# A Single Deep Preference-Conditioned Policy for Learning Pareto Coverage Sets

**Akihiro Kubo**[1,4,✉] **Kosuke Nakanishi**[4,2] **Shin Ishii**[4,3,1]
[1] International Research Center for Neurointelligence, The University of Tokyo, Tokyo, Japan
[2] Honda R&D Co., Ltd., Tokyo, Japan
[3] ATR Neural Information Analysis Laboratories, Kyoto, Japan
[4] Department of Information Science, Kyoto University, Kyoto, Japan
✉ Corresponding author: kubo-a@sys.i.kyoto-u.ac.jp

## Abstract

Preference-conditioned multi-objective reinforcement learning aims to learn a single policy that captures trade-offs across preferences, but under nonlinear scalarization the uniqueness and continuity of the preference-to-solution correspondence remain unclear. We study this problem using smooth Tchebycheff scalarization as a monotone utility. For nonempty compact convex attainable sets, we prove that each positive preference selects a unique return vector and that this Pareto-optimal vector depends Lipschitz-continuously on preferences bounded away from the simplex boundary. Under polyhedrality and a componentwise upper utopia point, sweeping positive preferences recovers the Pareto front. To compute these targets, we formulate the problem over occupancy measures and derive Concave Mirror Descent Policy Iteration (CMDPI), with an $O(k^{-1/2})$ objective-space convergence rate for exact tabular updates. We further show that each update is equivalent to solving a Kullback-Leibler-regularized MDP with the previous policy as reference, yielding a policy-iteration interpretation and finite-iterate policy continuity across preferences. We develop deep actor–critic surrogates that retain CMDPI's previous-policy regularization for discrete actions and use a simplified STCH-based update for continuous control. On eight discrete-action MO-Gymnasium tasks, deep CMDPI attains the best average hypervolume rank among evaluated baselines, with nonuniform task-wise and metric-wise gains. The continuous-control surrogate improves hypervolume on most, but not all, evaluated tasks.

## 1 Introduction

Multi-Objective Reinforcement Learning (MORL) optimizes multiple possibly competing reward signals, so the central object is not a single optimal policy but a set of Pareto-optimal trade-offs (Pareto set / Pareto front) [1–3]. With deep reinforcement learning, MORL has been extended to high-dimensional and continuous-control domains [4, 5]. A prominent direction is *preference-conditioned* learning, where a user-specified preference vector $\omega$ is input to a single deep model that outputs diverse trade-off policies on demand [6–8, 5, 9, 10]. We focus on *dense coverage* of Pareto-optimal objective vectors: by scanning preferences, we aim to reach a neighborhood of any Pareto-optimal objective vector in the objective space [3, 11, 2, 12, 5].

A standard route is to scalarize the objective vector by a utility function and solve a single-objective problem. In MORL taxonomies, if user utility is restricted to positive weighted sums, one targets a *convex coverage set (CCS)*; if one allows arbitrary monotonically increasing utilities, one targets a *Pareto coverage set (PCS)* [3, 13]. CCS-oriented methods admit clean formulations (often compatible with dynamic programming / linear programming), but even for convex attainable sets, the optimal

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

objective vector for a fixed preference can be non-unique (e.g., a whole supporting face), which makes the target ambiguous and can induce preference-wise switching/discontinuity [14, 3, 2, 15]. In practice, they also tend to concentrate on extreme (vertex) solutions, making dense coverage challenging [3, 2]. PCS-oriented approaches can reach non-vertex trade-offs, but the resulting problems are typically more complex; achieving stable optimization often relies on objective modifications (e.g., strong regularization) or heuristics, which may deviate from the Pareto front [16–18]. Figure 1 illustrates this linear–nonlinear tension.

To bridge this gap, we adopt the Smooth Tchebycheff (STCH) scalarization [19], a smooth approximation [20] of a Tchebycheff-type nonlinear utility [21]. For discounted multi-objective MDPs with finite state/action spaces, the attainable objective set $\mathcal{J}$ is a convex polytope [22]. For any preference $\omega \in \Omega^m_{++} := \{\omega \in \mathbb{R}^m \mid \sum_l \omega_l = 1,\ \omega_l > 0\ \forall l\}$, let $\mathbf{J}^\omega$ be an STCH-optimal objective vector. Beyond its known Pareto optimality [19], we prove that this vector is unique (Proposition 2.1). We also establish full Pareto-front coverage under the polyhedral and componentwise upper-bound assumptions of Proposition 2.2, and Lipschitz continuity of $\omega \mapsto \mathbf{J}^\omega$ on $\Omega^m_\delta := \{\omega \in \Omega^m_{++} \mid \omega_l \geq \delta\ \forall l\}$ for any $0 < \delta \leq 1/m$ (Theorem 2.3). These properties justify stable preference scanning.

To solve the resulting nonlinear scalarized problem *without altering the objective*, we derive mirror descent in the occupation-measure space. Using negative conditional entropy and relative smoothness (Lemma 3.1), we obtain an $O(1/k)$ utility-suboptimality bound under the Lu–Freund–Nesterov framework [23]. Combined with STCH quadratic growth, this yields $O(k^{-1/2})$ convergence to $\mathbf{J}^\omega$ for exact tabular CMDPI (Theorem 3.2). Each iteration admits an implementation via policy evaluation based on a soft Bellman fixed point and policy improvement by exponential reweighting [16, 24]. This equivalence provides an explicit iterative policy-update view of nonlinear MORL (often formulated as convex optimization over occupation measures) [25, 26] and motivates a preference-conditioned deep approximation. Importantly, for a fixed continuous full-support initialization, exact tabular CMDPI defines an algorithm-induced policy $\pi^\omega_k$ at every finite iteration $k$, and this policy is continuous in $\omega$. It neither selects a unique optimal policy nor guarantees convergence to a unique limiting policy selector. By contrast, prior single-network MORL methods [7, 6, 8] typically assume such preference-to-policy regularity without verification and rely on function approximation to generalize across preferences.

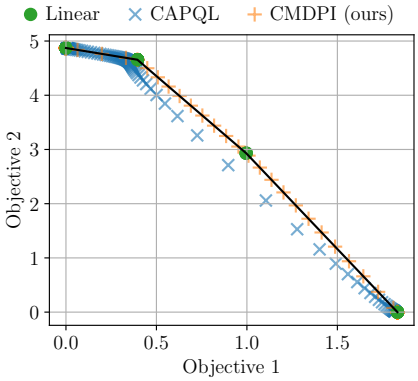



Figure 1: Illustration of a key limitation of existing methods: linear scalarization tends to favor vertex solutions of the Pareto front (black lines), while CAPQL [17] exhibits biased coverage. CMDPI (ours) achieves denser and more uniform Pareto-optimal coverage.

Finally, although we do not provide a theoretical analysis under function approximation, we study deep model-free and model-based actor–critic surrogates on MO-Gymnasium [27]. Their aggregate ranks are favorable, but the task-wise results show nonuniform gains and several regressions relative to linear-scalarization baselines. Our contributions are:

1. For STCH, we establish objective-space uniqueness and Lipschitz preference sensitivity for compact convex attainable sets.
2. We derive CMDPI in occupation-measure space and prove an $O(k^{-1/2})$ objective-space convergence rate for exact tabular updates, without modifying the scalarized objective.
3. We rewrite the update in an equivalent policy-iteration form and provide a tabular, finite-iterate motivation for deep preference-conditioned learning, without claiming a function-approximation guarantee.

## 2 Problem Setup and STCH Analysis

### 2.1 Multi-objective discounted MDP

We consider a discounted multi-objective Markov decision process (MOMDP) specified by a finite state space $\mathcal{S}$, a finite action space $\mathcal{A}$, a discount factor $\gamma \in (0, 1)$, a transition kernel $P(\cdot \mid s, a)$, and

an $m$-dimensional vector reward $\mathbf{r}(s,a) \in \mathbb{R}^m$. The initial distribution $p_0 \in \Delta(\mathcal{S})$ (the probability simplex over $\mathcal{S}$) is assumed to be strictly positive over states, i.e., $p_0(s) > 0$ for all $s \in \mathcal{S}$.

For a stochastic stationary policy $\pi(\cdot|s) \in \Delta(\mathcal{A})$, define $\mathbf{V}^\pi(s) \coloneqq \mathbb{E}_\pi\Big[\sum_{t\geq 0} \gamma^t \mathbf{r}(s_t,a_t) \mid s_0 = s\Big]$, $\mathbf{J}(\pi) \coloneqq \mathbb{E}_{p_0}[\mathbf{V}^\pi(s)]$ and let $\Pi$ be the set of such policies with achievable set $\mathcal{J} := \{\mathbf{J}(\pi) : \pi \in \Pi\}$.

### 2.2 Pareto optimality in MOMDP

For $\mathbf{x}, \mathbf{y} \in \mathbb{R}^m$, we say that $\mathbf{y}$ *(strictly) dominates* $\mathbf{x}$ if $y_i \geq x_i$ for all $i$ and $y_j > x_j$ for some $j$. An objective vector $\mathbf{x} \in \mathcal{J}$ is *Pareto optimal* if there is no $\mathbf{y} \in \mathcal{J}$ that dominates $\mathbf{x}$. A policy $\pi$ such that $\mathbf{J}(\pi) = \mathbf{x}$ for some Pareto-optimal $\mathbf{x}$ is called a *Pareto-optimal policy*. The set of all Pareto-optimal objective vectors $\mathcal{P}(\mathcal{J})$ is called the *Pareto front*, and a set of policies whose image in the objective space equals the Pareto front is called a *Pareto coverage set (PCS)* [3, 13]. A central goal in MOMDPs is to find a PCS.

For a stationary policy $\pi$, define its discounted occupancy measure and state marginal by

$$\mu_\pi(s,a) := (1-\gamma)\mathbb{E}_\pi\Big[\sum\nolimits_{t\geq 0} \gamma^t \mathbf{1}\{s_t = s, a_t = a\}\Big], \quad \rho_\pi(s) := \sum_a \mu_\pi(s,a),$$

and let $\mathcal{M} := \{\mu_\pi : \pi \in \Pi\}$. Then

$$\mathbf{J}(\pi) = \tfrac{1}{1-\gamma} \sum_{s,a} \mu_\pi(s,a)\mathbf{r}(s,a), \tag{1}$$

and $\mathcal{M}$ (hence $\mathcal{J}$) is a convex polytope [22, 28].

### 2.3 STCH Scalarization and Objective-Space Properties

We define the Smooth Tchebycheff (STCH) utility function[1] as

$$u(\mathbf{f}, \omega) := -\tau \log \sum_{k=1}^{m} \exp\left(\frac{\omega_k(I_k - f_k)}{\tau}\right). \tag{2}$$

Here, $\omega \in \Omega_{++}^m$ is a *preference vector* and $\tau > 0$ is the log-sum-exp smoothing parameter. The vector $\mathbf{I} = (I_1, \ldots, I_m)^\top$ is the utopia point used in the STCH scalarization. We impose a componentwise upper-bound condition on $\mathbf{I}$ for the full-front coverage claim in Proposition 2.2; uniqueness, Pareto optimality, and our Lipschitz sensitivity results do not require this positional condition. When its dependence matters, we write $u(\mathbf{f}, \omega; \mathbf{I})$ and $\mathbf{J}^\omega(\mathbf{I})$.

As shown by Lin et al. [19], as $\tau \to 0$, the STCH utility $u(\cdot, \omega)$ converges pointwise to the (non-smooth) weighted Tchebycheff scalarization. Thus, $\tau$ controls the degree of smoothing via the log-sum-exp approximation.

In tabular settings, the utopia point can be chosen explicitly. In deep or sample-based implementations, however, $\mathbf{I}$ is generally unknown and must be estimated from data; the estimation procedure used in this work is described in Appendix H. Appendix C establishes a Lipschitz bound on utopia-point sensitivity of the exact STCH-optimal return vector, complemented by an empirical sensitivity study of deep methods in Appendix J.5.1.

The function $u(\cdot, \omega)$ is concave, continuous, and strictly increasing component-wise, with $\nabla_{\mathbf{f}} u(\mathbf{f}, \omega) \in \mathbb{R}_{++}^m$ for every $\mathbf{f} \in \mathbb{R}^m$, where $\mathbb{R}_{++}^m := \{x \in \mathbb{R}^m : \ x_i > 0 \ \forall i\}$. Pareto optimality of its optima for positive preferences is known [19, Theorem 3.7]. Although $u(\cdot, \omega)$ is not generally strictly concave, we next establish objective-space uniqueness, polyhedral coverage[2], and Lipschitz preference sensitivity for the scalarized MOMDP

$$\max_{\pi \in \Pi} u(\mathbf{J}(\pi), \omega). \tag{3}$$

**Proposition 2.1** (Uniqueness in the objective space)**.** *Fix $\omega \in \Omega_{++}^m$ and assume that $\mathcal{J}$ is a compact convex set. Then problem* (3) *admits a* unique *optimal objective vector $\mathbf{J}^\omega \in \mathcal{J}$, and $\mathbf{J}^\omega$ is Pareto optimal.*

[1] Lin et al. [19] use the minimization form; we use its sign-reversed maximization form.

[2] For finite-horizon MOMDPs, Qiu et al. [29, Propositions 7.3 and 7.2] establish the corresponding STCH uniqueness and coverage results, respectively.

**Proposition 2.2** (Pareto-front coverage for polytopes)**.** *Fix $\tau > 0$. Assume that $\mathcal{J}$ is a compact convex polytope and that $I_i \geq \max_{\mathbf{J} \in \mathcal{J}} J_i$ for every $i$. If $\mathbf{J} \in \mathcal{J}$ is Pareto optimal, then there exists $\omega \in \Omega_{++}^m$ such that $\mathbf{J}$ is the optimal objective vector of* (3)*.*

Under these assumptions, $\{\mathbf{J}^\omega : \omega \in \Omega_{++}^m\}$ coincides with the Pareto front. Unlike the smooth-front analysis of Lin et al. [19], our proof uses polyhedral normals; Appendix A details the assumptions, a nonpolyhedral counterexample, and a two-objective strengthening.

Since the objective $\mathbf{J}^\omega$ is uniquely determined for each $\omega$, we can consider the mapping from $\omega \in \Omega_{++}^m$ to the objective space:

**Theorem 2.3** (Preference sensitivity)**.** *Let $\mathcal{J}$ be a compact convex set and let $\mathbf{I} \in \mathbb{R}^m$ be arbitrary. For $0 < \delta \leq 1/m$, the mapping $\phi : \Omega_\delta^m \to \mathcal{J}$ defined by $\phi(\omega) := \mathbf{J}^\omega$ is Lipschitz continuous on $\Omega_\delta^m$.*

Proofs of Propositions 2.1–2.2 and Theorem 2.3 are deferred to Appendices A–B. Appendix C also establishes Lipschitz utopia-point sensitivity (Proposition C.1).

Under the assumptions of Proposition 2.2, exact coverage also permits arbitrarily dense approximation of the Pareto front; this is our *dense coverage* interpretation. Uniqueness concerns objective vectors, so convergence to $\mathbf{J}^\omega$ does not require a unique optimal policy.

### 2.4 Entropy geometry and KL/Bregman identity

For two policies $\pi$ and $\pi_0$ with $\pi_0(a \mid s) > 0$ for all $(s, a)$, define the occupancy-weighted conditional KL by

$$\Gamma_\gamma(\pi, \pi_0) := \sum_{s,a} \mu_\pi(s, a) \log \frac{\pi(a \mid s)}{\pi_0(a \mid s)} = \sum_s \rho_\pi(s)\, \mathrm{KL}(\pi(\cdot \mid s) \,\|\, \pi_0(\cdot \mid s))\,. \tag{4}$$

Since $p_0(s) > 0$ for all $s$, the relative interior of $\mathcal{M}$ is $\mathrm{ri}(\mathcal{M}) = \{\mu \in \mathcal{M} : \mu(s, a) > 0\, \forall (s, a)\} \neq \varnothing$ (see Appendix D). For $\mu, \mu_0 \in \mathrm{ri}(\mathcal{M})$, define the negative conditional entropy $\psi_\gamma$ and the induced Bregman divergence $D_{\psi_\gamma}$:

$$\psi_\gamma(\mu) := \sum_{s,a} \mu(s, a) \log \frac{\mu(s, a)}{\rho_\mu(s)}, \quad D_{\psi_\gamma}(\mu \mid \mu_0) := \psi_\gamma(\mu) - \psi_\gamma(\mu_0) - \langle \nabla \psi_\gamma(\mu_0), \mu - \mu_0 \rangle.$$

We extend $\psi_\gamma$ continuously to $\mathcal{M}$ using $0 \log 0 = 0$. Thus $D_{\psi_\gamma}(\mu \mid \mu_0)$ is defined for $\mu \in \mathcal{M}$ and $\mu_0 \in \mathrm{ri}(\mathcal{M})$; only its second argument requires strict positivity. The following lemma shows that $D_{\psi_\gamma}$ corresponds to the policy-space regularizer $\Gamma_\gamma$.

**Lemma 2.4** (Correspondence of $\Gamma_\gamma$ and $D_{\psi_\gamma}$)**.** *Assume $p_0(s) > 0$ for all $s$. Let $\mu \in \mathcal{M}$ and $\mu' \in \mathrm{ri}(\mathcal{M})$ correspond to policies $\pi \leftrightarrow \mu$ and $\pi' \leftrightarrow \mu'$. Then*

$$\Gamma_\gamma(\pi, \pi') = D_{\psi_\gamma}(\mu \mid \mu') \tag{5}$$

The proof is provided in Appendix D.

### 2.5 Relative smoothness and mirror descent

The following notion is central to convergence analyses of mirror descent for convex optimization.

