ± 1.1	82.5 ± 2.3	74.5 ± 8.5	86.3 ± 1.2	75.5 ± 8.0	83.9 ± 0.6
Walker2d-MR	79.2 ± 5.6	63.4 ± 18.7	73.5 ± 16.0	65.7 ± 6.9	72.3 ± 8.4	87.8 ± 0.7	79.1 ± 8.6
Walker2d-ME	97.4 ± 13.9	108.1 ± 1.4	106.1 ± 5.5	102.9 ± 2.1	62.5 ± 37.0	112.8 ± 2.5	98.7 ± 16.1
Worst-Case Sum $\uparrow$	547.0	605.4	611.6	574.6	648.9	707.1	**714.2**

Table 2: Variance of the mean returns across the eight disturbance settings (lower is better). The last row sums the variances over all nine tasks.

Dataset	Model-based SAC	CQL	MOPO	IQL	RAMBO	Mobile	**GIRO**
HC-M	1.46	0.91	3.47	0.37	0.73	2.69	0.51
HC-MR	67.95	4.42	45.01	3.02	13.57	38.99	14.94
HC-ME	94.10	56.43	83.69	77.64	85.40	158.32	60.34
Hopper-M	72.51	39.44	156.55	78.60	36.56	64.08	0.59
Hopper-MR	51.85	74.52	54.28	88.00	33.06	88.87	1.38
Hopper-ME	52.85	101.12	42.13	205.16	7.90	120.68	28.64
Walker2d-M	23.20	0.56	3.17	5.84	0.08	21.67	0.15
Walker2d-MR	4.84	43.16	12.76	11.97	3.21	0.46	9.88
Walker2d-ME	24.35	0.19	5.94	3.46	18.54	0.33	21.34
Variance Sum $\downarrow$	393.12	320.76	407.00	474.06	199.06	496.09	**137.78**

7 Conclusion

In this work, we study the problem of distributionally robust RL with a generative-model-induced uncertainty set. To solve this problem, we introduce a novel algorithm of GIRO, which adversarially updates the generative model to find the worst-case value function for robust policy evaluation, and performs robust policy improvement. We analyze the theoretical convergence of GIRO and establish guarantees for both its inner and outer optimization loops. Compared with well-established baselines, our algorithm achieves superior worst-case performance and exhibits greater consistency under environmental perturbations.

8 USE OF LARGE LANGUAGE MODELS (LLMS)

In accordance with the ICLR guidelines, we disclose the use of large language models (LLMs) in the preparation of this paper. LLMs were used solely as writing assistance tools for language polishing, grammar correction, and improving readability. They were not used for research ideation, experimental design, data analysis, or the generation of scientific content. All technical ideas, theoretical developments, proofs, and experimental results presented in this paper are the authors' own work. The authors take full responsibility for the final content of the submission.

9 REPRODUCIBILITY STATEMENT

We have made several efforts to ensure reproducibility. For the theoretical results, all assumptions are stated in Section 5.2 and in full in Appendices A.2 and A.3, and complete proofs of Lemma 1, Proposition 1, and Theorems 1 and 2 are provided in Appendix A. For the experiments, we use the publicly available D4RL datasets, and our implementation builds on the open-source OfflineRL-Kit codebase (Appendix B.1). Appendix B describes the world-model training, the robust policy training, and the differences between the implementation and the analyzed algorithm. The hyperparameters are listed in Tables 3–5, and the evaluation perturbations are defined in Appendix C.1.

REFERENCES

Alekh Agarwal, Sham Kakade, and Lin F Yang. Model-based reinforcement learning with a generative model is minimax optimal. In *Conference on Learning Theory*, pp. 67–83. PMLR, 2020.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *Journal of Machine Learning Research*, 22(98):1–76, 2021.

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

Ben Black, Trivikram Dokka, and Christopher Kirkbride. Robust markov decision processes under parametric transition distributions. *arXiv preprint arXiv:2211.07488*, 2022.

Julien Grand Clement and Christian Kroer. First-order methods for wasserstein distributionally robust mdp. In *International conference on machine learning*, pp. 2010–2019. PMLR, 2021.

Zihan Ding, Amy Zhang, Yuandong Tian, and Qinqing Zheng. Diffusion world model: Future modeling beyond step-by-step rollout for offline reinforcement learning. *arXiv preprint arXiv:2402.03570*, 2024.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Alison L Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435, 2002.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. *Advances in neural information processing systems*, 31, 2018.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Chin Pang Ho, Marek Petrik, and Wolfram Wiesemann. Efficient algorithms for robust markov decision processes with s-rectangular ambiguity sets. *arXiv preprint arXiv:2602.05591*, 2026.

Zhennan Jiang, Kai Liu, Yuxin Qin, Shuai Tian, Yupeng Zheng, Mingcai Zhou, Chao Yu, Haoran Li, and Dongbin Zhao. World4rl: Diffusion world models for policy refinement with reinforcement learning for robotic manipulation. *arXiv preprint arXiv:2509.19080*, 2025.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in neural information processing systems*, 33:1179–1191, 2020.

Guanghui Lan and Zhiqiang Zhou. Algorithms for stochastic optimization with function or expectation constraints: G. lan, z. zhou. *Computational Optimization and Applications*, 76(2):461–498, 2020.

Gen Li, Changxiao Cai, Yuxin Chen, Yuting Wei, and Yuejie Chi. Is q-learning minimax optimal? a tight sample complexity analysis. *Operations Research*, 72(1):222–236, 2024.

Mengmeng Li, Daniel Kuhn, and Tobias Sutter. Policy gradient algorithms for robust mdps with nonrectangular uncertainty sets. *SIAM Journal on Optimization*, 36(1):120–151, 2026. doi: 10.1137/24M1631250.

Yan Li, Guanghui Lan, and Tuo Zhao. First-order policy optimization for robust markov decision process. *arXiv preprint arXiv:2209.10579*, 2022.

Zongyue Li, Xiao Han, Yusong Li, Niklas Strauss, and Matthias Schubert. Dawm: Diffusion action world models for offline reinforcement learning via action-inferred transitions. *arXiv preprint arXiv:2509.19538*, 2025.

Xiaoteng Ma, Zhipeng Liang, Jose Blanchet, Mingwen Liu, Li Xia, Jiheng Zhang, Qianchuan Zhao, and Zhengyuan Zhou. Distributionally robust offline reinforcement learning with linear function approximation. *arXiv preprint arXiv:2209.06620*, 2022.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. *arXiv preprint arXiv:2209.00588*, 2022.

Wenfan Ou and Sheng Bi. Sequential decision-making under uncertainty: a robust mdps review. *Annals of Operations Research*, 353(3):1239–1285, 2025.

Kishan Panaganti and Dileep Kalathil. Sample complexity of robust reinforcement learning with a generative model. In *International Conference on Artificial Intelligence and Statistics*, pp. 9582–9602. PMLR, 2022.

Marc Rigter, Bruno Lacerda, and Nick Hawes. Rambo-rl: Robust adversarial model-based offline reinforcement learning. *Advances in neural information processing systems*, 35:16082–16097, 2022.

Laixi Shi, Gen Li, Yuting Wei, Yuxin Chen, Matthieu Geist, and Yuejie Chi. The curious price of distributional robustness in reinforcement learning with a generative model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Yihao Sun, Jiaji Zhang, Chengxing Jia, Haoxin Lin, Junyin Ye, and Yang Yu. Model-bellman inconsistency for model-based offline reinforcement learning. In *International Conference on Machine Learning*, pp. 33177–33194. PMLR, 2023.

Zhongchang Sun, Sihong He, Fei Miao, and Shaofeng Zou. Policy optimization for robust average reward mdps. *Advances in Neural Information Processing Systems*, 37:17348–17372, 2024.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Kiran K Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Efficient algorithms for smooth minimax optimization. *Advances in neural information processing systems*, 32, 2019.

Qiuhao Wang, Chin Pang Ho, and Marek Petrik. Policy gradient in robust mdps with global convergence guarantee. In *International conference on machine learning*, pp. 35763–35797. PMLR, 2023a.

Shengbo Wang and Nian Si. Non-rectangular average-reward robust mdps: Optimal policies and their transient values. *arXiv preprint arXiv:2603.00945*, 2026.

Wenlong Wang, Ivana Dusparic, Yucheng Shi, Ke Zhang, and Vinny Cahill. Drama: Mamba-enabled model-based reinforcement learning is sample and parameter efficient. In *International Conference on Learning Representations*, volume 2025, pp. 75688–75712, 2025.

Yudan Wang, Shaofeng Zou, and Yue Wang. Model-free robust reinforcement learning with sample complexity analysis. In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, UAI '24. JMLR.org, 2024.

Yue Wang, Alvaro Velasquez, George Atia, Ashley Prater-Bennette, and Shaofeng Zou. Robust average-reward markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 15215–15223, 2023b.

Jiaqi Wen and Jianyi Yang. Distributionally robust optimization via generative ambiguity modeling. *arXiv preprint arXiv:2602.08976*, 2026.

Tengyu Xu, Yingbin Liang, and Guanghui Lan. Crpo: A new approach for safe reinforcement learning with convergence guarantee. In *International Conference on Machine Learning*, pp. 11480–11491. PMLR, 2021.

Zhilong Zhang Yihao Sun. Offlinerl-kit: An elegant pytorch offline reinforcement learning library. `https://github.com/yihaosun1124/OfflineRL-Kit`, 2023.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in neural information processing systems*, 33:14129–14142, 2020.

Kaiqing Zhang, Alec Koppel, Hao Zhu, and Tamer Basar. Global convergence of policy gradient methods to (almost) locally optimal policies. *SIAM Journal on Control and Optimization*, 58(6): 3586–3612, 2020.