**Definition 2.5** (Relative smoothness)**.** Let $Q$ be a nonempty closed convex feasible set, $f$ a differentiable convex function on $Q$, and $\psi$ a continuous convex function on $Q$ that is differentiable on $\mathrm{ri}(Q)$. We say that $f$ is $L_{\mathrm{rel}}$*-relatively smooth with respect to* $\psi$ if there exists $L_{\mathrm{rel}} > 0$ such that, for all $x \in \mathrm{ri}(Q)$ and $y \in Q$,

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + L_{\mathrm{rel}} D_\psi(y \mid x). \tag{6}$$

If $f$ is $L_{\mathrm{rel}}$-relatively smooth and the exact iterates remain in $\mathrm{ri}(Q)$ starting from $x_0 \in \mathrm{ri}(Q)$, mirror descent [23, Theorem 3.1] with the update

$$x_{k+1} \in \arg\min_{x \in Q} \{\langle \nabla f(x_k), x \rangle + L_{\mathrm{rel}} D_\psi(x \mid x_k)\} \tag{7}$$

achieves an $O(1/k)$ rate in objective suboptimality:

$$f(x_k) - f^* \leq \frac{L_{\mathrm{rel}}}{k} D_\psi(x^* \mid x_0), \tag{8}$$

for $k \geq 1$, where $f^* := \min_{x \in Q} f(x)$ and $x^* \in \arg\min_{x \in Q} f(x)$ (not necessarily unique and possibly on the relative boundary of $Q$). In our applications, we will instantiate $x$ with an occupancy measure and $\psi$ with $\psi_\gamma$.

# 3 Mirror-Descent Policy Iteration under Fixed Preferences

## 3.1 Occupancy-Measure Formulation

The problem in (3) can be written over $\mu \in \mathcal{M}$ using (1) as

$$\bar{\mathbf{J}}(\mu) := \tfrac{1}{1-\gamma} \sum_{s,a} \mu(s,a)\mathbf{r}(s,a), \quad f(\mu) := -u(\bar{\mathbf{J}}(\mu), \omega).$$

yielding the convex program

$$\min_{\mu \in \mathcal{M}} f(\mu). \tag{9}$$

Convexity follows since $u$ is concave and $\bar{\mathbf{J}}$ is linear. By Proposition 2.1, the optimal objective $\mathbf{J}^\omega$ is unique; hence we analyze value/objective convergence, i.e., $f(\mu_k) \to f^\star$ and $\mathbf{J}(\pi_k) \to \mathbf{J}^\omega$.

The core idea of this work is to solve (9) by mirror descent over the discounted occupancy-measure polytope $\mathcal{M}$. Specifically, at iteration $k$ we perform

$$\mu_{k+1} = \arg\min_{\mu \in \mathcal{M}} \left\{ \langle \nabla f(\mu_k), \mu \rangle + L_{\mathrm{rel}}\, D_{\psi_\gamma}(\mu \mid \mu_k) \right\}. \tag{10}$$

**Lemma 3.1** (Occupancy-measure relative smoothness)**.** *The function $f$ is relatively smooth with respect to $\psi_\gamma$: there exists $L_{\mathrm{rel}} > 0$ such that* (6) *holds for $x \in \mathrm{ri}(\mathcal{M})$ and $y \in \mathcal{M}$.*

The proof is given in Appendix E. Applying the standard mirror-descent bound (8) with $x_k = \mu_k$ and $\psi = \psi_\gamma$ gives an $O(1/k)$ utility-suboptimality rate; Section 3.4 converts it to an objective-space rate.

## 3.2 From occupancy update to a KL-regularized policy update

The optimization (10) is convex in $\mu$ and does not directly yield an implementable policy update. However, Lemma 2.4 shows that $D_{\psi_\gamma}(\mu \mid \mu_k)$ can be identified with $\Gamma_\gamma(\pi, \pi_k)$, which allows us to pull back (10) to the policy space as a KL-regularized MDP [16] with reference policy $\pi_k$.

Let $\mu_k \leftrightarrow \pi_k$. By the chain rule,

$$\nabla_\mu f(\mu_k)(s,a) = -\tfrac{1}{1-\gamma} \langle \mathbf{g}_k, \mathbf{r}(s,a) \rangle, \quad \mathbf{g}_k := \nabla_{\mathbf{J}} u(\mathbf{J}(\pi_k), \omega) \in \mathbb{R}^m_{++}.$$

Define $r_k(s,a) := \langle \mathbf{g}_k, \mathbf{r}(s,a) \rangle$. Then (10) yields

$$\pi_{k+1} \in \arg\max_{\pi \in \Pi} \mathbb{E}_\pi \Big[ \sum\nolimits_{t \geq 0} \gamma^t r_k(s_t, a_t) \Big] - L_{\mathrm{rel}}\, \Gamma_\gamma(\pi, \pi_k). \tag{11}$$

## 3.3 Solving the regularized MDP: soft Bellman evaluation + multiplicative policy improvement

Our method can be implemented as a policy-iteration scheme that alternates between: (i) dynamic multi-objective $\to$ single-objective weighting (via $\mathbf{g}_k$), and (ii) KL-regularized greedy improvement with respect to the resulting scalar reward, as is discussed in the following.

Equation (11) is a special case of regularized MDPs with a closed-form optimal policy [16]. We set $\alpha_{\mathrm{KL}} := L_{\mathrm{rel}}(1-\gamma)$, the previous-policy KL/prox scale[3].

[3]This is distinct from the STCH smoothing parameter $\tau$ and from the entropy coefficient $\alpha_{\mathrm{ent}}$ used by the deep SAC surrogate.

**Policy evaluation (soft Bellman fixed point)** For any scale $\lambda > 0$, define

$$\chi_\lambda(s'; Q, \pi_k) := \lambda \log \sum_{b \in \mathcal{A}} \pi_k(b|s') \exp\left(\frac{Q(s', b)}{\lambda}\right). \tag{12}$$

Then the $Q$-function corresponding to (11) satisfies the soft Bellman fixed point

$$Q(s, a) = r_k(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[\chi_{\alpha_{\mathrm{KL}}}(s'; Q, \pi_k)]. \tag{13}$$

We denote its solution by $Q_{k+1}$. This operator is a $\gamma$-contraction and thus admits a unique fixed point; see Geist et al. [16].

**Policy improvement (KL-prox / multiplicative weights)** Given $Q_{k+1}$, the optimizer of (11) is obtained in closed form, state-wise:

$$\pi_{k+1}(a|s) \propto \pi_k(a|s) \exp\left(\frac{Q_{k+1}(s, a)}{\alpha_{\mathrm{KL}}}\right). \tag{14}$$

This is a KL-regularized greedy update with base measure $\pi_k$, and can be viewed as the policy-space representation of the mirror-descent step (10).

Putting everything together, our proposed method is summarized as Algorithm 3 in Appendix J.1. Since it is derived from mirror descent applied to (the negative of) a concave utility, we refer to it as *Concave Mirror Descent Policy Iteration (CMDPI)*.

**Connection to CAPQL** Our method is related to CAPQL [17] in the following sense: (i) when the scalarization is linear, i.e., $u(\mathbf{J}, \omega) = \omega^\top \mathbf{J}$ and $f(\mu) = -\frac{1}{1-\gamma} \sum_{s,a} \mu(s, a)\, \omega^\top \mathbf{r}(s, a)$, and (ii) when the reference policy is fixed to the uniform distribution, i.e., $\pi_k = U(\Delta_{\mathcal{A}})$. In this uniform-reference special case, $\alpha_{\mathrm{KL}}$ acts as an entropy coefficient: larger values impose stronger regularization and may deviate further from optimizing the original unregularized objective.

**Finite-step continuity** Appendix G shows that exact tabular CMDPI preserves preference-continuity of the policy for every finite iteration, assuming a continuous full-support initialization. While the result is tabular, this algorithm-induced regularity motivates a preference-conditioned deep model. It does not imply uniqueness or convergence of a limiting policy selector, nor does it guarantee the function-approximation implementation.

### 3.4 Objective-Space Convergence of Exact CMDPI

**Theorem 3.2** (Objective-space convergence of exact CMDPI)**.** *Assume finite $\mathcal{S}, \mathcal{A}$ and $p_0(s) > 0$ for all $s$. Fix $\tau > 0$, $\mathbf{I} \in \mathbb{R}^m$, $0 < \delta \leq 1/m$, and $\omega \in \Omega_\delta^m$. Let $\pi_0$ have full support and generate $\mu_k \leftrightarrow \pi_k$ by the exact updates* (10) *with a valid relative-smoothness constant $L_{\mathrm{rel}}$ from Lemma 3.1. For any optimal occupancy $\mu^\star \in \arg\min_{\mu \in \mathcal{M}} f(\mu)$ and all $k \geq 1$,*

$$\|\mathbf{J}(\pi_k) - \mathbf{J}^\omega\|_2 \leq \sqrt{\frac{C_{\mathcal{J},\mathbf{I},\tau,\delta}\, L_{\mathrm{rel}} D_{\psi_\gamma}(\mu^\star \mid \mu_0)}{k}} = O(k^{-1/2}), \tag{15}$$

*where $C_{\mathcal{J},\mathbf{I},\tau,\delta}$ is defined in Corollary F.1.*

The proof in Appendix F first obtains the utility-suboptimality bound

$$u(\mathbf{J}^\omega, \omega) - u(\mathbf{J}(\pi_k), \omega) \leq (L_{\mathrm{rel}}/k) D_{\psi_\gamma}(\mu^\star \mid \mu_0)$$

from (8), then applies Corollary F.1. Theorem 2.3 bounds how much $\mathbf{J}^\omega$ changes with the preference, while the bound above shows how quickly exact tabular CMDPI approaches $\mathbf{J}^\omega$ for a fixed preference.

### 3.5 Deep Actor-Critic Implementation Sketch

Motivated by the equivalence in Section 3.2, we construct a deep actor–critic surrogate for multi-objective RL (Algorithm 2). It borrows the soft target and exponential-improvement structure of (13)–(14), but is not covered by the tabular equivalence. We adopt an off-policy optimization style akin to Off-PAC [30] and, more specifically, Soft Actor–Critic (SAC) [31].

Our goal is to build a *single* function approximator that continuously represents solutions across preference vectors. Concretely, the critic is a neural network $\mathbf{Q}_\phi$ that takes $(s, a, \omega)$ as input and outputs an $m$-dimensional vector, while the actor is a neural network $\pi_\theta$ that takes $(s, \omega)$ as input and outputs a distribution over $\mathcal{A}$. Here $\phi$ and $\theta$ are the critic and actor parameters, respectively.

**Critic learning** The critic minimizes the squared Bellman residual:

$$L_c(\phi) = \mathbb{E}_{s,a\sim\mathcal{D},\ \omega\sim U(\cdot)}\left[\sum_{k=1}^{m}\left(y^{(k)} - Q_\phi^{(k)}(s,a,\omega)\right)^2\right], \tag{16}$$

where $\mathcal{D}$ is a replay buffer and $U(\cdot)$ denotes the uniform distribution over $\Omega^m$. The target component $y^{(k)}$ is the $k$-th entry of the target vector

$$\mathbf{y} := \mathbf{r} + \gamma\big[\mathbf{Q}_{\bar{\phi}}(s',a',\omega) - \alpha_{\text{ent}}\log\pi_\theta(a'|s',\omega)\,\mathbf{1}\big], \tag{17}$$

constructed from a replay sample $(s,a,\mathbf{r},s')$, an action $a'\sim\pi_\theta(\cdot|s',\omega)$, and a target critic $\mathbf{Q}_{\bar{\phi}}$, which are commonly used in Deep RL field [31–33].

We interpret this as a vector SAC policy-evaluation target [31, 16]. For finite $\mathcal{A}$, the uniform reference $U(a|s) = 1/|\mathcal{A}|$ gives $-\mathrm{KL}(\pi\|U) = H(\pi) - \log|\mathcal{A}|$. For a fixed $\bar{\omega}\in\Omega^m$, suppose that $\pi_\theta$ closely approximates the policy induced by (14):

$$\pi^{\bar{\phi}}(\cdot|s,\omega) := \frac{1}{Z(s)}\exp\big(\bar{\omega}^\top\,\mathbf{Q}_{\bar{\phi}}(s,\cdot,\omega)/\alpha_{\text{ent}}\big)\,,$$

where $Z(s)$ is a normalization constant. For a fixed $\bar{\omega}\in\Omega^m$, $\pi^{\bar{\phi}}$ is the corresponding uniform-reference soft-greedy policy for $q(s,\cdot) = \bar{\omega}^\top\mathbf{Q}_{\bar{\phi}}(s,\cdot,\omega)$ [16]. By arguments analogous to Section 3.3,

$$\bar{\omega}^\top\mathbf{y} \approx \bar{\omega}^\top\mathbf{r} + \gamma\,\chi_{\alpha_{\text{ent}}}\big(s';\bar{\omega}^\top\mathbf{Q}_{\bar{\phi}},U\big) + \gamma\alpha_{\text{ent}}\log|\mathcal{A}|.$$

Here $\chi$ is defined in (12); the approximation holds in expectation over $a'$ for the fixed direction $\bar{\omega}$ under the stated soft-greedy assumption. PCSAC uses $\tilde{\mathbf{V}}_\phi = \mathbf{Q}_\phi(s,\tilde{a},\omega) - \alpha_{\text{ent}}\log\pi_\theta(\tilde{a}|s,\omega)\mathbf{1}$ and the critic-local direction $\mathbf{g}(\omega) = \nabla u(\tilde{\mathbf{V}}_\phi,\omega)$. Since $\mathbf{g}(\omega)^\top\mathbf{1}\in[\omega_{\min},\omega_{\max}]$, scalarizing the broadcast entropy term yields the sample-dependent coefficient $\alpha_{\text{ent}}\mathbf{g}(\omega)^\top\mathbf{1}$. This is a deep surrogate, not exact (13).

**Actor learning** Equation (14) is equivalent to minimizing the KL divergence to a target distribution proportional to $\pi_k\exp(Q/\alpha_{\text{KL}})$. For a fixed direction $\bar{\omega}$, the uniform-reference SAC analogue therefore gives

$$\arg\min_\theta\ \mathbb{E}_{s\sim\mathcal{D}}\big[\mathrm{KL}\big(\pi_\theta(\cdot|s,\omega)\ \big|\ \pi^\phi(\cdot|s,\omega)\big)\big] = \arg\max_\theta\ \mathbb{E}_{s\sim\mathcal{D}}\left[\bar{\omega}^\top\tilde{\mathbf{V}}_\phi(s)\right], \tag{18}$$

with the target distribution defined for that same $\bar{\omega}$. Accordingly, we define the actor loss

$$L_\pi(\theta) := -\mathbb{E}_{s\sim\mathcal{D}}\left[\mathrm{sg}[\mathbf{g}(\omega)]^\top\tilde{\mathbf{V}}_\phi(s)\right]. \tag{19}$$

We summarize the resulting policy optimization algorithm for deep MORL in Algorithm 2 (Appendix H), referred to as *PCSAC*.

With the replay distribution fixed, $\tilde{a} = f_\theta(s,\varepsilon,\omega)$ and stop-gradient through $\mathbf{g}$ give exactly the chain rule for $u(\tilde{\mathbf{V}}_\phi(s,\varepsilon,\omega),\omega)$. Thus, under the usual interchange of differentiation and expectation, PCSAC performs stochastic ascent on $\widetilde{U}(\theta) := \mathbb{E}_\omega\mathbb{E}_{s\sim\mathcal{D},\,\varepsilon}\big[u(\tilde{\mathbf{V}}_\phi(s,\varepsilon,\omega),\omega)\big]$. For every fixed $\omega$, concavity gives $\mathbb{E}_{s,\varepsilon}[u(\tilde{\mathbf{V}}_\phi,\omega)] \le u(\mathbb{E}_{s,\varepsilon}[\tilde{\mathbf{V}}_\phi],\omega)$, so $\widetilde{U}(\theta)$ is a replay-distribution Jensen lower bound, not the tabular objective $u(\mathbf{J}(\pi_\theta),\omega)$. The two can have different maximizers.

The critic-local direction is a heuristic, not an unbiased or consistent estimator of $\nabla_{\mathbf{J}}u(\mathbf{J}(\pi),\omega)$. Its discrepancy includes critic/entropy evaluation error, nonlinear aggregation, and replay/initial-distribution mismatch; larger minibatches do not remove this structural gap. Appendix I gives the exact decomposition and an $L_u = \omega_{\max}^2/(4\tau)$ bound. Utility-gradient estimators adapted to preference-conditioned off-policy data remain open.

## 4 Numerical Experiments

We empirically evaluate the proposed methods in three stages. First, in Section 4.1, we study a tabular MOMDP to examine whether the proposed CMDPI update exhibits the intended planning behavior in a controlled setting. Second, in Section 4.2, we evaluate discrete-action MO-Gymnasium benchmarks to test how the deep PCSAC and CMDPI surrogates scale beyond the tabular setting. Finally, we evaluate continuous-control MO-Gymnasium tasks to assess a partially incorporated STCH-based surrogate (PCSAC). The code is available at `https://github.com/4kubo/cmdpi-morl`

Table 1: Task-wise final means on the eight discrete-action tasks. Each cell is LS/PCSAC/CMDPI; the environment column gives the multipliers for (EUM, HV, SP), respectively. Full uncertainty estimates and all baselines are in Table 6. Higher EUM/HV and lower SP are better.

| Task (EUM/HV/SP scale) | EUM | HV | SP |
|---|---|---|---|
| DST ($10^0/10^3/10^{-1}$) | $2.60/-41.83/3.18$ | $0.80/0.01/1.01$ | $0.00/2.70/2.90$ |
| DST-Concave ($10^1/10^3/10^0$) | $0.33/-3.03/4.15$ | $2.01/0.18/8.31$ | $5.65/14.60/126.04$ |
| FishWood ($10^1/10^3/10^2$) | $9.16/9.22/7.65$ | $0.56/0.94/1.56$ | $13.48/61.27/3.63$ |
| Fruit ($10^0/10^4/10^{-4}$) | $4.27/4.99/4.85$ | $0.53/1.55/1.52$ | $7.07/3.19/4.05$ |
| Lunar ($10^1/10^{10}/10^1$) | $1.21/-0.11/-0.86$ | $0.00/3.77/0.78$ | $1.38/44.10/18.57$ |
| Minecart ($10^0/10^1/10^0$) | $-6.45/-6.17/-4.11$ | $0.00/2.43/6.51$ | $0.00/126.73/8.50$ |
| Reacher ($10^1/10^6/10^{-2}$) | $2.28/2.54/2.31$ | $1.36/2.08/1.47$ | $4.28/15.47/5.77$ |
| FourRoom ($10^0/10^0/10^{-2}$) | $1.96/1.39/1.90$ | $7.40/3.10/8.20$ | $3.00/13.00/11.00$ |

### 4.1 Tabular MOMDP (planning)

We study a two-objective tabular MOMDP ($|\mathcal{S}| = 4$, $|\mathcal{A}| = 2$) to isolate a key limitation of existing PCS methods: linear scalarization tends to favor vertex solutions, while regularized methods may produce biased Pareto coverage. In a fully known planning setting, we compare linear scalarization, CAPQL [17], and the proposed CMDPI. For 100 preference vectors $\omega$, we compute the converged objective vectors and visualize them in Figure 1.