A Theoretical Proofs

A.1 KL Reformulation of Generative Model Induced Uncertainty Sets

Starting with the diffusion world model, we first show that $J(\xi)$ is equivalent to the score-matching loss up to an additive constant that is independent of ξ:

$$J(\xi) = J_{\mathrm{SM}}(\xi) + C_1, \tag{19}$$

where

$$C_1 = \frac{1}{2}\int_0^K \mathbb{E}_{\mathcal{D},\mathrm{P}}\left[\mathbb{E}_{x_k\sim P_{k|0}(\cdot|s',s,a)}\left[r^2(k)\left(\left\|\nabla_{x_k}\log P_{k|0}(x_k \mid s',s,a)\right\|^2 - \left\|\nabla_{x_k}\log P_k(x_k \mid s,a)\right\|^2\right)\right]\right]dk. \tag{20}$$

For each (s,a), define the conditional score-matching objective as

$$J_{\mathrm{SM}}(\xi;s,a) \coloneqq \frac{1}{2}\int_0^K \mathbb{E}_{x_k\sim P_k(\cdot|s,a)}\left[r^2(k)\|\nabla_{x_k}\log P_k(x_k \mid s,a) - u_\xi(x_k,k,s,a)\|^2\right]dk. \tag{21}$$

By Girsanov's theorem,

$$J_{\mathrm{SM}}(\xi; s, a) = \mathrm{KL}(P(x_{0:K} \mid s, a) \| P_\xi(x_{0:K} \mid s, a)) - \mathrm{KL}(P_K(\cdot \mid s, a) \| \nu), \tag{22}$$

where $\nu = \mathcal{N}(0, I)$ and $P_\xi(x_{0:K} \mid s, a)$ denotes the conditional trajectory distribution induced by the diffusion world model. The terminal-distribution term is independent of ξ, and

$$C_2 \coloneqq \mathbb{E}_{(s,a)\sim\mathcal{D}} \left[\mathrm{KL}(P_K(\cdot \mid s, a) \| \nu)\right] \geq 0.$$

Since $\mathrm{KL}(P_K(\cdot|s, a)\|\nu)$ is ξ independent, we have that

$$\begin{aligned} J_{SM}(\xi) &= \mathbb{E}_{(s,a)\sim\mathcal{D}}[\mathrm{KL}(P(x_{0:K}|s, a)\|P_\xi(x_{0:K}|s, a))] - \mathbb{E}_{(s,a)\sim\mathcal{D}}[\mathrm{KL}(P_K(\cdot|s, a)\|\nu)] \\ &= \mathbb{E}_{(s,a)\sim\mathcal{D}}[\mathrm{KL}(P(x_{0:K}|s, a)\|P_\xi(x_{0:K}|s, a))] - C_2, \end{aligned} \tag{23}$$

where

$$C_2 = \mathbb{E}_{(s,a)\sim\mathcal{D}}[\mathrm{KL}(P_K(\cdot|s, a)\|\nu)], \tag{24}$$

and C_2 is also ξ independent. With eq. (19) and eq. (23), the uncertainty set in eq. (4) can be written as:

$$\Xi = \left\{\xi : \mathbb{E}_{(s,a)\sim\mathcal{D}} \left[\mathrm{KL}(P(x_{0:K} \mid s, a) \,\|\, P_\xi(x_{0:K} \mid s, a))\right] \leq \tau - C_1 + C_2\right\}. \tag{25}$$

For notational simplicity, define

$$\tau' \coloneqq \tau - C_1 + C_2. \tag{26}$$

Then

$$\Xi = \left\{\xi : \mathbb{E}_{(s,a)\sim\mathcal{D}} \left[\mathrm{KL}(P(x_{0:K} \mid s, a) \,\|\, P_\xi(x_{0:K} \mid s, a))\right] \leq \tau'\right\}. \tag{27}$$

For the Conditional VAE world model, when we fix the encoder parameter ζ_0, we have that

$$\begin{aligned} &J_{\mathrm{VAE}}(\xi, \zeta_0) \\ &= \mathbb{E}_{(s,a)\sim\mathcal{D},\, s'\sim P(\cdot|s,a)} \left[\mathbb{E}_{z\sim q_{\zeta_0}(\cdot|s,a,s')} \Big[-\log p_\xi(s' \mid s, a, z)\Big] + \mathrm{KL}\Big(q_{\zeta_0}(z \mid s, a, s')\|p_\xi(z \mid s, a)\Big)\right] \\ &= \mathbb{E}_{(s,a)\sim\mathcal{D}} \left[\mathbb{E}_{s'\sim P(\cdot|s,a),\, z\sim q_{\zeta_0}(\cdot|s,a,s')} \left[\log \frac{P(s' \mid s, a) q_{\zeta_0}(z \mid s, a, s')}{p_\xi(s' \mid s, a, z) p_\xi(z \mid s, a)}\right] + \mathcal{H}(P(\cdot \mid s, a))\right] \\ &= \mathbb{E}_{(s,a)\sim\mathcal{D}} \Big[\mathrm{KL}_{(s',z)}\Big(q_{\zeta_0,P}(s', z \mid s, a)\|P_\xi(s', z \mid s, a)\Big) + \mathcal{H}(P(\cdot \mid s, a))\Big], \end{aligned} \tag{28}$$

where

$$q_{\zeta_0,P}(s', z \mid s, a) \coloneqq P(s' \mid s, a) q_{\zeta_0}(z \mid s, a, s'), \tag{29}$$

$$P_\xi(s', z \mid s, a) \coloneqq P_\xi(s' \mid s, a, z) P_\xi(z \mid s, a). \tag{30}$$

The entropy term is independent of ξ. Therefore, the uncertainty set can be equivalently written as

$$\begin{aligned} \Xi_{\mathrm{VAE}} &= \left\{\xi : \mathbb{E}_{(s,a)\sim\mathcal{D}}\Big[\mathrm{KL}_{(s',z)}\Big(q_{\zeta_0,P}(s', z \mid s, a)\|P_\xi(s', z \mid s, a)\Big)\Big] \leq \tau'\right\}, \\ \tau' &\coloneqq \tau - \mathbb{E}_{(s,a)\sim\mathcal{D}}\Big[\mathcal{H}(P(\cdot \mid s, a))\Big]. \end{aligned} \tag{31}$$

A.2 Convergence of the Inner Loop

In this section, we prove the convergence of Algorithm 2 for finding an ϵ-optimal worst-case transition kernel, parameterized by ξ, for a given policy π. In this section, we omit the index t for notational simplicity. The key problem is to solve

$$\min_{\xi\in\Xi} V_\pi^{P_\xi}(\rho). \tag{32}$$

We first introduce several useful lemmas.

Lemma 2 (Performance Difference Lemma for Transition Kernels)**.** *For any two transition kernels P_1 and P_2, we have*

$$V_\pi^{\mathsf{P}_1}(\rho) - V_\pi^{\mathsf{P}_2}(\rho) = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_1}}\left[\mathbb{E}_{s'\sim\mathsf{P}_1(\cdot|s,a)}\left[G_\pi^{\mathsf{P}_2}(s,a,s')\right]\right], \tag{33}$$

where

$$G_\pi^{\mathsf{P}}(s,a,s') = \gamma V_\pi^{\mathsf{P}}(s') + R(s,a) - V_\pi^{\mathsf{P}}(s). \tag{34}$$

The discounted state–action occupancy density is defined as

$$d_{\pi,\rho}^{\mathsf{P}}(s,a) = (1-\gamma)\sum_{h=0}^{\infty}\gamma^h p_h^{\pi,\rho,\mathsf{P}}(s,a), \tag{35}$$

where $p_h^{\pi,\rho,\mathsf{P}}$ denotes the joint density of (s_h,a_h) with respect to a fixed reference measure, assuming these densities exist. The trajectory follows $s_0\sim\rho$, $a_h\sim\pi(\cdot\mid s_h)$, and $s_{h+1}\sim\mathsf{P}(\cdot\mid s_h,a_h)$. We use $d_{\pi,s}^{\mathsf{P}}$ to denote the corresponding discounted occupancy measure when $s_0=s$.

Proof. Expanding $V_\pi^{\mathsf{P}_1}$ using the Bellman equation, we have that

$$\begin{aligned}
&V_\pi^{\mathsf{P}_1}(\rho) - V_\pi^{\mathsf{P}_2}(\rho)\\
&= \mathbb{E}_{s_0\sim\rho,a_0\sim\pi(\cdot|s_0)}\left[R(s_0,a_0) + \gamma\mathbb{E}_{s_1\sim\mathsf{P}_1(\cdot|s_0,a_0)}\left[V_\pi^{\mathsf{P}_1}(s_1)\right] - V_\pi^{\mathsf{P}_2}(s_0)\right]\\
&= \mathbb{E}_{s_0\sim\rho,a_0\sim\pi(\cdot|s_0),s_1\sim\mathsf{P}_1(\cdot|s_0,a_0)}\left[\gamma\big(V_\pi^{\mathsf{P}_1}(s_1) - V_\pi^{\mathsf{P}_2}(s_1)\big) + R(s_0,a_0) + \gamma V_\pi^{\mathsf{P}_2}(s_1) - V_\pi^{\mathsf{P}_2}(s_0)\right]\\
&= \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_1},s'\sim\mathsf{P}_1(\cdot|s,a)}\left[\gamma V_\pi^{\mathsf{P}_2}(s') + R(s,a) - V_\pi^{\mathsf{P}_2}(s)\right]\\
&= \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_1},s'\sim\mathsf{P}_1(s'|s,a)}\left[G_\pi^{\mathsf{P}_2}(s,a,s')\right].
\end{aligned} \tag{36}$$

This completes the proof. □

Lemma 3 (Gradient with Respect to the Transition Kernel)**.** *The gradient of $V_\pi^{P_\xi}(\rho)$ with respect to the transition-kernel parameter ξ is given by*

$$\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\rho) = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_\xi}}\left[\mathbb{E}_{x_{0:K}\sim\mathsf{P}_\xi(\cdot|s,a)}\left[\nabla_\xi\log P_\xi(x_{0:K}|s,a)G_\pi^{\mathsf{P}_\xi}(s,a,x_0)\right]\right]. \tag{37}$$

Proof. First, since the initial distribution ρ does not depend on ξ, we can interchange the gradient

$$\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\rho) = \nabla_\xi\mathbb{E}_{\bar{s}\sim\rho}\left[V_\pi^{\mathsf{P}_\xi}(\bar{s})\right] = \mathbb{E}_{\bar{s}\sim\rho}\left[\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\bar{s})\right]. \tag{38}$$

Next, expand $V_\pi^{P_\xi}$ with Bellman equation, we have that

$$\begin{aligned}
&\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\bar{s})\\
&= \nabla_\xi\left(\mathbb{E}_{\bar{a}\sim\pi(\cdot|\bar{s})}\left[R(\bar{s},\bar{a}) + \gamma\mathbb{E}_{\bar{s}'\sim\mathsf{P}_\xi(\cdot|\bar{s},\bar{a})}\left[V_\pi^{\mathsf{P}_\xi}(\bar{s}')\right]\right]\right)\\
&\overset{(i)}{=} \gamma\mathbb{E}_{\bar{a}\sim\pi(\cdot|\bar{s})}\left[\nabla_\xi\mathbb{E}_{\bar{x}_{0:K}\sim P_\xi(\cdot|\bar{s},\bar{a})}\left[V_\pi^{\mathsf{P}_\xi}(\bar{x}_0)\right]\right]\\
&= \gamma\mathbb{E}_{\bar{a}\sim\pi(\cdot|\bar{s}),\bar{x}_{0:K}\sim P_\xi(\cdot|\bar{s},\bar{a})}\left[\nabla_\xi\log P_\xi(\bar{x}_{0:K}|\bar{s},\bar{a})V_\pi^{\mathsf{P}_\xi}(\bar{x}_0) + \nabla_\xi V_\pi^{\mathsf{P}_\xi}(\bar{x}_0)\right]\\
&= \frac{\gamma}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\bar{s}}^{\mathsf{P}_\xi}}\left[\mathbb{E}_{x_{0:K}\sim P_\xi(\cdot|s,a)}\left[\nabla_\xi\log P_\xi(x_{0:K}|s,a)V_\pi^{\mathsf{P}_\xi}(x_0)\right]\right]\\
&= \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\bar{s}}^{\mathsf{P}_\xi}}\left[\mathbb{E}_{x_{0:K}\sim P_\xi(\cdot|s,a)}\left[\nabla_\xi\log P_\xi(x_{0:K}|s,a)G_\pi^{\mathsf{P}_\xi}(s,a,x_0)\right]\right],
\end{aligned} \tag{39}$$

where (i) follows since $\bar{s}_0'$ is the generated next state associated with the diffusion trajectory $\bar{s}_{0:K}'$. Moreover, since

$$\mathbb{E}_{\bar{s}\sim\rho}\left[d_{\pi,\bar{s}}^{\mathsf{P}_\xi}(s,a)\right] = d_{\pi,\rho}^{\mathsf{P}_\xi}(s,a), \tag{40}$$

the desired result follows. This completes the proof. Similarly, the gradient with respect to the transition kernel for the VAE world model can be written as:

$$\nabla_\xi V_\pi^{\mathsf{P}_\xi}(\rho) = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{\mathsf{P}_\xi}}\Big[\mathbb{E}_{(s',z)\sim P_\xi(\cdot|s,a)}\Big[\nabla_\xi \log P_\xi(s',z|s,a) G_\pi^{\mathsf{P}_\xi}(s,a,s')\Big]\Big]. \tag{41}$$

□

Next, we derive separate bounds for the two update regimes, depending on whether $J(\xi_i) \le \tau + \eta$ or $J(\xi_i) > \tau + \eta$.

When $J(\xi_i) \le \tau + \eta$, from Algorithm 2, we update the transition model with approximate NPG update. The standard NPG update rule is

$$\xi_{i+1} = \xi_i - \alpha F(\xi_i)^\dagger \nabla_\xi V_\pi^{\mathsf{P}_{\xi_i}}(\rho). \tag{42}$$

Here, † denotes the Moore–Penrose pseudoinverse, and $F(\xi_i)$ is the Fisher information matrix:

$$F(\xi) = \mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P_\xi}}\Big[\mathbb{E}_{x_{0:K}\sim P_\xi(\cdot|s,a)}\Big[\nabla_\xi \log P_\xi(x_{0:K} \mid s,a)\,(\nabla_\xi \log P_\xi(x_{0:K} \mid s,a))^T\Big]\Big]. \tag{43}$$

It is helpful to define

$$L_G(w;\xi,d,P) = \mathbb{E}_{(s,a)\sim d}\Big[\mathbb{E}_{x_{0:K}\sim P(\cdot|s,a)}\Big[\Big(w^T\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a) - G_\pi^{\mathsf{P}_{\xi_i}}(s,a,x_0)\Big)^2\Big]\Big]. \tag{44}$$

The update rule w_i^* satisfies

$$w_i^* = \arg\min_{\|w\|\le W} L_G\Big(w;\xi_i, d_{\pi,\rho}^{P_{\xi_i}}, P_{\xi_i}\Big). \tag{45}$$

When $J(\xi_i) > \tau + \eta$, we minimize $J(\xi_i)$ with gradient descent. From Section 5.2.1 we have

$$J(\xi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[\mathrm{KL}(P(x_{0:K}|s,a)\|P_\xi(x_{0:K}|s,a))] + C_1 - C_2. \tag{46}$$

Since C_1 and C_2 are ξ independent, the gradient of $J(\xi)$ is

$$\nabla_\xi J(\xi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\big[\mathbb{E}_{x_{0:K}\sim P(\cdot|s,a)}[-\nabla_\xi \log P_\xi(x_{0:K}|s,a)]\big]. \tag{47}$$

The corresponding update rule is

$$\xi_{i+1} = \xi_i + \alpha w_i = \xi_i - \alpha\nabla_\xi J(\xi). \tag{48}$$

Next, we introduce some useful definitions and assumptions for the SDE-based diffusion model. Let $\xi^* = \arg\min_{\xi\in\Xi} V_\pi^{P_\xi}(\rho)$ and $P^* = P_{\xi^*}$.

Assumption 2.

1. $\log P_\xi(x_{0:K}|s,a)$ *is l-smooth.*
2. $\big\|\mathbb{E}_{(s,a)\sim\mathcal{D}}\big[\mathbb{E}_{x_{0:K}\sim P(\cdot|s,a)}[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)]\big]\big\| \le B_\xi$.
3. $\mathbb{E}_{(s,a)\sim\mathcal{D}}\Big[\mathbb{E}_{x_{0:K}\sim P(\cdot|s,a)}\Big[\|\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)\|^2\Big]\Big] \le M_\xi^2$.
4. $L_G(w_i^*;\xi_i, d_{\pi,\rho}^{P^*}, P^*) \le \epsilon_{bias}$.
5. $\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\mathrm{KL}(P^*(x_{0:K}|s,a)\|P_{\xi_0}(x_{0:K}|s,a))] \le B_{\mathrm{KL}}$.
6. $\sup_w \frac{w^T\Sigma_{d^*}^{(i)}w}{w^T\Sigma_\rho^{(i)}w} \le \kappa$, *and* $\sup_w \frac{w^T\Sigma_{d^*}^{(i)}w}{w^T\Sigma_{\mathcal{D},P}^{(i)}w} \le \kappa'$,

   where $\Sigma_\rho^{(i)} \coloneqq \mathbb{E}_{s\sim\rho,a\sim\pi,x_{0:K}\sim P_{\xi_i}}\Big[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)(\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T\Big]$,

   $\Sigma_{d^*}^{(i)} \coloneqq \mathbb{E}_{s\sim d_{\pi,\rho}^{P^*},a\sim\pi,x_{0:K}\sim P^*}\Big[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)(\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T\Big]$,

   and $\Sigma_{\mathcal{D},P}^{(i)} \coloneqq \mathbb{E}_{(s,a)\sim\mathcal{D},x_{0:K}\sim P}\Big[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)(\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T\Big]$.
7. *There exists* $\lambda > 0$ *such that* $\inf_{\xi:J(\xi)\ge\tau+\eta}\|\nabla_\xi J(\xi)\|^2 \ge \lambda\eta$.

Next, we will derive separate bounds for these two update paths.

A.2.1 CASE 1: $J(\xi_i) \le \tau + \eta$

The last assumption rules out stationary points outside the relaxed uncertainty set. Together with the smoothness of J and a sufficiently small step size, it ensures that each constraint-restoration update makes uniform progress toward the set $\{\xi : J(\xi) \le \tau + \eta\}$. When $J(\xi_i) \le \eta + \tau$, we have that

$$\begin{aligned}
&\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_i}(\cdot|s,a)) - \mathrm{KL}\big(P^*(\cdot|s,a)\|P_{\xi_{i+1}}(\cdot|s,a)\big)\big] \\
&= \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[\log \frac{P_{\xi_{i+1}}(x_{0:K}|s,a)}{P_{\xi_i}(x_{0:K}|s,a)}\right]\right] \\
&\overset{(i)}{\ge} \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)]\big](\xi_{i+1}-\xi_i) - \frac{l}{2}\|\xi_{i+1}-\xi_i\|^2 \\
&= -\alpha \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[G_\pi^{\mathsf{P}_{\xi_i}}(s,a,x_0)\right]\right] - \frac{\alpha^2 l}{2}\|w_i^*\|^2 \\
&\quad + \alpha \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[-(\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T w_i^* + G_\pi^{\mathsf{P}_{\xi_i}}(s,a,x_0)\right]\right], \qquad (49)
\end{aligned}$$

where (i) follows from the l-smoothness of $\log P_\xi(\cdot|s,a)$ with respect to ξ. From the performance difference lemma eq. (33), we have that

$$\begin{aligned}
&V_\pi^{P_{\xi_i}}(\rho) - V_\pi^{P^*}(\rho) \\
&= -\frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[G_\pi^{\mathsf{P}_{\xi_i}}(s,a,x_0)\right]\right] \\
&\le \frac{1}{\alpha(1-\gamma)}\Big(\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_i}(\cdot|s,a)) - \mathrm{KL}\big(P^*(\cdot|s,a)\|P_{\xi_{i+1}}(\cdot|s,a)\big)\big]\Big) \\
&\quad + \frac{\alpha l}{2(1-\gamma)}\|w_i^*\|^2 \\
&\quad + \frac{1}{1-\gamma}\underbrace{\mathbb{E}_{(s,a)\sim d^{\mathsf{P}^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[(\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T w_i^* - G_\pi^{\mathsf{P}_{\xi_i}}(s,a,x_0)\right]\right]}_{\mathrm{err}_i}. \qquad (50)
\end{aligned}$$

err_i can be bounded as

$$\begin{aligned}
\mathrm{err}_i &= \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[(\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T w_i^* - G_\pi^{P_{\xi_i}}(s,a,s')\right]\right] \\
&\le \sqrt{\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\left[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}\left[\left((\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a))^T w_i^* - G_\pi^{P_{\xi_i}}(s,a,s')\right)^2\right]\right]} \\
&= \sqrt{L_G(w_i^*;\xi_i,d^{P^*}_{\pi,\rho},P^*)} \le \sqrt{\epsilon_{\text{bias}}} =: \epsilon_1. \qquad (51)
\end{aligned}$$