Linear scalarization tends to favor vertex solutions, while CAPQL produces non-vertex solutions but exhibits a bias–coverage trade-off. In contrast, across the tested planning settings, CMDPI recovers Pareto-optimal solutions with more uniform (denser) coverage of the front. Full details and sweeps are provided in Appendix J.1.

### 4.2 MO-Gymnasium

**Discrete-Action Tasks** We first evaluate eight representative discrete-action MO-Gymnasium benchmarks [27] to investigate whether the proposed deep methods, PCSAC and CMDPI, scale beyond the tabular setting.

We compare against linear scalarization (LS), CAPQL [17], PreCo [34], and C-MORL [9]. PCSAC and CMDPI are our proposed STCH-based methods. Here, CMDPI denotes the deep actor–critic surrogate in Algorithm 1 (Appendix H), whose critic target incorporates KL regularization toward the previous policy.

To reduce implementation confounders, LS, CAPQL, PCSAC, and CMDPI are implemented on the same PreCo codebase. C-MORL is included as a recent Pareto-front discovery baseline, although it is not a single preference-conditioned policy method.

All methods are trained for 2M environment interaction steps on each task using eight random seeds. We evaluate each trained agent by sweeping preference vectors and computing hypervolume (HV), expected utility metric (EUM) [13], and sparsity (SP). Table 1 exposes the nonuniform task-wise effects, while Table 4 in Appendix J.3 shows the average-rank summary. Metric scales are task-specific (raw HV spans roughly $10^0$–$10^{11}$ over all benchmarks), so an unnormalized cross-task average would be dominated by a few tasks; ranks supplement rather than replace raw values.

CMDPI has the best aggregate HV rank, but it does not uniformly dominate either LS or PCSAC. Relative to LS, its mean HV ratio is about 1.08 on Reacher, 1.11 on FourRoom, and 1.26 on DST, versus roughly 2.8–4.1 on FishWood, Fruit, and DST-Concave; Lunar and Minecart have zero LS HV[4], so a ratio is not meaningful. CMDPI exceeds LS on EUM in five of eight tasks, but is worse on FishWood, Lunar, and FourRoom, with a substantial Lunar regression, and it is not best on SP.

Against PCSAC, CMDPI has higher HV on five tasks; PCSAC is clearly higher on Lunar and Reacher, and Fruit is comparable. PCSAC also falls below LS on DST, DST-Concave, and FourRoom,

[4]The HV reference point is the componentwise minimum over evaluation returns from all methods, checkpoints, and seeds (Appendix J.2.1). On these tasks, the final LS evaluations attain these minima in some objectives; their dominated hypervolume relative to this reference is zero.

Table 2: Task-wise final means for the corresponding continuous-action linear/STCH pairs. Each MF cell is CAPQL-MF/PCSAC-MF, and each MB cell is CAPQL-MB/PCSAC-MB. The task column gives multipliers for (EUM, HV, SP). Full uncertainties and all methods are in Table 7.

| Task (EUM/HV/SP scale) | EUM-MF | EUM-MB | HV-MF | HV-MB | SP-MF | SP-MB |
|---|---|---|---|---|---|---|
| HalfCheetah ($10^3/10^7/10^6$) | 2.66/2.57 | 4.64/3.63 | 4.43/5.03 | 7.00/6.25 | 5.48/0.28 | 4.91/0.35 |
| Hopper-2d ($10^3/10^7/10^6$) | 3.32/3.15 | 3.59/3.67 | 1.35/1.29 | 1.56/1.74 | 1.13/0.48 | 0.32/1.00 |
| Hopper-3d ($10^3/10^{10}/10^6$) | 2.09/1.96 | 2.74/2.63 | 3.08/2.60 | 5.23/5.14 | 0.05/0.09 | 0.02/0.01 |
| Swimmer ($10^1/10^5/10^4$) | 1.88/2.29 | 2.80/7.51 | 2.01/3.92 | 2.88/6.67 | 3.14/5.25 | 0.34/2.04 |
| TruncAnt-2d ($10^3/10^7/10^6$) | 4.94/5.06 | 5.93/6.17 | 6.78/6.96 | 8.64/8.97 | 0.79/0.44 | 2.10/3.61 |
| TruncAnt-3d ($10^3/10^{11}/10^5$) | 2.04/2.13 | 4.53/4.54 | 1.75/1.96 | 6.06/6.10 | 0.57/0.29 | 0.06/0.15 |
| TruncHumanoid ($10^3/10^7/10^5$) | 4.93/5.85 | 6.28/6.46 | 2.32/3.09 | 3.78/3.95 | 1.42/0.82 | 2.76/0.52 |
| Walker2d ($10^3/10^7/10^6$) | 2.11/2.39 | 1.73/2.08 | 2.08/2.54 | 1.75/2.23 | 1.75/0.99 | 0.42/0.58 |

whereas the previous-policy-regularized CMDPI variant recovers higher HV on those tasks. These observations are consistent with STCH accounting for much of the Pareto-front improvement and previous-policy regularization mitigating some STCH-only failures, but the experiment does not causally isolate a stabilizing effect. The mirror-descent structure supplies the utility-gap bound behind the tabular $O(k^{-1/2})$ objective-space rate and finite-iterate policy continuity, whereas Lipschitz continuity of $\omega \mapsto \mathbf{J}^\omega$ is a property of STCH itself. Full task-wise uncertainty estimates and all baselines remain in Appendix J.4.1.

**Continuous-Action Tasks** We next evaluate eight representative continuous-action MuJoCo benchmarks from MO-Gymnasium [27] to investigate whether the proposed STCH-based update remains effective in continuous-control domains.

We compare PD-MORL [8], COLA [35], GPI-PD [36], CAPQL-MF, PCSAC-MF, CAPQL-MB, and PCSAC-MB. Here, MF and MB denote model-free and model-based variants, respectively. CAPQL-MF/MB corresponds to the linear-scalarization actor–critic baseline in our implementation.

Model-based methods, namely GPI-PD, CAPQL-MB, and PCSAC-MB, are evaluated with 500K real-environment interaction steps, while model-free methods are evaluated with 2M steps. This protocol follows the standard sample-efficiency motivation of model-based RL while accounting for their higher computational cost per environment step.

We evaluate each trained agent using the same HV, EUM, and SP metrics as in the discrete-action experiments. The evaluation protocol is detailed in Appendix J.2.2. Appendix Table 5 summarizes the average ranks on the continuous-action tasks, separately aggregated for model-free and model-based methods.

Among the model-free methods, PCSAC-MF and COLA tie for the best average HV rank at 2.00, and COLA has the best average EUM rank. Among the model-based methods, PCSAC-MB achieves the best average rank on both HV and EUM.

The task-wise comparison in Table 2 limits the aggregate conclusion. PCSAC-MB improves HV over CAPQL-MB on six of eight tasks, but is clearly worse on HalfCheetah and marginally worse on Hopper-3d. Metric-wise behavior is likewise nonuniform. Thus the continuous experiments support scalability of an STCH-based deep surrogate on most tested tasks, not consistent superiority or verification of the tabular convergence theorem. Full uncertainties and all methods are provided in Appendix J.4.2.

We also ran separate preliminary tests of previous-policy KL regularization (Appendices J.6 and J.7).

Finally, we conduct sensitivity analyses on the utopia-point scale and the relative-smoothness/KL scale parameter. Additional details and sensitivity results are provided in Appendix J.5.

## 5 Related Work

**Coverage sets vs. preference sweeping (what "dense" means here).** Coverage sets—CCS and PCS—and representative computation methods are surveyed in MORL [3, 13]. CCS-oriented

planning typically returns a *set* sufficient for any (unknown) linear weight, e.g., via linear-support style enumeration of supporting solutions [37, 38]. This is different from our question: whether *sweeping preferences* yields *stable objective-space coverage*. Even with a convex attainable set, a fixed preference can select a whole supporting face, making the preference-to-solution map set-valued and potentially causing abrupt switching under infinitesimal preference changes [14, 15]. Our work isolates this "selection/regularity" issue and shows that STCH admits a unique, stable objective-space selection for compact convex attainable sets; full-front coverage additionally uses the polyhedral condition in Proposition 2.2.

**Nonlinear scalarization and SER/ESR.** Beyond linear scalarization, MORL distinguishes *scalarized expected returns* (SER: scalarize the expected return vector) and *expected scalarized returns* (ESR: take expectation of a nonlinear utility) [3]. ESR is closely tied to risk-sensitive/dynamic-programming viewpoints and faces time-consistency and optimization challenges [39, 40, 18], while recent work develops practical algorithms for concave/nonlinear objectives [25, 26]. However, for general nonlinear scalarizations it is still nontrivial whether a preference induces a *unique* objective vector and whether preference sweeping avoids unstable switching. In contrast, for STCH we prove objective-space uniqueness and Lipschitz continuity of $\omega \mapsto \mathbf{J}^\omega$ on $\Omega_\delta^m$ for compact convex attainable sets (SER viewpoint), and separately identify the stronger assumptions needed for full Pareto-front coverage.

**Deep MORL and an optimization view (single models, stability, and mirror descent).** Classical approaches grow an explicit *set of policies* to approximate coverage sets [2, 11, 12], whereas preference-conditioned deep methods aim to output a trade-off directly from a single model [6–8], including hypernetwork-style "many policies in one" [10] and meta-policy formulations [34]. Other recent work broadens solution notions beyond linear scalarization (e.g., Pareto stationarity) [17]. Yet, scalarization-grounded guarantees that preference sweeping stably fills the objective space (via uniqueness/continuity rather than implicit generalization) remain limited. On the optimization side, mirror descent and regularized (soft) policy-iteration schemes have been clarified for regularized MDPs [16, 41] and concave-utility RL [26, 24]. Our CMDPI combines occupancy-measure mirror descent under relative smoothness [23] with STCH quadratic growth, yielding an $O(k^{-1/2})$ objective-space rate for exact tabular updates without biasing the STCH objective, together with preference-continuity of finite-iteration policies.

## 6 Conclusion and Future Work

We established that STCH selects a unique Pareto-optimal target $\mathbf{J}^\omega$ for each positive preference on compact convex attainable sets, with Lipschitz continuity on $\Omega_\delta^m$. Under polyhedrality and a componentwise upper utopia point, positive preferences recover the full Pareto front. These results give preference-conditioned learning a uniquely defined objective-space target and control how the selected return changes with preferences.

We derived CMDPI as occupancy-measure mirror descent with an equivalent previous-policy KL-regularized policy-iteration form. An $O(1/k)$ utility-gap bound and STCH quadratic growth yield $O(k^{-1/2})$ objective-space convergence for exact tabular updates without modifying the scalarized objective. A continuous full-support initialization also yields preference-continuous policies at every finite iteration, motivating a single conditioned actor without requiring a unique optimal policy.

Tabular planning illustrates Pareto-front coverage. Deep surrogates on discrete and continuous tasks achieve favorable HV ranks, with nonuniform gains across tasks and metrics.

The deep actor optimizes a critic-local surrogate, so the tabular guarantees do not directly extend to it; nonlinear aggregation and distribution mismatch can persist even with accurate critics. Our utopia-point sensitivity bound controls the exact target, leaving robustness of learned policies to online estimation errors unproved. Continuity is not guaranteed at the simplex boundary, where objective-wise gradient signals can vanish. Future work should develop function-approximation bounds that account for surrogate mismatch, distribution shift, and error propagation, analyze online utopia estimation, and characterize the existence and preference-continuity of limiting-policy selections.

## Acknowledgments and Disclosure of Funding

This work was supported by the WPI AI Incubator Acceleration Grant 2026 to A.K.; JSPS KAKENHI (No. 22H04998, 25K24537) to K.N. and S.I.; JST Moonshot program Goal 3 (No. JPMJMS2631) to A.K., K.N. and S.I.; and NEDO Cyborg generative-AI project (No. JPNP14004) to S.I. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript. The authors declare no competing interests.

## References

[1] D. J. White. Multi-objective infinite-horizon discounted Markov decision processes. *Journal of Mathematical Analysis and Applications*, 89(2):639–647, 1982. doi: 10.1016/0022-247X(82)90122-6.

[2] Kristof Van Moffaert and Ann Nowé. Multi-objective reinforcement learning using sets of Pareto dominating policies. *Journal of Machine Learning Research*, 15(107):3663–3692, 2014. URL https://jmlr.org/papers/v15/vanmoffaert14a.html.

[3] Diederik M Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48: 67–113, 2013.

[4] Hossam Mossalam, Yannis M. Assael, Diederik M. Roijers, and Shimon Whiteson. Multi-objective deep reinforcement learning. In *Deep Reinforcement Learning Workshop at NIPS 2016*, 2016. URL https://arxiv.org/abs/1610.02707.

[5] Jie Xu, Yunsheng Tian, Pingchuan Ma, Daniela Rus, Shinjiro Sueda, and Wojciech Matusik. Prediction-guided multi-objective reinforcement learning for continuous robot control. In Hal Daumé, III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10607–10616. PMLR, 2020. URL https://proceedings.mlr.press/v119/xu20h.html.

[6] Runzhe Yang, Xingyuan Sun, and Karthik Narasimhan. A generalized algorithm for multi-objective reinforcement learning and policy adaptation. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://papers.neurips.cc/paper_files/paper/2019/hash/4a46fbfca3f1465a27b210f4bdfe6ab3-Abstract.html.

[7] Mathieu Reymond, Eugenio Bargiacchi, and Ann Nowé. Pareto conditioned networks. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pages 1110–1118. International Foundation for Autonomous Agents and Multiagent Systems, 2022.

[8] Toygun Basaklar, Suat Gumussoy, and Umit Y. Ogras. PD-MORL: Preference-driven multi-objective reinforcement learning algorithm. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=zS9sRyaPFlJ.

[9] Ruohong Liu, Yuxin Pan, Linjie Xu, Lei Song, Pengcheng You, Yize Chen, and Jiang Bian. Efficient discovery of Pareto front for multi-objective reinforcement learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=fDGPIuCdGi.

[10] Erlong Liu, Yu-Chang Wu, Xiaobin Huang, Chengrui Gao, Ren-Jian Wang, Ke Xue, and Chao Qian. Pareto set learning for multi-objective reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 18789–18797, 2025. doi: 10.1609/aaai.v39i18.34068.

[11] Matteo Pirotta, Simone Parisi, and Marcello Restelli. Multi-objective reinforcement learning with continuous Pareto frontier approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015. doi: 10.1609/aaai.v29i1.9617.

[12] Mathieu Reymond and Ann Nowe. Pareto-DQN: Approximating the Pareto front in complex multi-objective decision problems. In *Proceedings of the Adaptive and Learning Agents Workshop 2019 (ALA-19) at AAMAS*, 2019. URL https://ala2019.vub.ac.be.

[13] Conor F. Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M. Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1):26, 2022. doi: 10.1007/s10458-022-09552-y. URL https://link.springer.com/article/10.1007/s10458-022-09552-y.

[14] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[15] Axel Abels, Diederik Roijers, Tom Lenaerts, Ann Nowé, and Denis Steckelmacher. Dynamic weights in multi-objective deep reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 11–20. PMLR, 2019. URL https://proceedings.mlr.press/v97/abels19a.html.

[16] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2160–2169. PMLR, 2019. URL https://proceedings.mlr.press/v97/geist19a.html.

[17] Haoye Lu, Daniel Herman, and Yaoliang Yu. Multi-objective reinforcement learning: Convexity, stationarity and Pareto optimality. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=TjEzIsyEsQ6.

[18] Nianli Peng, Muhang Tian, and Brandon Fain. Multi-objective reinforcement learning with nonlinear preferences: Provable approximation for maximizing expected scalarized return. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 1632–1640. International Foundation for Autonomous Agents and Multiagent Systems, 2025.

[19] Xi Lin, Xiaoyuan Zhang, Zhiyuan Yang, Fei Liu, Zhenkun Wang, and Qingfu Zhang. Smooth Tchebycheff scalarization for multi-objective optimization. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 30479–30509. PMLR, 2024. URL https://proceedings.mlr.press/v235/lin24y.html.

[20] Amir Beck and Marc Teboulle. Smoothing and first order methods: A unified framework. *SIAM Journal on Optimization*, 22(2):557–580, 2012.

[21] V. Joseph Bowman, Jr. On the relationship of the Tchebycheff norm and the efficient frontier of multiple-criteria objectives. In Hervé Thiriez and Stanley Zionts, editors, *Multiple Criteria Decision Making*, volume 130 of *Lecture Notes in Economics and Mathematical Systems*, pages 76–86. Springer Berlin Heidelberg, 1976. doi: 10.1007/978-3-642-87563-2_5.

[22] Yining Li, Peizhong Ju, and Ness Shroff. How to find the exact Pareto front for multi-objective MDPs? In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=S4dItvpvAv.

[23] Haihao Lu, Robert M. Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018. doi: 10.1137/16M1099546. URL https://epubs.siam.org/doi/10.1137/16M1099546.

[24] Bianca M Moreno, Margaux Bregere, Pierre Gaillard, and Nadia Oudjane. Efficient model-based concave utility reinforcement learning through greedy mirror descent. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 2206–2214. PMLR, 2024. URL https://proceedings.mlr.press/v238/moreno24a.html.