A.2.2 CASE 2: $J(\xi_i) > \tau + \eta$

When $J(\xi) \ge \eta + \tau$, we update ξ to minimize $J(\xi)$. For notational convenience, let

$$g^{(i)}_{\mathcal{D},P} = \mathbb{E}_{(s,a)\sim\mathcal{D}}\big[\mathbb{E}_{x_{0:K}\sim P(\cdot|s,a)}[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)]\big], \qquad (52)$$

and

$$g^{(i)}_{d^*,P^*} = \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)]\big]. \qquad (53)$$

And the update rule is:

$$\xi_{i+1} = \xi_i + \alpha g^{(i)}_{\mathcal{D},P}, \qquad (54)$$

Similar to eq. (50), when $\alpha \le \frac{1}{l}$, we have that

$$
\begin{aligned}
&\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_i}(\cdot|s,a)) - \mathrm{KL}\big(P^*(\cdot|s,a)\|P_{\xi_{i+1}}(\cdot|s,a)\big)\big] \\
&\geq \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathbb{E}_{x_{0:K}\sim P^*(\cdot|s,a)}[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)]\big](\xi_{i+1}-\xi_i) - \frac{l}{2}\|\xi_{i+1}-\xi_i\|^2 \\
&= \alpha\Big(\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}, x_{0:K}\sim P^*(\cdot|s,a)}[\nabla_\xi \log P_{\xi_i}(x_{0:K}|s,a)]\Big)^T g^{(i)}_{\mathcal{D},P} - \frac{\alpha^2 l}{2}\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2 \\
&= \alpha\Big(g^{(i)}_{d^*,P^*} - g^{(i)}_{\mathcal{D},\mathcal{P}}\Big)^T g^{(i)}_{\mathcal{D},P} + \alpha(g^{(i)}_{\mathcal{D},\mathcal{P}})^T g^{(i)}_{\mathcal{D},P} - \frac{\alpha^2 l}{2}\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2 \\
&= -\alpha \mathrm{err}^{(i)}_{\text{shift}} + \left(\alpha - \frac{\alpha^2 l}{2}\right)\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2 \\
&\geq -\alpha \mathrm{err}^{(i)}_{\text{shift}} + \frac{\alpha}{2}\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2,
\end{aligned} \tag{55}
$$

where the distribution-shift error is defined as

$$
\mathrm{err}^{(i)}_{\text{shift}} := \Big(g^{(i)}_{\mathcal{D},P} - g^{(i)}_{d^*,P^*}\Big)^\top g^{(i)}_{\mathcal{D},P}. \tag{56}
$$

Rearranging, we have

$$
\frac{\alpha}{2}\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2 \leq \alpha \mathrm{err}^{(i)}_{\text{shift}} + \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_i}(\cdot|s,a)) - \mathrm{KL}\big(P^*(\cdot|s,a)\|P_{\xi_{i+1}}(\cdot|s,a)\big)\big]. \tag{57}
$$

Then, we proceed to bound the error terms. First, we bound $\mathrm{err}^{(i)}_{\text{shift}}$:

$$
\begin{aligned}
\mathrm{err}^{(i)}_{\text{shift}} &= \Big(g^{(i)}_{\mathcal{D},P} - g^{(i)}_{d^*,P^*}\Big)^T w_i \\
&\overset{(i)}{\leq} \sqrt{2}\,\mathrm{Hel}\Big(\mu_{d^{P^*}_{\pi,\rho},P^*}, \mu_{\mathcal{D},P}\Big)\Big(\sqrt{w_i^T \Sigma^{(i)}_{d^*} w_i} + \sqrt{w_i^T \Sigma^{(i)}_{\mathcal{D},P} w_i}\Big) \\
&\overset{(ii)}{\leq} \sqrt{2}\,\mathrm{Hel}\Big(\mu_{d^{P^*}_{\pi,\rho},P^*}, \mu_{\mathcal{D},P}\Big)\Big(\sqrt{\kappa'}+1\Big)\sqrt{w_i^T \Sigma^{(i)}_{\mathcal{D},P} w_i} \\
&\leq \sqrt{2}\,\mathrm{Hel}\Big(\mu_{d^{P^*}_{\pi,\rho},P^*}, \mu_{\mathcal{D},P}\Big)\Big(\sqrt{\kappa'}+1\Big) M_\xi B_\xi =: \epsilon_2,
\end{aligned} \tag{58}
$$

where $\mu_{d,p}(s,a,x_{0:K}) := d(s,a)P(x_{0:K}|s,a)$ denotes the joint distribution, and Hel denotes the Hellinger distance. Inequality (i) follows by applying the Cauchy–Schwarz inequality to the Hellinger factorization Gibbs & Su (2002), whereas inequality (ii) follows from the stated assumptions.

Let

$$
\mathcal{N}_0 := \{i : J(\xi_i) \leq \tau + \eta\}, \qquad \mathcal{N}_1 := \{i : J(\xi_i) > \tau + \eta\}. \tag{59}
$$

Combining eq. (50) and eq. (57), we have that

$$
\begin{aligned}
&\alpha(1-\gamma)\sum_{i\in\mathcal{N}_0}\Big(V^{P_\xi}_\pi(\rho) - V^{P^*}_\pi(\rho)\Big) + \frac{\alpha}{2}\sum_{i\in\mathcal{N}_1}\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2 \\
&\leq \sum_{i=0}^{N-1}\mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}\big[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_i}(\cdot|s,a)) - \mathrm{KL}\big(P^*(\cdot|s,a)\|P_{\xi_{i+1}}(\cdot|s,a)\big)\big] \\
&\quad + \sum_{i\in\mathcal{N}_0}\left(\frac{\alpha^2 l}{2}\|w_i\|^2 + \alpha \mathrm{err}_i\right) + \sum_{i\in\mathcal{N}_1}\Big(\alpha \mathrm{err}^{(i)}_{\text{shift}}\Big) \\
&\leq \mathbb{E}_{(s,a)\sim d^{P^*}_{\pi,\rho}}[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))] + \frac{\alpha^2 l W^2 N}{2} + \alpha\left(\sum_{i\in\mathcal{N}_0}\mathrm{err}_i + \sum_{i\in\mathcal{N}_1}\mathrm{err}^{(i)}_{\text{shift}}\right).
\end{aligned} \tag{60}
$$

When $i \in \mathcal{N}_1$, from the assumptions we have that

$$
\Big\|g^{(i)}_{\mathcal{D},P}\Big\|^2 \geq \lambda\eta. \tag{61}
$$

Then we have that,

$$\alpha \sum_{i\in\mathcal{N}_0}\left(V_\pi^{P_\xi}(\rho)-V_\pi^{P^*}(\rho)\right)+\frac{\alpha\lambda\eta}{2(1-\gamma)}|\mathcal{N}_1|$$
$$\leq \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))]+\frac{\alpha^2 lW^2N}{2(1-\gamma)}+\frac{\alpha}{1-\gamma}\left(\sum_{i\in\mathcal{N}_0}\mathrm{err}_i+\sum_{i\in\mathcal{N}_1}\mathrm{err}_{\text{shift}}^{(i)}\right). \tag{62}$$

Lemma 4. *If*

$$\frac{\alpha\lambda\eta}{4(1-\gamma)}N\geq\frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))]$$
$$+\frac{\alpha^2 lW^2N}{2(1-\gamma)}+\frac{\alpha}{1-\gamma}\left(\sum_{i\in\mathcal{N}_0}\mathit{err}_i+\sum_{i\in\mathcal{N}_1}\mathit{err}_{\mathit{shift}}^{(i)}\right), \tag{63}$$

then we have the following holds

1. $|\mathcal{N}_0|\neq 0$
2. *One of the following two statements must hold*
   - *(a)* $|\mathcal{N}_0|\geq\frac{N}{2}$
   - *(b)* $\sum_{i\in\mathcal{N}_0}\left(V_\pi^{P_{\xi_i}}(\rho)-V_\pi^{P^*}(\rho)\right)\leq 0$

Proof. If $|\mathcal{N}_0|=0$, then we have

$$\frac{\alpha\lambda\eta}{2(1-\gamma)}N$$
$$\leq\frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))]$$
$$+\frac{\alpha^2 lW^2N}{2(1-\gamma)}+\frac{\alpha}{1-\gamma}\left(\sum_{i\in\mathcal{N}_0}\mathrm{err}_i+\sum_{i\in\mathcal{N}_1}\mathrm{err}_{\text{shift}}\right), \tag{64}$$

which contradict with eq. (63).

If $\sum_{i\in\mathcal{N}_0}\left(V_\pi^{P_{\xi_i}}(\rho)-V_\pi^{P^*}(\rho)\right)\leq 0$, then (b) in item 2 holds. If $\sum_{i\in\mathcal{N}_0}\left(V_\pi^{P_{\xi_i}}(\rho)-V_\pi^{P^*}(\rho)\right)\geq 0$, then we have that

$$\frac{\alpha\lambda\eta}{2(1-\gamma)}|\mathcal{N}_1|$$
$$\leq\frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))]$$
$$+\frac{\alpha^2 lW^2N}{2(1-\gamma)}+\frac{\alpha}{1-\gamma}\left(\sum_{i\in\mathcal{N}_0}\mathrm{err}_i+\sum_{i\in\mathcal{N}_1}\mathrm{err}_{\text{shift}}\right). \tag{65}$$

If $|\mathcal{N}_0|\leq N/2$, then we have that $|\mathcal{N}_1|\geq N/2$. Then eq. (62) implies that

$$\frac{\alpha\lambda\eta}{2(1-\gamma)}\frac{N}{2}\leq\frac{\alpha\lambda\eta}{2(1-\gamma)}|\mathcal{N}_1|$$
$$\leq\frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\mathrm{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))]+\frac{\alpha^2 lW^2N}{2(1-\gamma)}+\frac{\alpha}{1-\gamma}\left(\sum_{i\in\mathcal{N}_0}\mathrm{err}_i+\sum_{i\in\mathcal{N}_1}\mathrm{err}_{\text{shift}}\right), \tag{66}$$

which contradict with eq. (63). □

Define $\epsilon_{\text{max}} = \max\{\epsilon_1, \epsilon_2\}$. If we choose

$$\alpha = \sqrt{\frac{2B_{KL}}{lW^2N}}, \quad N \geq \frac{2B_{\text{KL}}l}{W^2} \tag{67}$$

and

$$\eta = 4\sqrt{\frac{2B_{\text{KL}}lW^2}{\lambda^2 N}} + \frac{4\epsilon_{\text{max}}}{\lambda}, \tag{68}$$

We have that

$$\mathbb{E}\Big[V_\pi^{P_{\text{out}}}(\rho) - V_\pi^{P^*}(\rho)\Big] \leq \sqrt{\frac{8B_{\text{KL}}lW^2}{(1-\gamma)^2 N}} + \frac{2\epsilon_{\text{max}}}{(1-\gamma)}, \tag{69}$$

and

$$\mathbb{E}[J(\xi_{\text{out}})] - \tau \leq 4\sqrt{\frac{2B_{\text{KL}}lW^2}{\lambda^2 N}} + \frac{4\epsilon_{\text{max}}}{\lambda}. \tag{70}$$

Proof. We first prove the convergence of $V_\pi^{P_{\text{out}}}(\rho)$. With our selection of η, eq. (63) is satisfied. Then from eq. (62) we have that

$$\begin{aligned}
&\alpha \sum_{i\in\mathcal{N}_0} \Big(V_\pi^{P_\xi}(\rho) - V_\pi^{P^*}(\rho)\Big) \\
&\leq \alpha \sum_{i\in\mathcal{N}_0} \Big(V_\pi^{P_\xi}(\rho) - V_\pi^{P^*}(\rho)\Big) + \frac{\alpha\lambda\eta}{2(1-\gamma)}|\mathcal{N}_1| \\
&\leq \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\pi,\rho}^{P^*}}[\text{KL}(P^*(\cdot|s,a)\|P_{\xi_0}(\cdot|s,a))] + \frac{\alpha^2 lW^2N}{2(1-\gamma)} + \frac{\alpha}{1-\gamma}\left(\sum_{i\in\mathcal{N}_0}\text{err}_i + \sum_{i\in\mathcal{N}_1}\text{err}_{\text{shift}}\right) \\
&\leq \frac{B_{KL}}{1-\gamma} + \frac{\alpha^2 lW^2N}{2(1-\gamma)} + \frac{\alpha N\epsilon_{\text{max}}}{1-\gamma}.
\end{aligned} \tag{71}$$

Then we have that whether $\mathbb{E}\big[V_\pi^{P_{\text{out}}}(\rho) - V_\pi^{P^*}(\rho)\big] \leq 0$ or

$$\begin{aligned}
\mathbb{E}\Big[V_\pi^{P_{\text{out}}}(\rho) - V_\pi^{P^*}(\rho)\Big] &= \frac{1}{|\mathcal{N}_0|}\sum_{i\in\mathcal{N}_0}\Big(V_\pi^{P_\xi}(\rho) - V_\pi^{P^*}(\rho)\Big) \\
&\leq \frac{4B_{\text{KL}}}{\alpha(1-\gamma)N} + \frac{2\epsilon_{\text{max}}}{(1-\gamma)} \\
&= \sqrt{\frac{8B_{\text{KL}}lW^2}{(1-\gamma)^2N}} + \frac{2\epsilon_{\text{max}}}{(1-\gamma)}.
\end{aligned} \tag{72}$$

Then we proceed to bound the constraints violation,

$$\begin{aligned}
\mathbb{E}[J(\xi_{\text{out}})] - \tau &= \frac{1}{|\mathcal{N}_0|}\sum_{i\in\mathcal{N}_0} J(\xi_i) - \tau \\
&\leq \eta = 4\sqrt{\frac{2B_{\text{KL}}lW^2}{\lambda^2 N}} + \frac{4\epsilon_{\text{max}}}{\lambda}.
\end{aligned} \tag{73}$$

□

A.3 Convergence of the Outer Loop.

We first present the assumptions we make to prove the convergence of the outer-loop.

Assumption 3. *1.* $\log \pi_\theta(a|s)$ *is* l_θ*-smooth.*

2. $\|\nabla_\theta \log \pi_\theta(a|s)\| \leq B_\Theta$

Let $\Phi(\theta,\xi) = V_{\pi_\theta}^{P_\xi}(\rho)$, and $\Psi(\theta) = \min_{\xi\in\Xi} V_{\pi_\theta}^{P_\xi}(\rho)$. First, we have the following lemmas.

Lemma 5. *$\Phi(\theta,\xi)$ is smooth about θ:*

$$\|\nabla_\theta\Phi(\theta_1,\xi) - \nabla_\theta\Phi(\theta_2,\xi)\| \le L_\Theta\|\theta_1-\theta_2\|, \tag{74}$$

where

$$L_\Theta = \frac{l_\theta}{(1-\gamma)^2} + \frac{(1+\gamma)B_\Theta^2}{(1-\gamma)^3}. \tag{75}$$

And we have that,

$$|\Phi(\theta_1,\xi) - \Phi(\theta_2,\xi)| \le \frac{B_\Theta}{(1-\gamma)^2}\|\theta_1-\theta_2\|. \tag{76}$$

Lemma 6. *$\|\nabla_\theta\Phi(\theta,\xi)\|$ is bounded:*

$$\|\nabla_\theta\Phi(\theta,\xi)\| \le L_\pi = \frac{B_\Theta}{(1-\gamma)^2}. \tag{77}$$

Lemma 7. *The robust objective $\Psi(\theta)$ is $\frac{B_\Theta}{(1-\gamma)^2}$-Lipschitz continuous, i.e.,*

$$|\Psi(\theta_1) - \Psi(\theta_2)| \le \frac{B_\Theta}{(1-\gamma)^2}\|\theta_1-\theta_2\|. \tag{78}$$

Proof. Let $\xi_i \in \arg\min_{\xi\in\Xi}\Phi(\theta_i,\xi)$ for $i=1,2$. Then

$$\Psi(\theta_1) - \Psi(\theta_2) = \Phi(\theta_1,\xi_1) - \Phi(\theta_2,\xi_2) \le \Phi(\theta_1,\xi_2) - \Phi(\theta_2,\xi_2) \le \frac{B_\Theta}{(1-\gamma)^2}\|\theta_1-\theta_2\|. \tag{79}$$

Exchanging θ_1 and θ_2 gives the reverse inequality, and hence the result. □

Since Ψ is Lipschitz continuous and $\Phi(\theta,\xi)$ is L_Θ-smooth with respect to θ for every $\xi\in\Xi$, Lemma 3 of (Thekumparampil et al., 2019) implies that $\Psi(\theta) = \min_{\xi\in\Xi}\Phi(\theta,\xi)$ is L_Θ-weakly concave.

We have the Moreau envelope of Ψ defined as:

$$\Psi_{\frac{1}{2L_\Theta}}(\theta) = \max_{\theta'}\Big\{\Psi(\theta') - L_\Theta\|\theta'-\theta\|^2\Big\}, \tag{80}$$

and let

$$\tilde\theta_t = \arg\max_{\theta}\Big\{\Psi(\theta) - L_\Theta\|\theta-\theta_t\|^2\Big\}. \tag{81}$$

Then we have that

$$\begin{aligned}
&\Psi_{\frac{1}{2L_\Theta}}(\theta_{t+1})\\
&= \max_\theta\Big\{\Psi(\theta) - L_\Theta\|\theta-\theta_{t+1}\|^2\Big\}\\
&\ge \Psi(\tilde\theta_t) - L_\Theta\Big\|\theta_{t+1}-\tilde\theta_t\Big\|^2\\
&= \Psi(\tilde\theta_t) - L_\Theta\Big\|\theta_t+\beta\nabla_\theta\Phi(\theta_t,\xi_t)-\tilde\theta_t\Big\|^2\\
&= \Psi(\tilde\theta_t) - L_\Theta\Big\|\theta_t-\tilde\theta_t\Big\|^2 - L_\Theta\beta^2\|\nabla_\theta\Phi(\theta_t,\xi_t)\|^2 - 2\beta L_\Theta\langle\theta_t-\tilde\theta_t,\nabla_\theta\Phi(\theta_t,\xi_t)\rangle\\
&\ge \Psi_{\frac{1}{2L_\Theta}}(\theta_t) - L_\Theta\beta^2\|\nabla_\theta\Phi(\theta_t,\xi_t)\|^2 + 2\beta L_\Theta\left(\Psi(\tilde\theta_t) - \Psi(\theta_t) - \epsilon_t - \frac{L_\Theta}{2}\Big\|\tilde\theta_t-\theta_t\Big\|^2\right),
\end{aligned} \tag{82}$$

where the last inequality follows from

$$\begin{aligned}
\Psi(\tilde\theta_t) \le \Phi(\tilde\theta_t,\xi_t) &\le \Phi(\theta_t,\xi_t) + \langle\tilde\theta_t-\theta_t,\nabla_\theta\Phi(\theta_t,\xi_t)\rangle + \frac{L_\Theta}{2}\Big\|\tilde\theta_t-\theta_t\Big\|^2\\
&\le \Psi(\theta_t) + \epsilon_t + \langle\tilde\theta_t-\theta_t,\nabla_\theta\Phi(\theta_t,\xi_t)\rangle + \frac{L_\Theta}{2}\Big\|\tilde\theta_t-\theta_t\Big\|^2.
\end{aligned} \tag{83}$$

By summing eq. (82) up over t, we have that

$$\Psi_{\frac{1}{2L_\Theta}}(\theta_T) \geq \Psi_{\frac{1}{2L_\Theta}}(\theta_0) + 2\beta L_\Theta \sum_{t=0}^{T-1}\left(\Psi(\tilde{\theta}_t) - \Psi(\theta_t) - \epsilon_t - \frac{L_\Theta}{2}\left\|\tilde{\theta}_t - \theta_t\right\|^2\right) - TL_\Theta\beta^2 L_\pi^2. \tag{84}$$

Rearrange, we have that

$$\sum_{t=0}^{T-1}\left(\Psi(\tilde{\theta}_t) - \Psi(\theta_t) - \frac{L_\Theta}{2}\left\|\tilde{\theta}_t - \theta_t\right\|^2\right) \leq \frac{\Psi_{\frac{1}{2L_\Theta}}(\theta_T) - \Psi_{\frac{1}{2L_\Theta}}(\theta_0)}{2\beta L_\Theta} + \frac{T\beta}{2}L_\pi^2 + \sum_{t=0}^{T-1}\epsilon_t. \tag{85}$$