[25] Mridul Agarwal, Vaneet Aggarwal, and Tian Lan. Multi-objective reinforcement learning with non-linear scalarization. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pages 9–17. International Foundation for Autonomous Agents and Multiagent Systems, 2022. URL https://www.ifaamas.org/Proceedings/aamas2022/pdfs/p9.pdf.

[26] Matthieu Geist, Julien Pérolat, Mathieu Laurière, Romuald Elie, Sarah Perrin, Oliver Bachem, Rémi Munos, and Olivier Pietquin. Concave utility reinforcement learning: The mean-field game viewpoint. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pages 489–497. International Foundation for Autonomous Agents and Multiagent Systems, 2022.

[27] Florian Felten, Lucas N. Alegre, Ann Nowé, Ana L. C. Bazzan, El Ghazali Talbi, Grégoire Danoy, and Bruno C. da Silva. A toolkit for reliable benchmarking and research in multi-objective reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 23671–23700. Curran Associates, Inc., 2023. doi: 10.52202/075280-1028. URL https://papers.neurips.cc/paper_files/paper/2023/hash/4aa8891583f07ae200ba07843954caeb-Abstract-Datasets_and_Benchmarks.html.

[28] Eitan Altman. *Constrained Markov Decision Processes*. Routledge, 2021.

[29] Shuang Qiu, Dake Zhang, Rui Yang, Boxiang Lyu, and Tong Zhang. Traversing Pareto optimal policies: Provably efficient multi-objective reinforcement learning. arXiv:2407.17466, 2024. URL https://arxiv.org/abs/2407.17466.

[30] Thomas Degris, Martha White, and Richard S. Sutton. Off-policy actor-critic. In *Proceedings of the 29th International Conference on Machine Learning*, pages 179–186, Madison, WI, USA, 2012. Omnipress. ISBN 9781450312851.

[31] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, and Sergey Levine. Soft actor-critic algorithms and applications. arXiv:1812.05905, 2018. URL https://arxiv.org/abs/1812.05905.

[32] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[33] Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1587–1596. PMLR, 2018. URL https://proceedings.mlr.press/v80/fujimoto18a.html.

[34] Yucheng Yang, Tianyi Zhou, Mykola Pechenizkiy, and Meng Fang. Preference controllable reinforcement learning with advanced multi-objective optimization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 71612–71647. PMLR, 2025. URL https://proceedings.mlr.press/v267/yang25ax.html.

[35] Pengyi Li, Hongyao Tang, Yifu Yuan, Jianye Hao, Zibin Dong, and Yan Zheng. COLA: Towards efficient multi-objective reinforcement learning with conflict objective regularization in latent space. In *Advances in Neural Information Processing Systems*, volume 38, pages 89759–89782. Curran Associates, Inc., 2025. doi: 10.52202/085713-2704. URL https://papers.neurips.cc/paper_files/paper/2025/hash/745aab792345825f8b4b6219aaf285fe-Abstract-Conference.html.

[36] Lucas N Alegre, Ana LC Bazzan, Diederik M Roijers, Ann Nowé, and Bruno C da Silva. Sample-efficient multi-objective learning via generalized policy improvement prioritization. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems*, pages 2003–2012. International Foundation for Autonomous Agents and Multiagent Systems, 2023.

[37] Diederik M Roijers, Shimon Whiteson, and Frans A Oliehoek. Computing convex coverage sets for multi-objective coordination graphs. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory*, pages 309–323. Springer, 2013. doi: 10.1007/978-3-642-41575-3_24.

[38] Diederik M. Roijers, Shimon Whiteson, and Frans A. Oliehoek. Linear support for multi-objective coordination graphs. In *Proceedings of the 13th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1297–1304. International Foundation for Autonomous Agents and Multiagent Systems, 2014. URL https://www.ifaamas.org/Proceedings/aamas2014/aamas/p1297.pdf.

[39] Ronald A. Howard and James E. Matheson. Risk-sensitive Markov decision processes. *Management Science*, 18(7):356–369, 1972. doi: 10.1287/mnsc.18.7.356.

[40] Andrzej Ruszczyński. Risk-averse dynamic programming for Markov decision processes. *Mathematical Programming*, 125(2):235–261, 2010. doi: 10.1007/s10107-010-0393-3.

[41] Gergely Neu, Vicenç Gómez, and Anders Jonsson. A unified view of entropy-regularized Markov decision processes. In *Deep Reinforcement Learning Symposium at NIPS 2017*, 2017. URL https://sites.google.com/view/deeprl-symposium-nips2017/home.

[42] Akihiro Kubo, Paavo Parmas, and Shin Ishii. Double horizon model-based policy optimization. *Transactions on Machine Learning Research*, 2025.

[43] Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://papers.neurips.cc/paper_files/paper/2019/hash/5faf461eff3099671ad63c6f3f094f7f-Abstract.html.

[44] Brandon Amos, Samuel Stanton, Denis Yarats, and Andrew Gordon Wilson. On the model-based stochastic value gradient for continuous reinforcement learning. In *Proceedings of the 3rd Conference on Learning for Dynamics and Control*, volume 144 of *Proceedings of Machine Learning Research*, pages 6–20. PMLR, 2021. URL https://proceedings.mlr.press/v144/amos21a.html.

[45] Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, volume 31, pages 4754–4765. Curran Associates, Inc., 2018. URL https://papers.neurips.cc/paper_files/paper/2018/hash/3de568f8597b94bda53149c7d7f5958c-Abstract.html.

[46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *The Seventh International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

[47] Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. In *The Sixth International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1ANxQW0b.

[48] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

[49] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, pages 8024–8035. Curran Associates, Inc., 2019. URL https://papers.nips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

# APPENDIX

## A Uniqueness of STCH Solutions in the Objective Space

In the main text, we treat the policy $\pi$ as the decision variable and consider

$$\max_{\pi \in \Pi} u(\mathbf{J}(\pi), \omega).$$

Since the utility $u(\cdot, \omega)$ depends only on the objective vector, we can equivalently optimize over the achievable objective set $\mathcal{J} := \{\mathbf{J}(\pi) : \pi \in \Pi\} \subset \mathbb{R}^m$:

$$\max_{\pi \in \Pi} u(\mathbf{J}(\pi), \omega) \;=\; \max_{\mathbf{c} \in \mathcal{J}} u(\mathbf{c}, \omega).$$

Therefore, it suffices to establish the *uniqueness of the optimal objective vector* in the objective space; we do not require the optimal policy to be unique (multiple policies may attain the same optimal objective vector).

Below we work in a general multi-objective optimization setting, without restricting to MOMDPs. Let $\mathcal{C} \subset \mathbb{R}^m$ be a nonempty compact convex set and fix an arbitrary value of the utopia-point parameter $I \in \mathbb{R}^m$. Define

$$F_\omega(\mathbf{c}) := \exp\left(\frac{-u(\mathbf{c}, \omega)}{\tau}\right) = \sum_{i=1}^{m} \exp\left(\frac{\omega_i (I_i - c_i)}{\tau}\right).$$

By the monotonicity of the exponential function,

$$\arg\max_{\mathbf{c} \in \mathcal{C}} u(\mathbf{c}, \omega) \;=\; \arg\min_{\mathbf{c} \in \mathcal{C}} F_\omega(\mathbf{c}).$$

**Proposition A.1** (Uniqueness of STCH solutions and Pareto optimality in the objective space)**.** *Fix $\omega \in \mathbb{R}^m_{++}$ and let $\mathcal{C} \subset \mathbb{R}^m$ be a nonempty compact convex set. Then the problem*

$$\max_{\mathbf{c} \in \mathcal{C}} u(\mathbf{c}, \omega)$$

*admits a* unique *optimal solution $\mathbf{c}^\omega \in \mathcal{C}$. Moreover, $\mathbf{c}^\omega$ is Pareto optimal (for maximization).*

*Proof. (Existence.)* Since $\mathcal{C}$ is compact and $u(\cdot, \omega)$ is continuous, an optimum exists.

*(Pareto optimality.)* Because $\omega \in \mathbb{R}^m_{++}$, the utility $u(\cdot, \omega)$ is strictly increasing in each component. Hence, if $\mathbf{c}' \geq \mathbf{c}$ and $\mathbf{c}' \neq \mathbf{c}$, then $u(\mathbf{c}', \omega) > u(\mathbf{c}, \omega)$. Therefore, the optimizer $\mathbf{c}^\omega$ cannot be dominated by any other feasible point, and is Pareto optimal. This is also a special case of the general fact proved in Lin et al. [19] that STCH optima with strictly positive weights are Pareto optimal.

*(Uniqueness.)* Since the maximization is equivalent to $\min_{\mathbf{c} \in \mathcal{C}} F_\omega(\mathbf{c})$, it suffices to show that $F_\omega$ is strictly convex. Indeed,

$$F_\omega(\mathbf{c}) = \sum_{i=1}^{m} \exp\left(\frac{\omega_i I_i}{\tau}\right) \exp\Big(-\frac{\omega_i}{\tau} c_i\Big),$$

so its Hessian is diagonal:

$$\nabla^2 F_\omega(\mathbf{c}) = \operatorname{diag}\left(\left(\frac{\omega_1}{\tau}\right)^2 e^{\omega_1 (I_1 - c_1)/\tau},\; \ldots,\; \left(\frac{\omega_m}{\tau}\right)^2 e^{\omega_m (I_m - c_m)/\tau}\right).$$

Each diagonal entry is strictly positive for every value of the utopia-point parameter $I$ because $\omega_i > 0$ and the exponential term is positive; hence $\nabla^2 F_\omega(\mathbf{c}) \succ 0$ for all $\mathbf{c}$, which implies that $F_\omega$ is (globally) strictly convex. Therefore the minimizer over the convex set $\mathcal{C}$ is unique, and so is the maximizer of $u$. □

Next we prove Pareto-front coverage for polytopes. The coverage analysis of Lin et al. [19, Theorem 3.8 and Corollary 3.9] uses a smooth-front representation with a strictly positive gradient and bounded second derivatives (their Assumption A.1). Here we instead use strictly positive polyhedral normals and a utopia point satisfying the componentwise upper-bound condition to establish coverage for any fixed $\tau > 0$, without assuming a smooth Pareto front. We first recall the definition of the normal cone. For a convex set $\mathcal{C}$ and a point $\mathbf{c}^\star \in \mathcal{C}$, define

$$N_{\mathcal{C}}(\mathbf{c}^\star) := \{\mathbf{w} \in \mathbb{R}^m : \langle \mathbf{w}, \mathbf{c} - \mathbf{c}^\star \rangle \leq 0 \;\; \forall \mathbf{c} \in \mathcal{C}\}.$$

**Lemma A.2** (For polytopes, Pareto optimality implies the existence of a strictly positive normal)**.** *Let $\mathcal{C} \subset \mathbb{R}^m$ be a nonempty compact convex polytope, and let $\mathbf{c}^\star \in \mathcal{C}$ be Pareto optimal (for maximization), i.e., there is no $\mathbf{c} \in \mathcal{C}$ such that $\mathbf{c} \geq \mathbf{c}^\star$ and $\mathbf{c} \neq \mathbf{c}^\star$. Then*

$$N_{\mathcal{C}}(\mathbf{c}^\star) \cap \mathbb{R}^m_{++} \neq \varnothing.$$

*Proof.* We argue by contradiction. Assume $N_{\mathcal{C}}(\mathbf{c}^\star) \cap \mathbb{R}^m_{++} = \varnothing$. Since $\mathcal{C}$ is a polytope, there exist a matrix $A$ and vector $b$ such that $\mathcal{C} = \{\mathbf{c} : A\mathbf{c} \leq b\}$. Let

$$\mathcal{I} := \{j : \ a_j^\top \mathbf{c}^\star = b_j\}$$

be the set of active constraints at $\mathbf{c}^\star$ (where $a_j^\top$ denotes the $j$-th row of $A$). A standard polyhedral fact is that

$$N_{\mathcal{C}}(\mathbf{c}^\star) = \operatorname{cone}\{a_j : \ j \in \mathcal{I}\}, \qquad T_{\mathcal{C}}(\mathbf{c}^\star) = \{\mathbf{d} : \ a_j^\top \mathbf{d} \leq 0 \ \forall j \in \mathcal{I}\},$$

where $T_{\mathcal{C}}$ denotes the tangent cone.

Under the assumption that the cone $\operatorname{cone}\{a_j : j \in \mathcal{I}\}$ does not intersect $\mathbb{R}^m_{++}$, a separation argument for convex cones yields the existence of $\mathbf{d} \in \mathbb{R}^m_+ \setminus \{\mathbf{0}\}$ such that $a_j^\top \mathbf{d} \leq 0$ for all $j \in \mathcal{I}$; that is,

$$\mathbf{d} \in T_{\mathcal{C}}(\mathbf{c}^\star) \cap \left(\mathbb{R}^m_+ \setminus \{\mathbf{0}\}\right).$$

For polytopes, if $\mathbf{d} \in T_{\mathcal{C}}(\mathbf{c}^\star)$ then there exists $\varepsilon > 0$ small enough such that $\mathbf{c}^\star + \varepsilon\mathbf{d} \in \mathcal{C}$. Since $\mathbf{d} \in \mathbb{R}^m_+ \setminus \{\mathbf{0}\}$, we have $\mathbf{c}^\star + \varepsilon\mathbf{d} \geq \mathbf{c}^\star$ and $\mathbf{c}^\star + \varepsilon\mathbf{d} \neq \mathbf{c}^\star$, contradicting the Pareto optimality of $\mathbf{c}^\star$. Hence the assumption is false, and the claim follows. □

**Proposition A.3** (Every Pareto optimal point is STCH-optimal for some strictly positive preference)**.** *Let $\mathcal{C} \subset \mathbb{R}^m$ be a nonempty compact convex polytope. Assume that $I_i \geq \max_{\mathbf{c} \in \mathcal{C}} c_i$ for every $i$. If $\mathbf{c}^\star \in \mathcal{C}$ is Pareto optimal (for maximization), then there exists $\omega \in \mathbb{R}^m_{++}$ such that $\sum_i \omega_i = 1$ and $\mathbf{c}^\star$ is an optimizer of*

$$\max_{\mathbf{c} \in \mathcal{C}} u(\mathbf{c}, \omega).$$

*Proof.* For convenience we rewrite the problem as a minimization problem via a translation. Let $\mathbf{f} := I - \mathbf{c}$, $\mathcal{F} := \{I - \mathbf{c} : \mathbf{c} \in \mathcal{C}\}$, and $\mathbf{f}^\star := I - \mathbf{c}^\star$. Then $\mathcal{F}$ is also a nonempty compact convex polytope, and Pareto optimality of $\mathbf{c}^\star$ is equivalent to Pareto minimality of $\mathbf{f}^\star$ (componentwise minimization). Moreover,

$$\max_{\mathbf{c} \in \mathcal{C}} u(\mathbf{c}, \omega) \quad \Longleftrightarrow \quad \min_{\mathbf{f} \in \mathcal{F}} \ G_\omega(\mathbf{f}), \qquad G_\omega(\mathbf{f}) := \sum_{i=1}^m \exp\left(\frac{\omega_i f_i}{\tau}\right),$$

since $u(\mathbf{c}, \omega) = -\tau \log G_\omega(I - \mathbf{c})$ is a monotone transformation.

By Lemma A.2 applied to $\mathcal{C}$ and $\mathbf{c}^\star$, there exists $\lambda \in N_{\mathcal{C}}(\mathbf{c}^\star) \cap \mathbb{R}^m_{++}$. Under the affine change of variables $\mathbf{f} = I - \mathbf{c}$, the corresponding normal direction at $\mathbf{f}^\star$ is $-\lambda$, that is,

$$-\lambda \in N_{\mathcal{F}}(\mathbf{f}^\star).$$

Let $a_i := f_i^\star/\tau \ (\geq 0)$ and define for each $i$

$$\phi_i(t) := t\, e^{a_i t} \qquad (t > 0).$$

Since $a_i \geq 0$, $\phi_i$ is continuous, strictly increasing, and bijective from $(0, \infty)$ to $(0, \infty)$. Hence, for any constant $c > 0$, define

$$\omega_i := \phi_i^{-1}(c\,\lambda_i) \in (0, \infty).$$

The function $S(c) := \sum_i \phi_i^{-1}(c\lambda_i)$ is continuous and strictly increasing, with $S(c) \downarrow 0$ as $c \downarrow 0$ and $S(c) \uparrow \infty$ as $c \uparrow \infty$. Thus there is a unique $c > 0$ such that $S(c) = 1$; below we use this $c$, so $\omega \in \Omega^m_{++}$.

For this $\omega$, the gradient satisfies

$$\frac{\partial}{\partial f_i} G_\omega(\mathbf{f}) = \frac{\omega_i}{\tau} \exp\left(\frac{\omega_i f_i}{\tau}\right),$$

and therefore at $\mathbf{f}^\star$,

$$\big(\nabla G_\omega(\mathbf{f}^\star)\big)_i = \frac{1}{\tau}\,\omega_i e^{a_i\omega_i} = \frac{1}{\tau}\,\phi_i(\omega_i) = \frac{c}{\tau}\,\lambda_i.$$

Thus $\nabla G_\omega(\mathbf{f}^\star) = q\,\lambda$ with $q = c/\tau > 0$. Therefore,

$$-\nabla G_\omega(\mathbf{f}^\star) = q(-\lambda) \in N_{\mathcal{F}}(\mathbf{f}^\star),$$

because $-\lambda \in N_{\mathcal{F}}(\mathbf{f}^\star)$ and the normal cone is a cone. Equivalently,

$$\langle \nabla G_\omega(\mathbf{f}^\star), \mathbf{f} - \mathbf{f}^\star \rangle \geq 0 \qquad \forall \mathbf{f} \in \mathcal{F}.$$

Because $G_\omega$ is convex and $C^1$, this is the first-order optimality condition for convex minimization, implying $\mathbf{f}^\star \in \arg\min_{\mathbf{f}\in\mathcal{F}} G_\omega(\mathbf{f})$. Returning to the original variables gives $\mathbf{c}^\star \in \arg\max_{\mathbf{c}\in\mathcal{C}} u(\mathbf{c},\omega)$. Proposition A.1 makes this optimizer unique. □

*Remark* A.4 (Why polyhedrality is localized to coverage). The proof above needs a strictly positive normal at every Pareto point. This can fail for a compact convex nonpolyhedral set: for the unit disk in $\mathbb{R}^2$, $(0,1)$ is Pareto optimal, but its normal cone is $\{t(0,1) : t \geq 0\}$ and contains no strictly positive vector. Since $-\nabla_{\mathbf{c}} F_\omega$ is strictly positive for every $\omega > 0$, $(0,1)$ cannot be an STCH optimum. Thus compactness and convexity suffice for Proposition A.1, but not for full-front coverage.