We can bound the term in the summation as follows,

$$\begin{aligned}
&\Psi(\tilde{\theta}_t) - \Psi(\theta_t) - \frac{L_\Theta}{2}\left\|\tilde{\theta}_t - \theta_t\right\|^2 \\
&= \Psi(\tilde{\theta}_t) - L_\Theta\left\|\tilde{\theta}_t - \theta_t\right\|^2 - \Psi(\theta_t) + L_\Theta\left\|\tilde{\theta}_t - \theta_t\right\|^2 - \frac{L_\Theta}{2}\left\|\tilde{\theta}_t - \theta_t\right\|^2 \\
&\overset{(i)}{\geq} L_\Theta\left\|\tilde{\theta}_t - \theta_t\right\|^2 = \frac{1}{4L_\Theta}\left\|\nabla_\theta \Psi_{\frac{1}{2L_\Theta}}(\theta_t)\right\|^2,
\end{aligned} \tag{86}$$

where (i) follows from the strong concavity of $\Psi(\theta) - L_\Theta\|\theta - \theta_t\|^2$, and $\tilde{\theta}_t$ is its optimal point. Also, we have that

$$\begin{aligned}
&\Psi_{\frac{1}{2L_\Theta}}(\theta_1) - \Psi_{\frac{1}{2L_\Theta}}(\theta_2) \\
&= \max_\theta\left\{\Psi(\theta) - L_\Theta\|\theta - \theta_1\|^2\right\} - \max_\theta\left\{\Psi(\theta) - L_\Theta\|\theta - \theta_2\|^2\right\} \\
&\leq \sup_u\{\Psi(\theta_1 + u) - \Psi(\theta_2 + u)\} \\
&\leq L_\pi\|\theta_1 - \theta_2\| \leq \frac{1}{1-\gamma},
\end{aligned} \tag{87}$$

where the last inequality follows from the L_π-Lipschitz continuity of Ψ. Combining the above results, we have that,

$$\frac{1}{4L_\Theta}\sum_{t=0}^{T-1}\left\|\nabla_\theta \Psi_{\frac{1}{2L_\Theta}}(\theta_t)\right\|^2 \leq \frac{L_\pi\|\theta_T - \theta_0\|}{2\beta L_\Theta} + \frac{T\beta}{2}L_\pi^2 + \sum_{t=0}^{T-1}\epsilon_t. \tag{88}$$

Choosing

$$\beta = \sqrt{\frac{\|\theta_T - \theta_0\|}{L_\Theta L_\pi T}}, \tag{89}$$

we obtain

$$\frac{1}{T}\sum_{t=0}^{T-1}\left\|\nabla_\theta \Psi_{\frac{1}{2L_\Theta}}(\theta_t)\right\|^2 \leq 4\sqrt{\frac{L_\pi^3 L_\Theta\|\theta_T - \theta_0\|}{T}} + \frac{4L_\Theta}{T}\sum_{t=0}^{T-1}\epsilon_t. \tag{90}$$

Thus, the stationarity measure converges at rate $\mathcal{O}(T^{-1/2})$ up to the policy-evaluation error. However, this only provides a stationary-point guarantee rather than a global convergence guarantee. The latter is lost due to policy function approximation; under direct parameterization, one can derive global convergence with similar residual error terms.

B Experiment Details

B.1 Codebase

Since RAMBO (Rigter et al., 2022) is the closest algorithm to ours, we build our algorithm on the OfflineRLKit codebase (Yihao Sun, 2023), replacing its Gaussian dynamics model with more expressive generative world models, including diffusion models and conditional VAEs.

B.2 World Model Training

Following previous works Yu et al. (2020); Sun et al. (2023); Rigter et al. (2022), we adapt the essemble-elite structure of the world model. To implement our distributional robust RL algorithm we seperate the transition model and the reward model. The detailed training loss for diffusion world model and conditional VAE are shown in the main text, which is for the transition model. For the reward model we train with a seperate MSE loss. The detailed hyperparameter of the world models are shown in Table 3.

Table 3: Hyperparameters used for ensemble-CVAE world-model and robust policy training.

	Hyperparameter	**Value**
CVAE world model	Ensemble size / elite size	$7/5$
	Prior network	$(d_s + d_a, 200, 200, 2d_z)$
	Posterior network	$(2d_s + d_a, 200, 200, 2d_z)$
	Transition decoder	$(d_s + d_a + d_z, 200, 200, 2d_s)$
	Reward network	$(d_s + d_a, 200, 200, 1)$
	Latent dimension (d_z)	8
	Learning rate	1×10^{-4}
	Batch size	256
	Maximum pretraining epochs	1000
Diffusion world model	Ensemble size / elite size	$7/5$
	Input condition	$(d_s + d_a)$
	Denoising network	$(d_s + d_s + d_a, 256)$
	Denoising residual MLP blocks	2
	Time embedding	sinusoidal embedding, dimension 64
	Reward network	$(d_s + d_a, 256, 256, 1)$
	Hidden dimension	256
	Number of residual blocks	2
	Diffusion training steps	100
	DDIM sampling steps	5
	Learning rate	3×10^{-4}
	Batch size	2048
	Maximum pretraining epochs	1000

B.3 Robust Policy Training

For policy optimization, we follow the standard Soft Actor-Critic (SAC) implementation provided in OfflineRLKit. We use a 50% real-data ratio, such that each training batch contains equal proportions of transitions generated by the world model and transitions sampled from the offline dataset. We also employ automatic entropy-temperature tuning. Our primary modification is the addition of a robust policy evaluation step. Detailed hyperparameter settings are reported in Table 4. To improve computational efficiency, our implementation uses standard gradient updates rather than natural gradient updates during adversarial world-model optimization.

B.4 Hyperparameter Tuning

There are two main hyperparameters to tune in GIRO: the rollout length and the budget scale. We show how we tune them below.

Rollout Length. We sample short-horizon rollouts from the world models, following prior work Sun et al. (2023); Rigter et al. (2022), and tune the rollout length over $[1, 5]$.

Table 4: Hyperparameters for adversarial world-model updates and policy optimization.

Component	Hyperparameter	CVAE	Diffusion
Robust model update	Adversarial learning rate	1×10^{-7}	1×10^{-5}
	Model-fit learning rate	1×10^{-6}	1×10^{-4}
	Inner updates per policy step	1	
	Model-fit batch size	2048	256
Policy optimization	Actor architecture	$(d_s, 256, 256, 2d_a)$	
	Critic architecture	$2 \times (d_s + d_a, 256, 256, 1)$	
	Actor learning rate	1×10^{-4}	
	Critic learning rate	3×10^{-4}	
	Entropy learning rate	1×10^{-4}	
	Discount factor γ	0.99	
	Soft-update parameter τ	5×10^{-3}	
	Policy batch size	256	
	Real-data ratio	0.5	

Budget Scale. We set the uncertainty-set budget relative to the training loss of the pretrained world model, and tune the scale over $[0.95, 1.05]$.

Table 5: Rollout lengths and budget scales for D4RL tasks.

World Model	Task Name	Rollout Length	Budget Scale
CVAE Models	halfcheetah-medium	1	0.95
	hopper-medium	5	0.95
	walker2d-medium	5	0.95
	halfcheetah-medium-replay	5	1.00
	hopper-medium-replay	5	1.00
	walker2d-medium-replay	5	1.00
	halfcheetah-medium-expert	3	1.00
	hopper-medium-expert	4	1.00
	walker2d-medium-expert	5	0.98
Diffusion Models	halfcheetah-medium	3	1.02
	hopper-medium	5	0.95
	walker2d-medium	5	0.95
	halfcheetah-medium-replay	5	1.05
	hopper-medium-replay	3	1.05
	walker2d-medium-replay	3	1.02
	halfcheetah-medium-expert	3	1.05
	hopper-medium-expert	5	0.98
	walker2d-medium-expert	5	0.95

C More Experiment Results

C.1 Detailed Robustness results

Table 6 presents the nominal and disturbed-environment evaluation results. Each result is averaged over three random seeds, with 20 evaluation episodes conducted for every combination of task, policy seed, and disturbance condition. We report the mean and standard deviation across seeds for

Table 6: Normalized D4RL scores (mean ± population standard deviation across three seeds). The lowest disturbed mean in each policy row is bolded. Dist. Variance is the population variance of the eight disturbance means.