**Proposition A.5** (Two-objective coverage for an arbitrary utopia-point parameter)**.** *Let $\mathcal{C} \subset \mathbb{R}^2$ be a nonempty compact convex polytope, let $\tau > 0$, and let $I \in \mathbb{R}^2$ be arbitrary. Every Pareto-optimal $\mathbf{c}^\star \in \mathcal{C}$ is the unique STCH optimum for some $\omega \in \Omega^2_{++}$.*

*Proof.* Lemma A.2 gives $\lambda \in N_{\mathcal{C}}(\mathbf{c}^\star) \cap \mathbb{R}^2_{++}$. Put $f_i^\star = I_i - c_i^\star$ and parameterize $\omega = (t, 1-t)$ for $t \in (0,1)$. The continuous positive function

$$\Psi(t) := \frac{t\exp(t f_1^\star/\tau)}{(1-t)\exp((1-t)f_2^\star/\tau)}$$

satisfies $\Psi(t) \to 0$ as $t \downarrow 0$ and $\Psi(t) \to \infty$ as $t \uparrow 1$, irrespective of the signs of $f_1^\star, f_2^\star$. Hence some $t^\star \in (0,1)$ satisfies $\Psi(t^\star) = \lambda_1/\lambda_2$. For $\omega = (t^\star, 1-t^\star)$, the positive vectors $-\nabla_{\mathbf{c}} F_\omega(\mathbf{c}^\star)$ and $\lambda$ have the same component ratio, so $-\nabla_{\mathbf{c}} F_\omega(\mathbf{c}^\star) = a\lambda \in N_{\mathcal{C}}(\mathbf{c}^\star)$ for some $a > 0$. This is the first-order condition for minimizing the strictly convex $F_\omega$ over $\mathcal{C}$, proving optimality and uniqueness. □

## B Lipschitz continuity of the STCH optimizer map

**Lemma B.1** (Lipschitz continuity of the STCH optimizer map)**.** *Let $\mathcal{C} \subset \mathbb{R}^m$ be a nonempty compact convex set (in particular, it may be a convex polytope). Fix $\tau > 0$ and an arbitrary value of the utopia-point parameter $\mathbf{I} \in \mathbb{R}^m$. For any $0 < \delta \leq 1/m$, define*

$$\Omega^m_\delta := \Big\{\omega \in \mathbb{R}^m \;\Big|\; \sum_{i=1}^m \omega_i = 1,\ \omega_i \geq \delta\ (i = 1,\dots,m)\Big\}.$$

*Consider the smooth Tchebycheff utility*

$$u(\mathbf{c},\omega) := -\tau \log \sum_{i=1}^m \exp\Big(\tfrac{\omega_i(I_i - c_i)}{\tau}\Big) \quad (\mathbf{c} \in \mathcal{C},\ \omega \in \Omega^m_\delta).$$

*For each $\omega \in \Omega^m_\delta$, let*

$$\mathbf{c}^\omega \in \arg\max_{\mathbf{c}\in\mathcal{C}} u(\mathbf{c},\omega)$$

*denote the optimizer, which is unique by Proposition A.1. Then the map $\phi : \Omega^m_\delta \to \mathcal{C}$ defined by $\phi(\omega) := \mathbf{c}^\omega$ is Lipschitz continuous on $\Omega^m_\delta$. More precisely, there exists a constant $L_\omega < \infty$ such that for all $\omega, \omega' \in \Omega^m_\delta$,*

$$\|\mathbf{c}^\omega - \mathbf{c}^{\omega'}\|_2 \;\leq\; L_\omega\,\|\omega - \omega'\|_2,$$

*where $\|\cdot\|_2$ denotes the Euclidean norm. Equivalent finite-dimensional norms give an adjusted finite constant.*

*Proof.* Since $-\tau\log(\cdot)$ is strictly decreasing, we have

$$\arg\max_{\mathbf{c}\in\mathcal{C}} u(\mathbf{c},\omega) = \arg\min_{\mathbf{c}\in\mathcal{C}} \Phi(\mathbf{c},\omega), \qquad \Phi(\mathbf{c},\omega) := \sum_{i=1}^{m} \exp\Big(\tfrac{\omega_i(I_i-c_i)}{\tau}\Big).$$

Thus, we treat $\mathbf{c}^\omega$ as the unique minimizer of $\Phi(\cdot,\omega)$.

The set $\mathcal{C}\times\Omega_\delta^m$ is compact. Hence the continuous quantities below attain finite extrema; define

$$h_- := \min_{i,\mathbf{c},\omega} \omega_i(I_i-c_i), \qquad U := \max_{i,\mathbf{c},\omega} \frac{1}{\tau}\left|1+\frac{\omega_i(I_i-c_i)}{\tau}\right| \exp\left(\frac{\omega_i(I_i-c_i)}{\tau}\right),$$

where all extrema range over $i\in[m]$ and $(\mathbf{c},\omega)\in\mathcal{C}\times\Omega_\delta^m$.

First, the Hessian of $\Phi(\cdot,\omega)$ with respect to $\mathbf{c}$ is diagonal:

$$\nabla^2_{\mathbf{cc}}\Phi(\mathbf{c},\omega) = \operatorname{diag}\Big(\Big(\tfrac{\omega_i}{\tau}\Big)^2 \exp\Big(\tfrac{\omega_i(I_i-c_i)}{\tau}\Big)\Big)_{i=1}^m.$$

For $\omega\in\Omega_\delta^m$, we have $\omega_i\ge\delta$ and $\omega_i(I_i-c_i)\ge h_-$, hence

$$\Big(\tfrac{\omega_i}{\tau}\Big)^2 \exp\Big(\tfrac{\omega_i(I_i-c_i)}{\tau}\Big) \;\ge\; \Big(\tfrac{\delta}{\tau}\Big)^2 \exp\Big(\tfrac{h_-}{\tau}\Big) =: \mu_\Phi \;>0.$$

Therefore, for every $\omega\in\Omega_\delta^m$, the function $\Phi(\cdot,\omega)$ is $\mu_\Phi$-strongly convex on $\mathcal{C}$, uniformly in $\omega$. Equivalently, its gradient is $\mu_\Phi$-strongly monotone:

$$\big\langle \nabla_{\mathbf{c}}\Phi(\mathbf{x},\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{y},\omega),\ \mathbf{x}-\mathbf{y}\big\rangle \;\ge\; \mu_\Phi\|\mathbf{x}-\mathbf{y}\|_2^2 \qquad (\mathbf{x},\mathbf{y}\in\mathcal{C}). \tag{20}$$

Next, we bound the Lipschitz constant of $\nabla_{\mathbf{c}}\Phi$ with respect to $\omega$. The gradient is

$$\nabla_{\mathbf{c}}\Phi(\mathbf{c},\omega) = \Big(-\tfrac{\omega_i}{\tau}\exp\big(\tfrac{\omega_i(I_i-c_i)}{\tau}\big)\Big)_{i=1}^m,$$

and the Jacobian $\nabla_\omega\nabla_{\mathbf{c}}\Phi(\mathbf{c},\omega)$ is also diagonal. For each coordinate $i$,

$$\frac{\partial}{\partial\omega_i}\Big(\frac{\partial\Phi}{\partial c_i}\Big) = -\frac{1}{\tau}\Big(1+\tfrac{\omega_i(I_i-c_i)}{\tau}\Big)\exp\Big(\tfrac{\omega_i(I_i-c_i)}{\tau}\Big).$$

By the definition of $U$, for all $(\mathbf{c},\omega)\in\mathcal{C}\times\Omega_\delta^m$,

$$\big\|\nabla_\omega\nabla_{\mathbf{c}}\Phi(\mathbf{c},\omega)\big\| \;\le\; U<\infty.$$

(For a diagonal matrix, the operator norm is bounded by the maximum absolute diagonal entry.) Hence, by the mean value inequality, for any $\mathbf{c}\in\mathcal{C}$ and any $\omega,\omega'\in\Omega_\delta^m$,

$$\big\|\nabla_{\mathbf{c}}\Phi(\mathbf{c},\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c},\omega')\big\| \;\le\; U\,\|\omega-\omega'\|_2. \tag{21}$$

We now use first-order optimality conditions written with the normal cone $N_{\mathcal{C}}(\cdot)$:

$$\mathbf{0}\in\nabla_{\mathbf{c}}\Phi(\mathbf{c}^\omega,\omega)+N_{\mathcal{C}}(\mathbf{c}^\omega), \qquad \mathbf{0}\in\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega')+N_{\mathcal{C}}(\mathbf{c}^{\omega'}).$$

Thus, there exist $\mathbf{v}\in N_{\mathcal{C}}(\mathbf{c}^\omega)$ and $\mathbf{v}'\in N_{\mathcal{C}}(\mathbf{c}^{\omega'})$ such that

$$\nabla_{\mathbf{c}}\Phi(\mathbf{c}^\omega,\omega)+\mathbf{v}=\mathbf{0}, \qquad \nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega')+\mathbf{v}'=\mathbf{0}.$$

Let $\mathbf{d}:=\mathbf{c}^\omega-\mathbf{c}^{\omega'}$. Taking the inner product of the difference of the two equations with $\mathbf{d}$ yields

$$\big\langle\nabla_{\mathbf{c}}\Phi(\mathbf{c}^\omega,\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega'),\ \mathbf{d}\big\rangle+\big\langle\mathbf{v}-\mathbf{v}',\ \mathbf{d}\big\rangle=0.$$

By monotonicity of the normal cone, we have $\langle\mathbf{v}-\mathbf{v}',\mathbf{d}\rangle\ge0$, hence

$$\big\langle\nabla_{\mathbf{c}}\Phi(\mathbf{c}^\omega,\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega'),\ \mathbf{d}\big\rangle\le0.$$

Decomposing the left-hand side using the same $\omega$ gives

$$\big\langle\nabla_{\mathbf{c}}\Phi(\mathbf{c}^\omega,\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega),\ \mathbf{d}\big\rangle+\big\langle\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega'),\ \mathbf{d}\big\rangle \;\le\; 0.$$

The first term is lower-bounded by $\mu_\Phi\|\mathbf{d}\|_2^2$ via (20). The second term is bounded by Cauchy–Schwarz and (21):

$$\big\langle\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega'),\ \mathbf{d}\big\rangle \;\ge\; -\big\|\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega)-\nabla_{\mathbf{c}}\Phi(\mathbf{c}^{\omega'},\omega')\big\|_2\,\|\mathbf{d}\|_2 \;\ge\; -U\|\omega-\omega'\|_2\,\|\mathbf{d}\|_2.$$

Combining these inequalities yields

$$\mu_\Phi\|\mathbf{d}\|_2^2\le U\|\omega-\omega'\|_2\|\mathbf{d}\|_2.$$

If $\mathbf{d}=\mathbf{0}$ the claim is trivial; otherwise, dividing both sides by $\|\mathbf{d}\|_2$ gives

$$\|\mathbf{c}^\omega-\mathbf{c}^{\omega'}\|_2=\|\mathbf{d}\|_2 \;\le\; \frac{U}{\mu_\Phi}\,\|\omega-\omega'\|_2.$$

Therefore $\phi$ is Lipschitz continuous on $\Omega_\delta^m$ with constant $L_\omega:=U/\mu_\Phi$. □

## C Utopia-point Sensitivity

**Proposition C.1** (Lipschitz sensitivity to the utopia point)**.** *Let $\mathcal{C} \subset \mathbb{R}^m$ be a nonempty compact convex set, let $\mathcal{I} \subset \mathbb{R}^m$ be nonempty, bounded, and convex, and fix $\tau > 0$ and $0 < \delta \le 1/m$. For $\omega \in \Omega_\delta^m$, write*

$$\mathbf{c}^\omega(I) := \arg\min_{\mathbf{c}\in\mathcal{C}} \Phi(\mathbf{c}, I; \omega), \qquad \Phi(\mathbf{c}, I; \omega) := \sum_{i=1}^m \exp\left(\frac{\omega_i(I_i - c_i)}{\tau}\right).$$

*Define the finite extrema*

$$g_- := \min_{i,\mathbf{c},I,\omega} \omega_i(I_i - c_i), \qquad g_+ := \max_{i,\mathbf{c},I,\omega} \omega_i(I_i - c_i),$$

*over $i \in [m]$, $(\mathbf{c}, I, \omega) \in \mathcal{C} \times \overline{\mathcal{I}} \times \Omega_\delta^m$. Then, uniformly over $\omega \in \Omega_\delta^m$,*

$$\|\mathbf{c}^\omega(I) - \mathbf{c}^\omega(I')\|_2 \le \kappa \|I - I'\|_2, \qquad \kappa := \delta^{-2} \exp\left(\frac{g_+ - g_-}{\tau}\right) \tag{22}$$

*for all $I, I' \in \mathcal{I}$.*

*Proof.* The Hessian in $\mathbf{c}$ and the mixed derivative are diagonal and satisfy

$$\nabla^2_{\mathbf{cc}}\Phi(\mathbf{c}, I; \omega) = \operatorname{diag}\left(\left(\frac{\omega_i}{\tau}\right)^2 e^{\omega_i(I_i - c_i)/\tau}\right)_{i=1}^m, \qquad \nabla_I \nabla_{\mathbf{c}}\Phi = -\nabla^2_{\mathbf{cc}}\Phi.$$

Consequently, on $\mathcal{C} \times \overline{\mathcal{I}} \times \Omega_\delta^m$,

$$\nabla^2_{\mathbf{cc}}\Phi \succeq \underbrace{(\delta/\tau)^2 e^{g_-/\tau}}_{\mu_I} \operatorname{Id}, \qquad \|\nabla_I \nabla_{\mathbf{c}}\Phi\|_{2\to 2} \le \underbrace{\tau^{-2} e^{g_+/\tau}}_{U_I}.$$

Convexity of $\mathcal{I}$ and the mean-value inequality therefore give $\|\nabla_{\mathbf{c}}\Phi(\mathbf{c}, I; \omega) - \nabla_{\mathbf{c}}\Phi(\mathbf{c}, I'; \omega)\|_2 \le U_I \|I - I'\|_2$. Let $\mathbf{c} = \mathbf{c}^\omega(I)$, $\mathbf{c}' = \mathbf{c}^\omega(I')$, and $\mathbf{d} = \mathbf{c} - \mathbf{c}'$. The first-order conditions and monotonicity of the normal cone imply

$$\langle \nabla_{\mathbf{c}}\Phi(\mathbf{c}, I; \omega) - \nabla_{\mathbf{c}}\Phi(\mathbf{c}', I'; \omega), \mathbf{d} \rangle \le 0.$$

Adding and subtracting $\nabla_{\mathbf{c}}\Phi(\mathbf{c}', I; \omega)$, strong monotonicity in $\mathbf{c}$, and Cauchy–Schwarz yield $\mu_I \|\mathbf{d}\|_2^2 \le U_I \|I - I'\|_2 \|\mathbf{d}\|_2$. Division when $\mathbf{d} \ne 0$ gives (22), because $U_I/\mu_I = \kappa$; the case $\mathbf{d} = 0$ is immediate. □

*Remark* C.2. No componentwise domination or fixed-margin condition on $I$ is needed for Proposition C.1; boundedness of $\mathcal{I}$ makes the constant uniform. The displayed upper bound deteriorates as $\tau \downarrow 0$ or $\delta \downarrow 0$, but this does not prove that the smallest possible Lipschitz constant does. The result concerns the exact STCH target, not a learned deep policy. Together with Proposition A.5, it shows that in two objectives a utopia-point error continuously reparameterizes the same Pareto front.

## D Equivalence between the Bregman divergence and the discounted-occupancy-weighted conditional KL

Let $\operatorname{ri}(\mathcal{M})$ denote the relative interior of $\mathcal{M}$. Under the standing assumptions, it satisfies

$$\operatorname{ri}(\mathcal{M}) = \{\mu \in \mathcal{M} : \ \mu(s, a) > 0 \ \forall (s, a)\}.$$

Assume that $p_0(s) > 0$ for all $s$, and that there exists a full-support policy $\bar{\pi}$ (i.e., $\bar{\pi}(a \mid s) > 0$ for all $(s, a)$). Then

$$\mu^{\bar{\pi}}(s, a) \ \ge \ (1 - \gamma) p_0(s) \bar{\pi}(a \mid s) \ > 0,$$

A full-support policy exists, for example the uniform policy. Thus $\mathcal{M}$ contains a strictly positive point; since $\mathcal{M}$ is defined by the discounted flow equalities and coordinatewise nonnegativity, its relative interior is exactly the strictly positive feasible set above and is nonempty. Also, $\rho_\mu(s) \ge (1 - \gamma) p_0(s) > 0$ for every $\mu \in \mathcal{M}$, including boundary points.