Dataset	Policy	Nominal	State-IID	State-Static	Action-IID	Action-Static	Mass ×0.95	Mass ×1.05	Friction ×0.95	Friction ×1.05	Dist. Variance
HC-M	Model-based SAC	69.2 ± 1.5	**66.2 ± 1.8**	67.5 ± 1.3	68.3 ± 1.8	67.6 ± 2.5	70.4 ± 1.7	68.2 ± 1.5	68.7 ± 2.0	69.6 ± 1.7	1.46
	CQL	49.2 ± 0.3	48.7 ± 0.1	48.5 ± 0.1	48.5 ± 0.2	48.4 ± 0.0	51.0 ± 0.1	**47.7 ± 0.2**	49.5 ± 0.1	49.5 ± 0.1	0.91
	MOPO	69.3 ± 6.9	71.7 ± 5.3	73.2 ± 4.1	72.9 ± 2.1	70.7 ± 4.6	73.2 ± 0.6	**69.9 ± 5.6**	70.9 ± 5.8	76.2 ± 1.7	3.47
	IQL	47.5 ± 0.3	47.0 ± 0.3	46.7 ± 0.1	46.6 ± 0.3	46.5 ± 0.4	48.3 ± 0.6	**46.3 ± 0.4**	47.3 ± 0.5	47.5 ± 0.5	0.37
	RAMBO	78.4 ± 0.7	76.6 ± 0.6	76.9 ± 0.6	76.7 ± 2.0	**76.2 ± 0.8**	78.7 ± 0.4	77.4 ± 0.4	76.4 ± 1.9	78.3 ± 0.2	0.73
	Mobile	73.4 ± 2.9	**70.7 ± 5.3**	74.2 ± 2.0	71.9 ± 3.8	73.7 ± 2.4	75.7 ± 2.2	73.5 ± 1.6	75.5 ± 0.5	75.1 ± 2.3	2.69
	GIRO (VAE)	66.2 ± 2.2	65.0 ± 1.6	**64.6 ± 1.3**	65.7 ± 2.0	65.0 ± 1.7	66.7 ± 2.3	65.6 ± 2.2	66.1 ± 2.2	66.5 ± 2.2	0.51
	GIRO (Diffusion)	70.7 ± 0.2	**68.3 ± 0.3**	68.6 ± 0.6	69.7 ± 0.2	69.8 ± 0.4	71.2 ± 0.3	69.8 ± 0.4	70.4 ± 0.2	71.0 ± 0.1	0.93
HC-MR	Model-based SAC	58.7 ± 11.2	**32.8 ± 12.1**	50.9 ± 4.6	52.4 ± 14.3	55.3 ± 9.6	53.6 ± 17.2	61.9 ± 5.3	58.5 ± 11.3	57.0 ± 9.4	67.95
	CQL	47.2 ± 0.4	42.3 ± 0.1	**42.0 ± 0.4**	46.3 ± 0.4	45.9 ± 0.3	47.8 ± 0.6	46.2 ± 0.3	47.3 ± 0.4	47.4 ± 0.4	4.42
	MOPO	73.1 ± 3.6	**55.5 ± 1.2**	59.4 ± 1.4	71.7 ± 3.4	72.1 ± 3.3	74.3 ± 3.4	71.9 ± 3.8	73.1 ± 3.9	72.7 ± 3.3	45.01
	IQL	43.6 ± 0.5	40.3 ± 0.8	**39.5 ± 2.1**	43.0 ± 1.0	43.1 ± 0.8	43.3 ± 0.5	44.1 ± 0.1	44.7 ± 0.4	44.0 ± 0.8	3.02
	RAMBO	70.4 ± 1.9	**60.0 ± 0.4**	62.8 ± 0.6	69.4 ± 1.3	69.1 ± 0.8	70.3 ± 2.5	69.6 ± 1.2	70.4 ± 1.6	69.5 ± 1.8	13.57
	Mobile	72.7 ± 3.4	**56.3 ± 1.4**	58.7 ± 1.6	71.0 ± 3.7	71.1 ± 3.4	73.5 ± 2.8	70.4 ± 3.5	72.5 ± 3.0	71.7 ± 2.9	38.99
	GIRO (VAE)	56.9 ± 1.3	**47.1 ± 1.1**	49.2 ± 1.3	56.8 ± 1.2	57.1 ± 1.6	57.6 ± 1.2	56.5 ± 1.2	56.7 ± 1.3	57.1 ± 1.1	14.94
	GIRO (Diffusion)	64.4 ± 0.9	**55.5 ± 0.2**	56.0 ± 1.1	64.6 ± 1.2	64.7 ± 1.2	65.9 ± 1.5	63.8 ± 0.7	65.2 ± 1.5	65.4 ± 1.4	16.16
HC-ME	Model-based SAC	86.3 ± 1.7	**62.9 ± 7.0**	73.4 ± 5.4	78.8 ± 3.7	84.6 ± 5.7	85.3 ± 1.7	92.9 ± 4.1	92.3 ± 3.0	89.8 ± 5.9	94.10
	CQL	91.8 ± 1.1	**68.0 ± 4.4**	78.8 ± 0.2	84.8 ± 3.5	83.7 ± 6.5	90.1 ± 4.5	89.7 ± 2.7	90.3 ± 2.1	91.8 ± 1.4	56.43
	MOPO	83.3 ± 8.3	**45.8 ± 5.1**	74.8 ± 6.5	65.9 ± 6.8	67.7 ± 13.1	72.5 ± 14.5	71.6 ± 14.1	76.4 ± 12.8	73.2 ± 7.9	83.69
	IQL	88.3 ± 2.1	**65.9 ± 1.4**	73.3 ± 2.2	75.6 ± 1.3	80.7 ± 1.2	88.8 ± 1.6	87.9 ± 1.2	90.9 ± 1.4	91.2 ± 1.2	77.64
	RAMBO	97.1 ± 3.5	**68.8 ± 6.9**	83.1 ± 8.3	90.9 ± 6.8	95.0 ± 3.1	95.4 ± 6.1	97.0 ± 7.1	96.8 ± 6.5	96.0 ± 3.9	85.40
	Mobile	100.5 ± 1.6	**62.8 ± 6.1**	81.2 ± 3.9	95.8 ± 1.8	95.5 ± 4.0	99.7 ± 0.6	101.6 ± 1.3	100.9 ± 1.7	98.4 ± 1.3	158.32
	GIRO (VAE)	86.4 ± 5.4	**68.8 ± 4.4**	73.2 ± 3.9	81.8 ± 0.7	84.3 ± 3.1	73.7 ± 8.7	89.9 ± 1.8	89.6 ± 2.7	88.4 ± 5.0	60.34
	GIRO (Diffusion)	96.2 ± 1.6	**70.8 ± 3.6**	87.5 ± 0.3	88.4 ± 2.2	91.3 ± 5.0	96.7 ± 4.0	93.0 ± 7.6	95.8 ± 1.2	96.8 ± 3.3	64.11
Hopper-M	Model-based SAC	66.3 ± 46.3	43.9 ± 31.0	61.1 ± 42.6	**40.4 ± 29.5**	51.2 ± 36.6	68.5 ± 47.9	57.8 ± 40.4	55.7 ± 41.3	56.9 ± 40.2	72.51
	CQL	58.6 ± 1.6	58.9 ± 1.2	62.0 ± 4.1	**48.3 ± 0.9**	52.9 ± 1.0	68.4 ± 3.9	55.9 ± 2.0	59.1 ± 2.6	49.3 ± 1.2	39.44
	MOPO	63.0 ± 37.1	41.2 ± 17.9	50.1 ± 24.9	**30.9 ± 11.7**	51.2 ± 26.6	71.9 ± 43.1	39.4 ± 21.0	46.0 ± 24.9	64.3 ± 38.1	156.55
	IQL	70.5 ± 1.6	69.8 ± 6.4	68.3 ± 3.6	**57.1 ± 3.8**	60.6 ± 5.2	84.8 ± 5.2	60.5 ± 6.6	72.4 ± 3.0	57.3 ± 4.3	78.60
	RAMBO	69.6 ± 38.3	**53.2 ± 17.7**	68.7 ± 18.8	66.7 ± 2.9	69.4 ± 18.5	75.8 ± 42.1	70.0 ± 27.9	66.7 ± 32.9	70.2 ± 38.2	36.56
	Mobile	103.6 ± 0.6	93.4 ± 12.6	92.2 ± 10.2	86.6 ± 16.5	88.6 ± 6.4	104.1 ± 0.8	101.2 ± 4.2	**82.4 ± 29.0**	105.8 ± 0.6	64.08
	GIRO (VAE)	99.7 ± 1.4	100.3 ± 1.4	99.5 ± 1.4	100.5 ± 2.0	99.7 ± 1.8	99.5 ± 1.4	99.9 ± 1.6	100.3 ± 2.2	**97.9 ± 1.4**	0.59
	GIRO (Diffusion)	106.1 ± 0.6	100.0 ± 5.9	**90.2 ± 9.1**	104.0 ± 0.7	96.0 ± 1.3	107.1 ± 0.4	105.2 ± 0.7	104.4 ± 0.8	108.1 ± 0.2	32.79
Hopper-MR	Model-based SAC	37.0 ± 1.7	45.7 ± 2.3	58.0 ± 2.2	55.7 ± 1.5	44.0 ± 1.6	37.9 ± 2.1	44.4 ± 5.9	47.8 ± 9.3	**35.9 ± 2.5**	51.85
	CQL	96.9 ± 4.9	87.1 ± 4.8	75.6 ± 3.6	82.2 ± 6.3	83.8 ± 8.2	98.3 ± 1.1	91.9 ± 6.6	91.7 ± 3.1	**70.0 ± 28.1**	74.52
	MOPO	88.5 ± 22.1	87.4 ± 12.0	92.5 ± 7.3	**83.5 ± 19.2**	86.8 ± 10.6	94.8 ± 13.7	102.9 ± 1.4	103.8 ± 0.7	101.2 ± 6.5	54.28
	IQL	94.3 ± 8.7	86.1 ± 9.5	**65.9 ± 9.7**	77.9 ± 15.2	81.0 ± 6.2	98.0 ± 3.1	83.6 ± 17.3	94.3 ± 7.7	78.3 ± 14.0	88.00
	RAMBO	89.5 ± 17.6	79.1 ± 16.9	85.4 ± 10.8	**73.4 ± 20.8**	77.7 ± 16.2	90.7 ± 16.8	83.9 ± 25.3	84.5 ± 23.7	90.9 ± 17.3	33.06
	Mobile	104.8 ± 1.4	96.5 ± 6.6	**80.1 ± 13.5**	81.8 ± 27.4	82.5 ± 9.9	105.2 ± 1.2	96.8 ± 11.3	95.2 ± 12.2	104.4 ± 3.0	88.87
	GIRO (VAE)	98.4 ± 2.0	99.8 ± 1.7	97.5 ± 1.6	98.6 ± 2.0	**95.9 ± 0.8**	98.4 ± 2.0	98.5 ± 2.0	97.9 ± 2.1	99.9 ± 1.8	1.38
	GIRO (Diffusion)	104.8 ± 1.3	100.8 ± 2.3	**91.1 ± 7.5**	99.9 ± 2.1	91.2 ± 0.3	105.3 ± 1.7	103.8 ± 1.0	103.5 ± 1.2	105.9 ± 1.3	30.83
Hopper-ME	Model-based SAC	90.0 ± 24.6	69.6 ± 8.8	84.7 ± 11.4	65.6 ± 15.3	75.7 ± 11.9	79.6 ± 22.1	**63.4 ± 14.4**	76.6 ± 23.9	82.4 ± 18.6	52.85
	CQL	108.5 ± 1.7	90.9 ± 6.4	**75.8 ± 2.0**	90.3 ± 4.2	86.9 ± 1.8	106.8 ± 3.8	100.5 ± 5.9	105.6 ± 4.1	102.2 ± 7.5	101.12
	MOPO	75.0 ± 51.0	73.6 ± 39.3	69.8 ± 24.5	68.7 ± 45.8	**63.9 ± 42.1**	78.7 ± 47.3	74.3 ± 50.6	74.6 ± 49.7	86.8 ± 36.0	42.13
	IQL	103.7 ± 9.6	67.3 ± 7.9	**56.8 ± 10.9**	58.2 ± 3.7	65.8 ± 12.4	87.7 ± 18.8	84.2 ± 11.0	89.7 ± 11.3	95.6 ± 16.1	205.16
	RAMBO	102.7 ± 3.1	101.3 ± 2.1	98.1 ± 2.5	96.8 ± 7.2	**96.2 ± 4.9**	102.9 ± 3.0	102.1 ± 3.0	101.3 ± 2.0	104.4 ± 4.2	7.90
	Mobile	112.4 ± 0.2	106.1 ± 8.0	**78.7 ± 4.4**	106.6 ± 4.2	95.2 ± 2.2	111.7 ± 1.6	111.7 ± 0.4	110.6 ± 0.4	112.8 ± 2.0	120.68
	GIRO (VAE)	91.5 ± 8.4	87.0 ± 3.3	87.7 ± 4.5	81.6 ± 13.7	**78.3 ± 4.6**	97.4 ± 10.0	82.4 ± 14.4	84.7 ± 15.7	87.4 ± 13.2	28.64
	GIRO (Diffusion)	75.8 ± 3.1	66.9 ± 0.8	77.2 ± 6.5	58.6 ± 2.2	69.4 ± 4.5	105.9 ± 2.8	**55.0 ± 1.0**	80.6 ± 3.6	59.4 ± 0.8	238.75
Walker2d-M	Model-based SAC	87.0 ± 18.1	76.4 ± 8.9	78.3 ± 15.8	73.9 ± 9.7	**68.8 ± 9.8**	85.9 ± 20.9	71.4 ± 10.7	73.8 ± 33.5	76.9 ± 21.4	23.20
	CQL	83.8 ± 0.5	83.1 ± 0.8	83.3 ± 0.6	84.1 ± 0.7	**82.0 ± 1.1**	83.3 ± 0.4	84.7 ± 0.9	83.1 ± 0.4	83.8 ± 0.8	0.56
	MOPO	85.0 ± 2.9	85.6 ± 2.0	86.3 ± 2.2	85.9 ± 3.2	83.3 ± 1.7	**82.5 ± 2.3**	87.7 ± 2.4	84.7 ± 2.9	87.7 ± 2.8	3.17
	IQL	80.2 ± 2.2	83.1 ± 1.6	**74.5 ± 8.5**	79.8 ± 2.0	80.8 ± 1.2	81.8 ± 1.1	82.1 ± 0.9	80.8 ± 3.4	80.3 ± 0.9	5.84
	RAMBO	86.7 ± 0.7	86.8 ± 0.9	87.1 ± 0.1	86.7 ± 0.6	86.5 ± 0.5	**86.3 ± 1.2**	87.2 ± 0.5	86.4 ± 1.7	86.7 ± 0.4	0.08
	Mobile	87.8 ± 2.6	87.5 ± 3.4	84.4 ± 8.5	82.8 ± 6.8	**75.5 ± 8.0**	88.7 ± 3.0	80.9 ± 12.7	89.8 ± 2.3	89.1 ± 3.1	21.67
	GIRO (VAE)	84.1 ± 0.8	84.6 ± 1.0	84.7 ± 1.2	84.4 ± 0.9	84.6 ± 0.8	83.9 ± 1.4	85.1 ± 0.9	**83.9 ± 0.6**	84.2 ± 0.7	0.15
	GIRO (Diffusion)	89.7 ± 17.7	82.4 ± 19.6	89.9 ± 11.7	**75.9 ± 19.3**	76.9 ± 9.9	85.5 ± 20.7	87.4 ± 10.4	85.7 ± 15.9	84.6 ± 22.3	21.09
Walker2d-MR	Model-based SAC	83.2 ± 3.2	81.8 ± 2.3	81.2 ± 3.6	**79.2 ± 5.6**	81.8 ± 6.6	85.8 ± 5.3	85.5 ± 3.7	81.2 ± 6.4	84.3 ± 5.2	4.84
	CQL	75.8 ± 15.9	83.6 ± 4.1	**63.4 ± 18.7**	68.6 ± 18.8	70.6 ± 17.7	80.2 ± 7.9	66.7 ± 22.3	77.9 ± 5.8	74.0 ± 18.8	43.16
	MOPO	81.1 ± 10.2	82.9 ± 7.9	**73.5 ± 16.0**	80.5 ± 10.7	82.6 ± 8.4	81.3 ± 9.4	75.4 ± 15.4	81.0 ± 9.2	84.5 ± 9.3	12.76
	IQL	83.1 ± 6.5	69.9 ± 9.6	**65.7 ± 6.9**	70.2 ± 8.3	73.1 ± 5.0	73.8 ± 9.4	73.8 ± 7.2	76.2 ± 12.6	77.0 ± 8.2	11.97
	RAMBO	70.5 ± 9.6	76.8 ± 2.4	77.5 ± 2.5	72.8 ± 6.8	72.8 ± 7.6	74.2 ± 6.0	**72.3 ± 8.4**	74.1 ± 6.5	73.2 ± 6.3	3.21
	Mobile	90.0 ± 1.4	90.2 ± 1.2	**87.8 ± 0.7**	89.6 ± 1.2	89.1 ± 1.1	89.5 ± 2.0	89.7 ± 0.7	89.9 ± 1.6	89.2 ± 1.4	0.46
	GIRO (VAE)	83.0 ± 4.6	84.4 ± 5.0	86.2 ± 3.9	84.0 ± 3.2	79.7 ± 7.8	85.6 ± 1.6	82.8 ± 6.4	**79.1 ± 8.6**	89.3 ± 0.8	9.88
	GIRO (Diffusion)	93.0 ± 3.3	90.2 ± 6.9	93.2 ± 1.5	91.6 ± 5.2	92.5 ± 3.2	91.8 ± 5.0	**89.7 ± 6.3**	94.1 ± 1.7	94.1 ± 1.0	2.38
Walker2d-ME	Model-based SAC	98.0 ± 18.5	**97.4 ± 13.9**	98.0 ± 12.5	100.3 ± 11.7	100.0 ± 11.6	109.9 ± 2.4	111.0 ± 4.0	102.9 ± 8.1	106.4 ± 9.3	24.35
	CQL	109.6 ± 0.2	108.7 ± 0.3	108.5 ± 0.5	108.8 ± 0.3	**108.1 ± 1.4**	109.4 ± 1.0	108.8 ± 0.8	109.1 ± 0.0	109.5 ± 0.5	0.19
	MOPO	109.9 ± 2.8	111.6 ± 2.1	109.6 ± 1.2	111.9 ± 2.1	106.1 ± 2.6	**106.1 ± 5.5**	112.9 ± 2.2	111.2 ± 2.1	111.2 ± 1.9	5.94
	IQL	109.3 ± 0.7	105.8 ± 4.1	105.6 ± 3.2	108.2 ± 1.4	**102.9 ± 2.1**	107.6 ± 2.5	109.0 ± 0.2	107.5 ± 2.1	108.4 ± 3.7	3.46
	RAMBO	69.3 ± 45.1	75.8 ± 38.5	72.3 ± 38.2	68.6 ± 38.4	71.2 ± 42.7	77.1 ± 46.7	**62.5 ± 37.0**	71.9 ± 39.7	68.5 ± 44.6	18.54
	Mobile	114.8 ± 2.3	114.2 ± 2.2	113.9 ± 1.9	113.7 ± 2.2	**112.8 ± 2.5**	114.2 ± 2.6	113.5 ± 2.0	114.6 ± 1.9	114.7 ± 2.7	0.33
	GIRO (VAE)	110.9 ± 2.7	**98.7 ± 16.1**	101.4 ± 11.0	108.3 ± 2.4	101.8 ± 8.8	110.5 ± 2.9	110.4 ± 2.0	109.3 ± 2.4	110.9 ± 3.0	21.34
	GIRO (Diffusion)	105.1 ± 3.7	103.7 ± 3.5	**94.5 ± 14.6**	99.9 ± 3.6	96.9 ± 14.4	109.3 ± 7.0	98.3 ± 15.0	98.8 ± 11.2	100.9 ± 9.7	18.07