If $\mu(s,a) > 0$, then by the previous rewriting we can differentiate $\psi_\gamma$ directly, obtaining

$$\frac{\partial \psi_\gamma(\mu)}{\partial \mu(s,a)} = \log \frac{\mu(s,a)}{\rho_\mu(s)} = \log \pi_\mu(a \mid s).$$

Therefore,

$$(\nabla \psi_\gamma(\mu))(s,a) = \log \pi_\mu(a \mid s) \quad (\mu \in \mathrm{ri}(\mathcal{M})). \tag{23}$$

Fix $\mu_0 \in \mathrm{ri}(\mathcal{M})$ and take an arbitrary $\mu \in \mathcal{M}$. Using (23) and $\psi_\gamma(\mu) = \sum_{s,a} \mu(s,a) \log \pi_\mu(a \mid s)$, we have

$$\begin{aligned} D_{\psi_\gamma}(\mu|\mu_0) &= \sum_{s,a} \mu(s,a) \log \pi_\mu(a|s) - \sum_{s,a} \mu_0(s,a) \log \pi_{\mu_0}(a|s) \\ &\quad - \sum_{s,a} \log \pi_{\mu_0}(a|s) \big(\mu(s,a) - \mu_0(s,a)\big) \\ &= \sum_{s,a} \mu(s,a) \log \frac{\pi_\mu(a|s)}{\pi_{\mu_0}(a|s)} \\ &= \Gamma_\gamma(\mu, \mu_0), \end{aligned}$$

where the last line simply rewrites $\Gamma_\gamma(\mu, \mu_0)$ in terms of the policies induced by the occupancy measures, namely $\pi_\mu$ and $\pi_{\mu_0}$.

Hence,

$$D_{\psi_\gamma}(\mu|\mu_0) = \Gamma_\gamma(\mu, \mu_0) \quad (\mu_0 \in \mathrm{ri}(\mathcal{M}),\ \mu \in \mathcal{M}).$$

# E Proof of Lemma 3.1

This appendix makes explicit the definitions of $f$ and the reference function $\psi_\gamma$ used in Lemma 3.1, and clarifies how the (discounted) occupancy-measure theorem induces a unique correspondence from an occupancy measure to a stationary policy. We then prove a concrete form of relative smoothness on the feasible set of occupancy measures.

**Proposition E.1** (Relative smoothness on the occupancy-measure space (concrete form))**.** *Fix finite sets $\mathcal{S}, \mathcal{A}$ and a discount factor $\gamma \in (0,1)$. Let $\mathcal{M} \subset \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ be the discounted occupancy-measure polytope from the main text, with $p_0(s) > 0$ for all $s$. Then $\mathrm{ri}(\mathcal{M}) \neq \varnothing$, as shown in Appendix D. For each $\mu \in \mathrm{ri}(\mathcal{M})$, define*

$$\rho_\mu(s) := \sum_{a \in \mathcal{A}} \mu(s,a), \qquad \pi_\mu(a|s) := \frac{\mu(s,a)}{\rho_\mu(s)}$$

*(Here $\mu(s,a) > 0$ for every $(s,a)$, so in particular $\rho_\mu(s) > 0$.)*

*Let $\mathbf{r}(s,a) \in \mathbb{R}^m$ be componentwise bounded and set $R_{1,\max} := \max_{s,a} \|\mathbf{r}(s,a)\|_1 < \infty$. Define the linear map*

$$\mathbf{J}(\mu) := \frac{1}{1-\gamma} \sum_{s,a} \mu(s,a) \mathbf{r}(s,a) \in \mathbb{R}^m.$$

*Fix a temperature $\tau > 0$ and a preference vector $\omega \in \mathbb{R}^m_{++}$, and let $u(\cdot, \omega)$ be the Smooth Tchebycheff utility. Define*

$$\ell(\mathbf{z}) := -u(\mathbf{z}, \omega), \qquad f(\mu) := \ell(\mathbf{J}(\mu)).$$

*Moreover, define the reference function (negative conditional entropy)*

$$\psi_\gamma(\mu) := \sum_{s,a} \mu(s,a) \log \frac{\mu(s,a)}{\rho_\mu(s)} \qquad (\mu \in \mathcal{M}),$$

*using $0 \log 0 = 0$ at the boundary, and its Bregman divergence $D_{\psi_\gamma}(\nu|\mu)$ for $\nu \in \mathcal{M}$, $\mu \in \mathrm{ri}(\mathcal{M})$.*

*Then there exists a constant $L_{\mathrm{rel}} > 0$ such that, for all $\nu \in \mathcal{M}$ and $\mu \in \mathrm{ri}(\mathcal{M})$,*

$$f(\nu) \leq f(\mu) + \langle \nabla f(\mu), \nu - \mu \rangle + L_{\mathrm{rel}}\, D_{\psi_\gamma}(\nu|\mu).$$

*In particular, one may take*

$$L_{\mathrm{rel}} = \frac{\omega_{\max}^2 R_{1,\max}^2}{4\tau(1-\gamma)^4}, \qquad \omega_{\max} := \max_{k\in[m]} \omega_k.$$

*Hence $f$ is relatively smooth with respect to $\psi_\gamma$.*

*Proof.* **(Bridging the notation: uniqueness of $\mu \mapsto \pi_\mu$).** If $\mu \in \mathrm{ri}(\mathcal{M})$, then $\mu(s,a) > 0$ holds, and hence $\rho_\mu(s) = \sum_a \mu(s,a) > 0$ so that $\pi_\mu(\cdot|s)$ is a well-defined probability distribution. Moreover, since $\mu(s,a) = \rho_\mu(s)\pi_\mu(a|s)$, the conditional distribution $\pi_\mu$ induced by a given $\mu$ is unique. Therefore $f(\mu) = \ell(\mathbf{J}(\mu))$ is uniquely defined as a function of $\mu$.

First take $\mu, \nu \in \mathrm{ri}(\mathcal{M})$. We proceed in two steps: (1) $\ell_1$-smoothness of $f$ (Lipschitz continuity of the gradient in the dual norm $\ell_\infty$), and (2) an inequality controlling $\|\nu - \mu\|_1^2$ by $D_{\psi_\gamma}(\nu|\mu)$. Combining them yields the desired relative smoothness inequality.

**(1) $\ell_1$-smoothness: $\|\nabla f(\nu) - \nabla f(\mu)\|_\infty \le L_1 \|\nu - \mu\|_1$.** Since $\ell(\mathbf{z}) = \tau \log \sum_{k=1}^m \exp(\omega_k(I_k - z_k)/\tau)$ is a log-sum-exp function, its Hessian can be written as

$$\nabla^2 \ell(\mathbf{z}) = \frac{1}{\tau}\Omega\big(\mathrm{Diag}(p) - pp^\top\big)\Omega,$$

where $p$ is the softmax vector and $\Omega = \mathrm{diag}(\omega)$. For each $k, l$, we have $|(\mathrm{Diag}(p) - pp^\top)_{kl}| \le 1/4$, hence

$$\big|(\nabla^2 \ell(\mathbf{z}))_{kl}\big| \le \frac{\omega_{\max}^2}{4\tau}.$$

By the mean value theorem, for any $\Delta\mathbf{z}$ and any component $k$,

$$\big|(\nabla \ell(\mathbf{z} + \Delta\mathbf{z}) - \nabla \ell(\mathbf{z}))_k\big| \le \sum_{l=1}^m \big|(\nabla^2 \ell(\tilde{\mathbf{z}}))_{kl}\big|\, |\Delta z_l| \le \frac{\omega_{\max}^2}{4\tau}\|\Delta\mathbf{z}\|_1,$$

that is,

$$\|\nabla \ell(\mathbf{z} + \Delta\mathbf{z}) - \nabla \ell(\mathbf{z})\|_\infty \le \frac{\omega_{\max}^2}{4\tau}\|\Delta\mathbf{z}\|_1. \tag{24}$$

Since $\mathbf{J}(\mu)$ is linear, the chain rule gives, for each $(s,a)$,

$$\frac{\partial f}{\partial \mu(s,a)}(\mu) = \left\langle \nabla \ell(\mathbf{J}(\mu)), \frac{1}{1-\gamma}\mathbf{r}(s,a) \right\rangle.$$

Thus,

$$\left| \frac{\partial f}{\partial \mu(s,a)}(\nu) - \frac{\partial f}{\partial \mu(s,a)}(\mu) \right| \le \frac{\|\mathbf{r}(s,a)\|_1}{1-\gamma} \big\|\nabla \ell(\mathbf{J}(\nu)) - \nabla \ell(\mathbf{J}(\mu))\big\|_\infty.$$

Using $\|\mathbf{r}(s,a)\|_1 \le R_{1,\max}$ and (24), we obtain

$$\|\nabla f(\nu) - \nabla f(\mu)\|_\infty \le \frac{R_{1,\max}}{1-\gamma} \cdot \frac{\omega_{\max}^2}{4\tau}\|\mathbf{J}(\nu) - \mathbf{J}(\mu)\|_1.$$

Moreover, by the definition of $\mathbf{J}$ and the triangle inequality,

$$\|\mathbf{J}(\nu) - \mathbf{J}(\mu)\|_1 = \left\| \frac{1}{1-\gamma} \sum_{s,a} (\nu - \mu)(s,a)\mathbf{r}(s,a) \right\|_1 \le \frac{R_{1,\max}}{1-\gamma}\|\nu - \mu\|_1.$$

Hence,

$$\|\nabla f(\nu) - \nabla f(\mu)\|_\infty \le L_1 \|\nu - \mu\|_1, \qquad L_1 := \frac{\omega_{\max}^2 R_{1,\max}^2}{4\tau(1-\gamma)^2}.$$

This $\ell_1$-smoothness implies the standard descent lemma:

$$f(\nu) \le f(\mu) + \langle \nabla f(\mu), \nu - \mu \rangle + \frac{L_1}{2}\|\nu - \mu\|_1^2. \tag{25}$$

**(2) Controlling $\|\nu - \mu\|_1^2$ by $D_{\psi_\gamma}(\nu|\mu)$.** The gradient of $\psi_\gamma$ is $\nabla\psi_\gamma(\mu)(s,a) = \log\mu(s,a) - \log\rho_\mu(s)$, and one can verify that

$$D_{\psi_\gamma}(\nu|\mu) = \sum_{s\in\mathcal{S}} \rho_\nu(s)\,\mathrm{KL}(\pi_\nu(\cdot|s)\,\|\,\pi_\mu(\cdot|s))\,. \tag{26}$$

Using the occupancy-measure representation $\mu = \mu_{\pi_\mu}$ and $\nu = \mu_{\pi_\nu}$,

$$\mu_\pi(s,a) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t\, d_t^\pi(s)\pi(a|s), \qquad \rho_\pi(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t\, d_t^\pi(s),$$

where $d_t^\pi$ is the state distribution at time $t$. Let $\Delta(s) := \|\pi_\nu(\cdot|s) - \pi_\mu(\cdot|s)\|_1$. A standard argument using the non-expansiveness of Markov kernels in $\ell_1$ yields

$$\|\nu - \mu\|_1 \le \frac{1}{1-\gamma}\sum_{s\in\mathcal{S}}\rho_\nu(s)\Delta(s). \tag{27}$$

(Proof sketch: expand both occupancy measures as $(1-\gamma)\sum_t\gamma^t$ and bound the distribution shift at each time step by the accumulated action-distribution shifts, leading to the geometric factor $1/(1-\gamma)$.)

Applying Cauchy–Schwarz to (27) gives

$$\|\nu - \mu\|_1^2 \le \frac{1}{(1-\gamma)^2}\left(\sum_s \rho_\nu(s)\Delta(s)\right)^2 \le \frac{1}{(1-\gamma)^2}\sum_s \rho_\nu(s)\Delta(s)^2.$$

By Pinsker's inequality $\|\pi - \pi'\|_1^2 \le 2\mathrm{KL}(\pi\|\pi')$ applied to each state $s$ and (26), we obtain

$$\sum_s \rho_\nu(s)\Delta(s)^2 \le 2\sum_s \rho_\nu(s)\mathrm{KL}(\pi_\nu(\cdot|s)\,\|\,\pi_\mu(\cdot|s)) = 2D_{\psi_\gamma}(\nu|\mu).$$

Therefore,

$$\|\nu - \mu\|_1^2 \le \frac{2}{(1-\gamma)^2} D_{\psi_\gamma}(\nu|\mu). \tag{28}$$

**(3) Combining the two inequalities.** Substituting (28) into (25) yields

$$f(\nu) \le f(\mu) + \langle\nabla f(\mu), \nu - \mu\rangle + \frac{L_1}{2}\cdot\frac{2}{(1-\gamma)^2} D_{\psi_\gamma}(\nu|\mu),$$

that is,

$$f(\nu) \le f(\mu) + \langle\nabla f(\mu), \nu - \mu\rangle + L_{\mathrm{rel}} D_{\psi_\gamma}(\nu|\mu), \qquad L_{\mathrm{rel}} = \frac{L_1}{(1-\gamma)^2} = \frac{\omega_{\max}^2 R_{1,\max}^2}{4\tau(1-\gamma)^4}.$$

For fixed $\mu \in \mathrm{ri}(\mathcal{M})$ and arbitrary $\nu \in \mathcal{M}$, apply this inequality to $\nu_\varepsilon = (1-\varepsilon)\nu + \varepsilon\mu \in \mathrm{ri}(\mathcal{M})$ and let $\varepsilon \downarrow 0$. Continuity of $f$ and of the extended $\psi_\gamma$ yields the same inequality at $\nu$. This proves Proposition E.1 and Lemma 3.1. □

*Remark* E.2 (Scale and non-tightness of $L_{\mathrm{rel}}$). The displayed $L_{\mathrm{rel}}$ is a worst-case sufficient constant; we do not claim that it is tight. Its $(1-\gamma)^{-4}$ factor arises in two stages. The chain rule for $f = \ell \circ \mathbf{J}$ and the bound on $\|\mathbf{J}(\nu) - \mathbf{J}(\mu)\|_1$ contribute $(1-\gamma)^{-2}$ to $L_1$. Controlling occupancy perturbations by the conditional-KL Bregman divergence in (28) contributes the second $(1-\gamma)^{-2}$. The entrywise $1/4$ covariance bound, replacement of all weights by $\omega_{\max}^2$, and Pinsker's inequality are additional loose steps. A direct comparison with $D_{\psi_\gamma}$, without the intermediate $\ell_1^2$ bound, may be sharper.

The $O(1/k)$ utility-gap bound used in Theorem 3.2 needs only a finite valid constant, so the effective horizon changes the constant rather than the order. It should not be read as prescribing a practical deep-RL coefficient: at $\gamma = 0.995$, the worst-case scale in our continuous-control setup is of order $10^{11}$ for $L_{\mathrm{rel}}$, and $\alpha_{\mathrm{KL}} = L_{\mathrm{rel}}(1-\gamma)$ remains many orders above the tuned entropy coefficient $\alpha_{\mathrm{ent}} = 10^{-2}$. Function approximation and the online utopia-point estimate further prevent transferring this worst-case constant directly. The five-value sweeps in Tables 10–11 are empirical sensitivity checks only; their lack of monotone degradation does not validate the worst-case constant.

## F Objective-Space Convergence of Exact CMDPI

**Corollary F.1** (STCH quadratic growth)**.** *Let $\mathcal{C} \subset \mathbb{R}^m$ be nonempty, compact, and convex, fix $I \in \mathbb{R}^m$, $\tau > 0$, and $0 < \delta \leq 1/m$, and define*

$$\mathbf{c}^\omega := \arg\min_{\mathbf{c}\in\mathcal{C}} \Phi(\mathbf{c};\omega), \qquad \Phi(\mathbf{c};\omega) := \sum_{i=1}^{m} \exp\left(\frac{\omega_i(I_i - c_i)}{\tau}\right).$$

*and*

$$\mu_\Phi := \min_{\mathbf{c}\in\mathcal{C},\, \omega\in\Omega_\delta^m} \lambda_{\min}\big(\nabla^2_{\mathbf{cc}}\Phi(\mathbf{c};\omega)\big) > 0, \qquad \Phi_{\max} := \max_{\mathbf{c}\in\mathcal{C},\, \omega\in\Omega_\delta^m} \Phi(\mathbf{c};\omega) < \infty.$$

*Set $C_{\mathcal{C},I,\tau,\delta} := 2\Phi_{\max}/(\tau\mu_\Phi)$. For every $\omega \in \Omega_\delta^m$ and $\mathbf{c} \in \mathcal{C}$, with $\eta := u(\mathbf{c}^\omega,\omega) - u(\mathbf{c},\omega) \geq 0$,*

$$\|\mathbf{c} - \mathbf{c}^\omega\|_2^2 \leq C_{\mathcal{C},I,\tau,\delta}\, \eta. \tag{29}$$

*Proof.* Strong convexity of $\Phi(\cdot;\omega)$ and the variational inequality at its constrained minimizer $\mathbf{c}^\omega$ give

$$\|\mathbf{c} - \mathbf{c}^\omega\|_2^2 \leq \frac{2}{\mu_\Phi}\big(\Phi(\mathbf{c};\omega) - \Phi(\mathbf{c}^\omega;\omega)\big).$$

Since $\Phi = \exp(-u/\tau)$,

$$\Phi(\mathbf{c};\omega) = \Phi(\mathbf{c}^\omega;\omega)e^{\eta/\tau}.$$

For $x \geq 0$, $e^x - 1 \leq xe^x$; hence

$$\Phi(\mathbf{c};\omega) - \Phi(\mathbf{c}^\omega;\omega) \leq \frac{\eta}{\tau}\Phi(\mathbf{c};\omega) \leq \frac{\eta}{\tau}\Phi_{\max}.$$

Combining the two displays proves (29). □

*Remark* F.2. The constant explicitly depends on the Euclidean norm, $\mathcal{C}$, $I$, $\tau$, and $\delta$; it is not only a function of $(\tau, \delta, m)$. The bound applies to any $\mathbf{c} \in \mathcal{C}$, including the return of a stationary or parameterized policy, but $\eta$ is measured against the global optimum over $\mathcal{C}$. A restricted policy class therefore retains its global approximation gap.

*Proof of Theorem 3.2.* The full-support initialization and (14) ensure $\mu_k \in \mathrm{ri}(\mathcal{M})$ for every finite $k$. Lemma 3.1 and the standard three-point argument [23, proof of Theorem 3.1] give, for every $\nu \in \mathcal{M}$,

$$f(\mu_{k+1}) - f(\nu) \leq L_{\mathrm{rel}}\big[D_{\psi_\gamma}(\nu \mid \mu_k) - D_{\psi_\gamma}(\nu \mid \mu_{k+1})\big].$$

The gradients of $\psi_\gamma$ are evaluated only at $\mu_k$ and $\mu_{k+1}$, whose components are all strictly positive. Taking $\nu = \mu_k$ gives monotonicity of $f(\mu_k)$; summing with $\nu = \mu^\star$ yields (8) with $x_k = \mu_k$ and $\psi = \psi_\gamma$. The gap $f(\mu_k) - f^\star$ equals $u(\mathbf{J}^\omega,\omega) - u(\mathbf{J}(\pi_k),\omega)$, giving the stated utility-suboptimality bound. Apply Corollary F.1 with $\mathcal{C} = \mathcal{J}$ and take square roots to obtain (15). □

## G Continuity Guarantee for Finite-Step Tabular CMDPI

**Proposition G.1** (Continuity of finite-step tabular CMDPI)**.** *Consider a finite discounted multi-objective MDP with finite state and action spaces $\mathcal{S}$ and $\mathcal{A}$, discount factor $\gamma \in (0,1)$, and bounded vector-valued rewards. Let $\Omega \subset \mathbb{R}^m_{++}$ be a preference set, and suppose that the scalarization $u(\mathbf{J},\omega)$ is continuously differentiable in $\mathbf{J}$ and that*

$$(\mathbf{J},\omega) \mapsto \nabla_{\mathbf{J}} u(\mathbf{J},\omega)$$

*is continuous on its domain. Assume that the initialization $\omega \mapsto \pi_0(\omega)$ is continuous and has full support, i.e., $\pi_0(a \mid s;\omega) > 0$ for all $(s,a,\omega)$.*

*For fixed $\alpha_{\mathrm{KL}} > 0$, suppose that the exact tabular CMDPI recursion defines $\pi_{k+1}$ from $\pi_k$ as follows. First, set*

$$\mathbf{g}_k(\omega) = \nabla_{\mathbf{J}} u(\mathbf{J}(\pi_k(\omega)),\omega)\,,$$

*and define the scalarized reward*

$$r_k^\omega(s,a) = \mathbf{g}_k(\omega)^\top \mathbf{r}(s,a).$$

*Then let $Q^{\omega}_{k+1}$ be the unique fixed point of the soft Bellman equation*

$$Q(s,a) = r^{\omega}_{k}(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[ \chi(s'; Q, \pi_k(\omega)) \right],$$

*where*

$$\chi(s; Q, \pi) = \alpha_{\mathrm{KL}} \log \sum_{b \in \mathcal{A}} \pi(b \mid s) \exp\left( \frac{Q(s,b)}{\alpha_{\mathrm{KL}}} \right).$$

*Finally, update the policy by*

$$\pi_{k+1}(a \mid s; \omega) = \frac{\pi_k(a \mid s; \omega) \exp\big(Q^{\omega}_{k+1}(s,a)/\alpha_{\mathrm{KL}}\big)}{\sum_{b \in \mathcal{A}} \pi_k(b \mid s; \omega) \exp\big(Q^{\omega}_{k+1}(s,b)/\alpha_{\mathrm{KL}}\big)}.$$

*Then, for every finite $k \geq 0$, the map*

$$\omega \mapsto \pi_k(\omega)$$

*is continuous.*

*Proof.* We prove the claim by induction on $k$. The induction step consists of the following three parts:

(i) continuity of the scalarized reward $\omega \mapsto r^{\omega}_{k}$;

(ii) continuity of the Bellman fixed point $\omega \mapsto Q^{\omega}_{k+1}$;

(iii) continuity of the updated policy $\omega \mapsto \pi_{k+1}(\omega)$.

The only nontrivial step is (ii), because $Q^{\omega}_{k+1}$ is defined implicitly as the fixed point of a $\omega$-dependent Bellman operator. We handle this step using a standard perturbation argument for parameterized contractions.

The base case $k = 0$ holds by the assumption that $\omega \mapsto \pi_0(\omega)$ is continuous.

**(i) Continuity of the scalarized reward.** Assume that $\omega \mapsto \pi_k(\omega)$ is continuous. Since the state and action spaces are finite and $\gamma < 1$, the objective vector $\mathbf{J}(\pi)$ is continuous as a function of the tabular policy $\pi$. Hence

$$\omega \mapsto \mathbf{g}_k(\omega) = \nabla_{\mathbf{J}} u(\mathbf{J}(\pi_k(\omega)), \omega)$$

is continuous. Since the vector reward is bounded and finite-dimensional, $\omega \mapsto r^{\omega}_{k}$ is also continuous.

**(ii) Continuity of the Bellman fixed point.** For fixed $\pi$, the map

$$Q \mapsto \chi(s; Q, \pi)$$

is 1-Lipschitz with respect to the sup norm. Therefore, the soft Bellman operator

$$(T_{k,\omega} Q)(s,a) := r^{\omega}_{k}(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[ \chi(s'; Q, \pi_k(\omega)) \right]$$

is a $\gamma$-contraction in $Q$. Thus, for each $\omega$, it has a unique fixed point $Q^{\omega}_{k+1}$.

It remains to show that $\omega \mapsto Q^{\omega}_{k+1}$ is continuous. Let $\omega' \to \omega$. Since $Q^{\omega'}_{k+1}$ and $Q^{\omega}_{k+1}$ are fixed points of $T_{k,\omega'}$ and $T_{k,\omega}$, respectively, and all operators have the same contraction modulus $\gamma$, we have

$$\begin{aligned}
\left\| Q^{\omega'}_{k+1} - Q^{\omega}_{k+1} \right\|_{\infty} &= \left\| T_{k,\omega'} Q^{\omega'}_{k+1} - T_{k,\omega} Q^{\omega}_{k+1} \right\|_{\infty} \\
&\leq \left\| T_{k,\omega'} Q^{\omega'}_{k+1} - T_{k,\omega'} Q^{\omega}_{k+1} \right\|_{\infty} + \left\| T_{k,\omega'} Q^{\omega}_{k+1} - T_{k,\omega} Q^{\omega}_{k+1} \right\|_{\infty} \\
&\leq \gamma \left\| Q^{\omega'}_{k+1} - Q^{\omega}_{k+1} \right\|_{\infty} + \left\| T_{k,\omega'} Q^{\omega}_{k+1} - T_{k,\omega} Q^{\omega}_{k+1} \right\|_{\infty}.
\end{aligned}$$

Rearranging gives

$$\left\| Q^{\omega'}_{k+1} - Q^{\omega}_{k+1} \right\|_{\infty} \leq \frac{1}{1-\gamma} \left\| T_{k,\omega'} Q^{\omega}_{k+1} - T_{k,\omega} Q^{\omega}_{k+1} \right\|_{\infty}.$$

**Algorithm 1** Deep CMDPI for Discrete-Action Tasks

**Require:** Preference-conditioned actor: $\pi_\theta$ and its target actor $\pi_{\bar{\theta}}$, Preference-conditioned vector critic ($m$-dimension outputs): $\mathbf{Q}_\phi(s, a, \omega)$ and its target network, the deep KL coefficient $\lambda_{\mathrm{KL}}$
**for** $k = 0, 1, \cdots$ **do**
  Collect multiple episodes with fixed $\pi_\theta$; sample $\omega$ once per episode and keep it fixed within that episode.
  Update running reward statistics and the utopia-point estimate using (33).
  $\mathbf{y} = \mathbf{r} + \gamma[\mathbf{Q}_{\bar{\phi}}(s', a', \omega) - \lambda_{\mathrm{KL}}\{\log \pi_\theta(a'|s', \omega) - \log \pi_{\bar{\theta}}(a'|s', \omega)\}\mathbf{1}]$
  $L_\phi$, a sample average of $\|\mathbf{y} - \mathbf{Q}_\phi(s, a, \omega)\|_2^2$.
  $\tilde{\mathbf{V}}_\phi = \sum_{a'} \pi_\theta(a'|s, \omega)[\mathbf{Q}_\phi(s, a', \omega) - \lambda_{\mathrm{KL}}\{\log \pi_\theta(a'|s, \omega) - \log \pi_{\bar{\theta}}(a'|s, \omega)\}]$
  $L_\theta$, a sample average of $-\mathbf{g}^\top \tilde{\mathbf{V}}_\phi$ with $\mathbf{g} = \mathrm{sg}[\nabla_{\mathbf{z}} u(\mathbf{z}, \omega)|_{\mathbf{z}=\tilde{\mathbf{V}}_\phi}]$.
  update $\mathbf{Q}_\phi$ based on $L_\phi$ and $\pi_\theta$ based on the $L_\theta$.
  Update targets: $\bar{\theta} \leftarrow c\bar{\theta} + (1-c)\theta$, $\bar{\phi} \leftarrow c\bar{\phi} + (1-c)\phi$.
**end for**

**Algorithm 2** Preference-conditioned Single Actor-Critic

**Require:** Preference-conditioned actor: $\pi_\theta$ and its target actor $\pi_{\bar{\theta}}$, Preference-conditioned vector critic ($m$-dimension outputs): $\mathbf{Q}_\phi(s, a, \omega)$ and its target network, Replay buffer $\mathcal{D}$, STCH smoothing $\tau$, and entropy coefficient $\alpha_{\mathrm{ent}}$
**for** $k = 0, 1, \cdots$ **do**
  Store samples with $\pi_\theta$ into the $\mathcal{D}$
  Update running reward statistics and the utopia-point estimate using (33).
  Sample transitions from $\mathcal{D}$ and temporarily relabel each with a freshly sampled $\omega$.
  Update $\phi$ based on $L_c$ in equation (16) for the relabeled samples
  Update $\theta$ based on $L_\pi$ in equation (19) for relabeled samples from $\mathcal{D}$
  Update targets: $\bar{\theta} \leftarrow c\bar{\theta} + (1-c)\theta$, $\bar{\phi} \leftarrow c\bar{\phi} + (1-c)\phi$.
**end for**

Here the right-hand side is evaluated at the fixed vector $Q_{k+1}^{\omega}$. Therefore, it is enough to show that $\omega \mapsto T_{k,\omega}Q$ is continuous for each fixed $Q$.

This follows from the continuity of $\omega \mapsto r_k^{\omega}$, the induction hypothesis that $\omega \mapsto \pi_k(\omega)$ is continuous, and the joint continuity of $\chi(s; Q, \pi)$ in $(Q, \pi)$. Hence

$$\left\|T_{k,\omega'}Q_{k+1}^{\omega} - T_{k,\omega}Q_{k+1}^{\omega}\right\|_{\infty} \to 0 \qquad (\omega' \to \omega),$$

and consequently

$$Q_{k+1}^{\omega'} \to Q_{k+1}^{\omega}.$$

**(iii) Continuity of the multiplicative policy update.** Finally, the policy update

$$\pi_{k+1}(a \mid s; \omega) = \frac{\pi_k(a \mid s; \omega)\exp\big(Q_{k+1}^{\omega}(s, a)/\alpha_{\mathrm{KL}}\big)}{\sum_{b \in \mathcal{A}} \pi_k(b \mid s; \omega)\exp\big(Q_{k+1}^{\omega}(s, b)/\alpha_{\mathrm{KL}}\big)}$$

is a ratio of continuous functions. The denominator is strictly positive because $\pi_k(\cdot \mid s; \omega)$ has full support and the exponential term is strictly positive. Therefore, $\omega \mapsto \pi_{k+1}(\omega)$ is continuous. Moreover, the same update preserves full support, so the induction continues. This proves the claim for all finite $k$. □

## H Implementation Details of Preference-Conditioned Single Actor-Critic

Both algorithms update the target actor and target critic by exponential averaging at the end of each loop, with the shared coefficient $c = 0.995$ in Table 3. The deep KL coefficient $\lambda_{\mathrm{KL}}$ and the tabular prox scale $\alpha_{\mathrm{KL}} = L_{\mathrm{rel}}(1-\gamma)$ denote distinct quantities.

**Base implementations.** We implement both the model-free (MF) and model-based (MB) variants on top of open-source codebases. As MF baseline, we start from Soft Actor-Critic (SAC) [31]. As

MB baseline, we start from the DHMBPO implementation [42], which combines MBPO [43] with SVG-style policy optimization [44] and uses deep ensembles for learned dynamics. Importantly, the DHMBPO codebase already contains a SAC implementation; hence, we keep the network architectures (actor/critic), optimizer choices, and most training hyperparameters identical across MF/MB, and only change the components required for multi-objective preference-conditioning and scalarization.

**Preference-conditioned actor and vector critic.** We modify the actor and critic networks so that they are conditioned on a preference vector $\omega$. The actor is $\pi_\theta(a \mid s, \omega)$, and the critic outputs an $m$-dimensional vector $\mathbf{Q}_\phi(s, a, \omega) \in \mathbb{R}^m$. In our MuJoCo experiments, $m \in \{2, 3\}$ (see Appendix J.2.2 for the environment list).

**Preference sampling during training.** During real-environment data collection, we sample $\omega$ at the beginning of each episode and keep it fixed within the episode, including in Algorithm 1. In Algorithm 2, each time a transition is sampled from replay, we temporarily replace its preference by a fresh $\omega$ and use the value for the associated actor, critic, and target-critic evaluations; the stored transition is unchanged. Concretely, $\omega$ was sampled using PyTorch `RelaxedOneHotCategorical` with temperature 1 and uniform class probabilities $\mathbf{1}/m$. This sampling can yield near-zero (and in principle zero) coordinates, but we did not observe numerical errors in our experiments.

**Vector SAC targets.** Given a mini-batch $(s, a, \mathbf{r}, s')$ from the replay buffer and a sampled preference $\omega$, we compute the target action $a' \sim \pi_\theta(\cdot \mid s', \omega)$ and define the vector target

$$\mathbf{y} = \mathbf{r} + \gamma\big[\mathbf{Q}_{\bar{\phi}}(s', a', \omega) - \alpha_{\text{ent}} \log \pi_\theta(a' \mid s', \omega)\, \mathbf{1}\big], \tag{30}$$

where $\mathbf{1} \in \mathbb{R}^m$ broadcasts the entropy term to each reward dimension. We use the squared loss $L_\phi = \frac{1}{2}\|\mathbf{y} - \mathbf{Q}_\phi(s, a, \omega)\|_2^2$.

**Scalarized actor update via utility gradients.** To update the actor, we convert the vector critic output into a scalar objective using the gradient of the utility. Let $\tilde{a} \sim \pi_\theta(\cdot \mid s, \omega)$ and $\tilde{\mathbf{V}}_\phi = \mathbf{Q}_\phi(s, \tilde{a}, \omega) - \alpha_{\text{ent}} \log \pi_\theta(\tilde{a} \mid s, \omega)\mathbf{1}$. We compute

$$\mathbf{g} = \nabla_{\mathbf{z}} u(\mathbf{z}, \omega)\big|_{\mathbf{z}=\tilde{\mathbf{V}}_\phi}, \tag{31}$$

and use the following actor loss:

$$L_\theta = -\,\text{sg}[\mathbf{g}]^\top \tilde{\mathbf{V}}_\phi. \tag{32}$$

For linear scalarization, $\mathbf{g} = \omega$ and this recovers the standard linear weighted objective. For the proposed STCH scalarization, $\mathbf{g}$ depends on $\mathbf{z}$ and $\omega$ through $u$. We treat $\mathbf{g}$ as a constant with respect to $\theta$ (the stop-gradient operator sg), which avoids second-order derivatives. Because $\mathbf{g}$ and the differentiated factor are evaluated at the same entropy-adjusted vector, the resulting sample direction is exactly $\nabla_\theta u(\tilde{\mathbf{V}}_\phi, \omega)$; averaging samples gives the surrogate $\widetilde{U}(\theta)$ defined in Section 3.5, not the tabular STCH objective. The training loops for deep CMDPI and PCSAC are summarized in Algorithms 1 and 2, respectively.

**Reward normalization and utopia-point estimation.** We normalize rewards to stabilize training across objectives and environments. At each policy optimization step, we compute the sample mean and standard deviation of *all* real-environment reward samples collected up to that time, and normalize each reward component in the sampled mini-batch accordingly. Using the normalized mini-batch rewards, we maintain an online estimate of the utopia point $\mathbf{I} \in \mathbb{R}^m$ as follows: we compute the 99th percentile (component-wise) of the normalized rewards within the mini-batch, multiply it by $1/(1-\gamma)$, and then update $\mathbf{I}$ by exponential averaging

$$\mathbf{I} \leftarrow (1-\beta)\mathbf{I} + \beta\,\widehat{\mathbf{I}}, \qquad \beta = 3 \times 10^{-4}, \tag{33}$$

with initialization $\mathbf{I} = \mathbf{1}$. Similarly, we compute the 1st percentile and apply the same averaging update to obtain a lower bound vector. When computing the critic targets, we clip the target critic output using these bounds (this clipping is inherited from the DHMBPO implementation).

**STCH smoothing and entropy parameters.** PCSAC introduces the STCH log-sum-exp smoothing parameter $\tau$, and we test $\tau \in \{0.1, 0.01, 0.001\}$. We fix SAC's entropy coefficient $\alpha_{\text{ent}}$ to $0.01$ for all environments, motivated by the fact that we normalize rewards (and thus avoid environment-specific tuning of $\alpha_{\text{ent}}$). This coefficient is unrelated to $\alpha_{\text{KL}} = L_{\text{rel}}(1-\gamma)$ in exact CMDPI. Because the actor uses the unnormalized direction $\mathbf{g} = \nabla u$, scalarizing the broadcast entropy term produces an effective sample-wise coefficient $\alpha_{\text{ent}}\mathbf{g}^\top \mathbf{1} \in [\alpha_{\text{ent}}\omega_{\min}, \alpha_{\text{ent}}\omega_{\max}]$.

**Hyperparameters.** All remaining hyperparameters are shared with DHMBPO [42] unless otherwise stated. Table 3 lists the values used in our experiments.