each task and condition. The disturbance mechanisms applied to the evaluation environment are described below. GIRO and model-based SAC are trained with generated samples from the CVAE world model. GIRO diffusion results are trained with the diffusion world model.

State-IID: At every evaluation step, we independently sample uniform observation noise and add it to the state observed by the policy. For state dimension j, the noise is sampled as $\epsilon_{t,j} \sim \mathcal{U}(-0.1\sigma_j, 0.1\sigma_j)$, where σ_j is the standard deviation of that state dimension in the task's offline dataset.

State-Static: At the beginning of each episode, we sample a uniform bias for every state dimension using $b_j \sim \mathcal{U}(-0.1\sigma_j, 0.1\sigma_j)$. The same bias vector is added to every state observed by the policy throughout that episode. A new bias is sampled for each episode.

Action-IID: At every evaluation step, we add independently sampled Gaussian noise to the action produced by the policy. For action dimension j, the noise standard deviation is $0.1|a_{t,j}|$. The perturbed action is clipped to the valid action range before being applied to the environment.

Action-Static: At the beginning of each episode, we sample a multiplicative gain for every action dimension as $g_j \sim \mathcal{N}(1,\, 0.1^2)$. The same gain is applied throughout the episode, so the executed action is $\tilde{a}_{t,j} = g_j a_{t,j}$, followed by clipping to the valid action range.

Mass Disturbance: We multiply all MuJoCo body masses by a global scaling factor from $\{0.95,\, 1.05\}$ during policy evaluation and report performance under each setting.

Friction Disturbance: We multiply all MuJoCo geometry-friction parameters by a global scaling factor from $\{0.95,\, 1.05\}$ during policy evaluation and report performance under each setting.

Table 7: Aggregate robustness across nine D4RL tasks. Worst-Case Sum is the sum of the lowest disturbed mean for each task; Variance Sum is the sum of the reported task-level disturbance variances.

Policy	Worst-Case Sum ↑	Variance Sum ↓
Model-based SAC	547.0	393.11
CQL	605.4	320.75
MOPO	611.6	407.00
IQL	574.6	474.06
RAMBO	648.9	199.05
GIRO (VAE)	**714.2**	**137.77**
Mobile	707.1	496.09
GIRO (Diffusion)	691.0	425.11