# I Scope of the critic-local utility gradient

Fix a preference $\omega$ and write $\mathbf{g}(\mathbf{z}) := \nabla_{\mathbf{z}} u(\mathbf{z}, \omega)$. The STCH Hessian calculation in (24) gives

$$\|\mathbf{g}(\mathbf{z}) - \mathbf{g}(\mathbf{z}')\|_\infty \le L_u \|\mathbf{z} - \mathbf{z}'\|_1, \qquad L_u := \frac{\omega_{\max}^2}{4\tau}. \tag{34}$$

Freeze $\phi$, $\mathbf{I}$, and the replay distribution during the actor update, set $\pi = \pi_\theta$, and define

$$\tilde{\mathbf{V}}_\phi(s,a,\omega) := \mathbf{Q}_\phi(s,a,\omega) - \alpha_{\text{ent}} \log \pi(a|s,\omega)\mathbf{1}.$$
$$\nu(s,a) := d_{\mathcal{D}}(s)\pi_\theta(a \mid s,\omega), \qquad \xi_\pi(s,a) := p_0(s)\pi(a \mid s,\omega), \qquad \mathbf{J}_\nu^\pi := \mathbb{E}_\nu[\mathbf{Q}^\pi].$$

Here $\mathbf{Q}^\pi$ denotes the unregularized vector action value and $d_{\mathcal{D}}$ the replay-state distribution, so $\mathbf{J}(\pi) = \mathbb{E}_{\xi_\pi}[\mathbf{Q}^\pi]$. For a bounded scalar function $h$, define $\operatorname{sp}(h) := \sup h - \inf h$, and let $\operatorname{TV}(P,Q) := \frac{1}{2}\|P - Q\|_1$.

**Proposition I.1** (Evaluation, nonlinear-aggregation, and distribution decomposition)**.** *If $\mathbf{Q}^\pi$ is bounded and $\mathbb{E}_\nu\|\tilde{\mathbf{V}}_\phi - \mathbf{Q}^\pi\|_1 < \infty$, then*

$$\begin{aligned} &\mathbb{E}_\nu[\mathbf{g}(\tilde{\mathbf{V}}_\phi)] - \mathbf{g}(\mathbf{J}(\pi)) \\ &= \underbrace{\mathbb{E}_\nu[\mathbf{g}(\tilde{\mathbf{V}}_\phi) - \mathbf{g}(\mathbf{Q}^\pi)]}_{A_{\text{eval}}} + \underbrace{\mathbb{E}_\nu[\mathbf{g}(\mathbf{Q}^\pi)] - \mathbf{g}(\mathbf{J}_\nu^\pi)}_{A_{\text{nonlin}}} + \underbrace{\mathbf{g}(\mathbf{J}_\nu^\pi) - \mathbf{g}(\mathbf{J}(\pi))}_{A_{\text{dist}}}, \end{aligned} \tag{35}$$

*and*

$$\begin{aligned} &\|\mathbb{E}_\nu[\mathbf{g}(\tilde{\mathbf{V}}_\phi)] - \mathbf{g}(\mathbf{J}(\pi))\|_\infty \\ &\le L_u\Big[\mathbb{E}_\nu\|\tilde{\mathbf{V}}_\phi - \mathbf{Q}^\pi\|_1 + \mathbb{E}_\nu\|\mathbf{Q}^\pi - \mathbf{J}_\nu^\pi\|_1 + \Big(\sum_{i=1}^m \operatorname{sp}(Q_i^\pi)\Big)\operatorname{TV}(\nu,\xi_\pi)\Big]. \end{aligned} \tag{36}$$

*Proof.* Equation (35) follows by adding and subtracting $\mathbb{E}_\nu[\mathbf{g}(\mathbf{Q}^\pi)]$ and $\mathbf{g}(\mathbf{J}_\nu^\pi)$. Jensen's inequality for the norm and (34) bound the first two terms by the first two quantities in (36). For the last term, (34) gives $L_u\|\mathbf{J}_\nu^\pi - \mathbf{J}(\pi)\|_1$. Componentwise, the span–TV inequality yields $|\mathbb{E}_\nu Q_i^\pi - \mathbb{E}_{\xi_\pi} Q_i^\pi| \le \operatorname{sp}(Q_i^\pi)\operatorname{TV}(\nu,\xi_\pi)$; summing over $i$ completes the proof. □

The evaluation term includes approximation error and the entropy-related difference between $\tilde{\mathbf{V}}_\phi$ and the raw-return $\mathbf{Q}^\pi$; it need not vanish for an exact entropy-regularized critic. Even if the evaluation and distribution terms vanish, nonlinear aggregation generally remains. A sample average converges to $\mathbb{E}_\nu[\mathbf{g}(\tilde{\mathbf{V}}_\phi)]$, not generally to $\mathbf{g}(\mathbf{J}(\pi))$. For linear scalarization, $\mathbf{g} \equiv \omega$ and all three terms vanish. Estimating the inner expectation at a fixed preference from mixed-preference replay remains open. None of these observations changes the exact tabular results.

# J Detail of Numerical Experiments

## J.1 Planning experiment details for the toy tabular MOMDP

We report implementation details of the planning-based comparison in the toy two-objective MOMDP used in Fig. 2. All methods are executed in the *planning* setting: the transition kernel $P$ and reward functions $\mathbf{r}$ are fully known, the horizon is infinite with discount $\gamma = 0.8$, and all quantities (e.g., $\mathbf{J}(\pi)$) are computed exactly by matrix operations. The policies were initialized $\pi_0(a|s) = 1/|\mathcal{A}| > 0$ for all $s, a$ and preference $\omega$. Therefore, no sampling-based learning is involved.

We summarize the CMDPI algorithm in Algorithm 3.

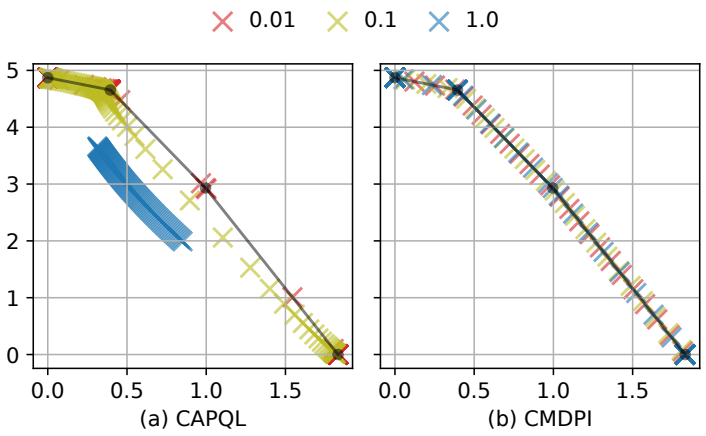



Figure 2: Scatter plots of the converged objective vectors in a two-objective MOMDP, obtained by (a) the regularized method CAPQL [17] and (b) the proposed method CMDPI. The black solid curve indicates the Pareto front, and the black dots indicate its vertices, which are recovered by the linear-utility baseline. The legend is shared across panels, but the symbols have different roles: CAPQL sweeps its uniform-reference entropy coefficient $\alpha_{\mathrm{CAPQL}}$, whereas CMDPI sweeps STCH smoothing $\tau$ and uses the separate empirical coupling $\alpha_{\mathrm{KL}} = 1/\tau$ for its prox step. CAPQL exhibits a trade-off between achieving a dense front and accurately recovering a subset of Pareto-optimal points. In contrast, our method recovers Pareto-optimal solutions accurately and more uniformly (visually denser) across all tested hyperparameter values.

**Algorithm 3** Concave Mirror Descent Policy Iteration

**Require:** preference $\omega \in \mathbb{R}^m_{++}$, STCH smoothing $\tau > 0$, KL/prox scale $\alpha_{\mathrm{KL}} > 0$, initial policy $\pi_0$ with full support.
**for** $k = 0, 1, \ldots, K-1$ **do**
  Compute/estimate $\mathbf{J}(\pi_k)$. (objective evaluation)
  **Dynamic scalarization weight:** $\mathbf{g}_k = \nabla_{\mathbf{J}} u\big(\mathbf{J}(\pi_k), \omega\big) \in \mathbb{R}^m_{++}$.
  **Scalar reward:** $r_k(s,a) = \langle \mathbf{g}_k, \mathbf{r}(s,a) \rangle$.
  **Soft policy evaluation:** solve the fixed point (13) to obtain $Q_{k+1}$.
  **Policy improvement:** update $\pi_{k+1}$ by (14).
**end for**

**Environment.** We consider a discounted MOMDP with $\mathcal{S} = \{0,1,2,3\}$ and $\mathcal{A} = \{0,1\}$. The initial distribution is uniform: $p_0(s) = 1/|\mathcal{S}|$ for all $s$. The transition kernel is specified by two row-stochastic matrices $P^a \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ where $P^a_{s,s'} = P(s' \mid s, a)$. For action $a = 0$,

$$P^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

and for action $a = 1$,

$$P^1 = \begin{bmatrix} 0.1 & 0.9 & 0 & 0 \\ 0.1 & 0 & 0.9 & 0 \\ 0 & 0.1 & 0 & 0.9 \\ 0 & 0 & 0.1 & 0.9 \end{bmatrix}.$$

Intuitively, action 0 moves "left" deterministically (with reflection at the left boundary), and action 1 moves "right" with probability 0.9 (and "left" with probability 0.1), with reflection at the right boundary.

The two-dimensional reward vector is $\mathbf{r}(s,a) = (r_1(s,a), r_2(s,a)) \in \mathbb{R}^2$ and is given by

$$\mathbf{r}(s,0) = \begin{cases} (0,0) & s=0 \\ (0,2) & s=1 \\ (0,1.6) & s=2 \\ (1,0) & s=3 \end{cases} \qquad \mathbf{r}(s,1) = \begin{cases} (0,0) & s=0 \\ (0,0) & s=1 \\ (0,0) & s=2 \\ (0,1) & s=3. \end{cases}$$

This parameterization is chosen so that the attainable objective set is visibly nontrivial and the Pareto boundary is not axis-aligned.

**Preference vectors and STCH utopia point.** We use 100 preference vectors on the simplex, equally spaced and including endpoints:

$$\omega^{(i)} = \Big(\frac{i}{99}, 1 - \frac{i}{99}\Big), \quad i = 0, 1, \ldots, 99.$$

Although the theory often assumes $\omega \in \mathbb{R}^2_{++}$, in this environment the Pareto boundary has no segments parallel to coordinate axes, so including endpoints $\omega^{(0)} = (0,1)$ and $\omega^{(99)} = (1,0)$ does not introduce degeneracy in practice.

For the STCH utility, we set the utopia point componentwise as

$$I_\ell = \max_{s \in \mathcal{S},\, a \in \mathcal{A}} \left| \frac{r_\ell(s,a)}{1-\gamma} \right|, \quad \ell \in \{1,2\}.$$

With $\gamma = 0.8$ and the above rewards, this yields $I_1 = \frac{1}{0.2} = 5$ and $I_2 = \frac{2}{0.2} = 10$.

**Compared planning methods.** For each preference vector $\omega^{(i)}$, we independently run the following methods and obtain a stationary policy $\pi$.

*(i) Linear scalarization + value iteration (baseline points/curve).* We define the scalar reward $r_\omega(s,a) = \langle \omega, \mathbf{r}(s,a) \rangle$ and solve $\max_{\pi \in \Pi} \mathbb{E}_{s \sim p_0}[V^\pi_\omega(s)]$ by standard discounted value iteration in the tabular model. Let $\pi^{\mathrm{lin}}_\omega$ be an optimal policy and $\mathbf{J}(\pi^{\mathrm{lin}}_\omega)$ its (vector) objective. The black dots in Fig. 2 indicate the distinct objective vectors obtained by this linear-scalarization planning (which correspond to supported extreme points in this toy problem). The black solid curve is drawn by sorting these points along the boundary and connecting them by piecewise-linear interpolation.

*(ii) CAPQL.* We implement the CAPQL planning update of Lu et al. [17] in the tabular setting with known $P$ and $\mathbf{r}$, and iterate the associated operator until convergence to obtain a stationary policy. We sweep three uniform-reference entropy coefficients $\alpha_{\mathrm{CAPQL}}$ (as shown in Fig. 2).

*(iii) Proposed CMDPI under STCH utility.* We run the proposed mirror-descent policy iteration algorithm for maximizing the STCH utility $u(\mathbf{J}(\pi), \omega)$ with smoothing parameter $\tau > 0$. For this planning visualization, we set the CMDPI prox parameter as $\alpha_{\mathrm{KL}} = 1/\tau$. This is an empirical coupling used for the sweep, not the theoretical identity $\alpha_{\mathrm{KL}} = L_{\mathrm{rel}}(1-\gamma)$, because $L_{\mathrm{rel}}$ is only a conservative sufficient relative-smoothness constant. We sweep three values of $\tau$ (as shown in Fig. 2).

For CAPQL and CMDPI, we use the uniform initial policy specified above and then run deterministic planning iterations. This full-support, preference-independent initialization satisfies the continuity assumption of Proposition G.1.

**Exact evaluation of objective vectors.** For every returned stationary policy $\pi$, we compute its objective vector $\mathbf{J}(\pi)$ exactly (no rollouts). Let $P_\pi \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$ denote the induced transition matrix $P_\pi(s,s') = \sum_a \pi(a \mid s) P(s' \mid s,a)$. We compute the discounted state occupancy $\rho_\pi \in \mathbb{R}^{|\mathcal{S}|}$ as the unique solution of

$$\rho_\pi^\top = (1-\gamma) p_0^\top + \gamma \rho_\pi^\top P_\pi,$$

and then form the discounted occupancy measure $\mu_\pi(s,a) = \rho_\pi(s)\pi(a \mid s)$. Finally, we compute

$$\mathbf{J}(\pi) = \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \mu_\pi(s,a)\, \mathbf{r}(s,a),$$

which matches the definition used throughout the paper.

**Visualization.** For each method and each hyperparameter setting (three values for CAPQL and three values for the proposed method), we plot the 100 exact objective vectors $\{\mathbf{J}(\pi_{\omega^{(i)}})\}_{i=0}^{99}$ as a scatter plot, where different colors correspond to different hyperparameters. The black dots and the black solid line are produced by linear scalarization + value iteration as described above.

### J.2 Experimental Protocol and Baselines on MO-Gymnasium

#### J.2.1 Experimental Protocol for Discrete-Action MO-Gymnasium Tasks

**Environments.** We evaluate on eight MO-Gymnasium [27] tasks with discrete action spaces: `deep-sea-treasure-concave-v0` (DST-Concave), `deep-sea-treasure-v0` (DST), `fishwood-v0` (FishWood), `four-room-v0` (FourRoom), `fruit-tree-v0` (Fruit), `minecart-v0` (Minecart), `mo-lunar-lander-v3` (Lunar), and `mo-reacher-v5` (Reacher). The reward dimension ranges from two to six. We use the default episode-length settings of MO-Gymnasium and train each method for 2M environment interaction steps.

**Baselines and codebases.** We compare LS (linear scalarization), CAPQL [17], PreCo [34], C-MORL [9], PCSAC, and CMDPI. LS, CAPQL, PreCo, PCSAC, and CMDPI are run within the same PreCo's official codebase to reduce implementation-level confounders. PCSAC and CMDPI are our STCH-based preference-conditioned actor–critic variants. For the PreCo-based implementations, we apply a common per-objective running reward normalization in critic learning, using the empirical mean and standard deviation of collected reward vectors. This normalization is used only during training; all reported return vectors and metrics are computed from unnormalized episodic returns.

C-MORL is included as a strong multi-policy Pareto-front discovery baseline, although it is not a single preference-conditioned policy. We use the official public implementation and modify the interaction accounting and logging needed to align the training budget and post-hoc metric computation with the other methods.

**Training seeds.** We run eight random seeds for each method and environment.

**Evaluation and metric computation.** At each evaluation checkpoint, we collect unnormalized episodic return vectors from each method and save them for post-hoc evaluation. For preference-conditioned methods, we perform preference-sweep evaluation by conditioning the policy on a fixed rollout preference set and recording the resulting return vectors. To compute a fixed set of preference vectors for evaluation, we ladapted the `utils/test.py` script from the official PreCo implementation J.9, with modifications to support arbitrary numbers of objective $m \geq 2$. For C-MORL, which outputs a set of policies rather than a single preference-conditioned policy, we evaluate the policies in the discovered set to obtain return vectors.

We do not use the metric values logged by the original C-MORL implementation. Instead, after all runs are completed, we compute HV, EUM, and SP for all methods from the saved unnormalized return vectors using the same post-processing script. HV is computed with a common environment-wise reference point. For each evaluation preference, we compute the maximum weighted sum over the evaluated return vectors. EUM is the average of these maxima over a fixed preference set shared by all methods.

**Evaluation preferences.** We distinguish between two preference sets. The rollout preference set is used as input to preference-conditioned policies during evaluation rollouts and therefore applies only to LS, CAPQL, PreCo, PCSAC, and CMDPI. The EUM evaluation preference set is used only for metric aggregation and is shared by all methods, including C-MORL. For C-MORL, the EUM preferences are used only at the aggregation stage because C-MORL returns a policy set rather than a preference-conditioned policy. All preference sets are generated by the deterministic simplex-grid routine used in PreCo-based codebase and are fixed across methods, checkpoints, and seeds.

**Reference points.** We distinguish between reference points used for reported HV computation and reference values used internally by C-MORL. For reported HV, we use a common environment-wise reference point shared by all methods. Following the same protocol as in the continuous-action experiments, we determine this reference point a posteriori as the component-wise minimum over the unnormalized evaluation return vectors collected across all methods, checkpoints, and random seeds.

C-MORL additionally uses internal reference values in its online multi-policy selection procedure. These internal values are used only during C-MORL training and are never used to compute the reported HV, EUM, or SP values. When the official C-MORL implementation provides environment-specific internal values, we keep them; otherwise, we use a conservative lower-bound vector with all entries set to $-10^4$.

**Counting environment interactions.** We count only environment interactions used for training and exclude evaluation rollouts for all methods. For C-MORL, we count interactions across parallel sampling workers when necessary, so that the logged sampling steps reflect the actual number of environment transitions used for training. This adjustment affects only sample-efficiency accounting.

**Hyperparameters.** PCSAC uses the off-policy replay-buffer optimization scheme with entropy coefficient $\alpha_{\text{ent}} = 0.3$ tuned out of $\{0.03, 0.1, 0.3\}$. CMDPI uses the on-policy optimization scheme motivated by the KL-regularized formulation, with tuned smoothness/KL parameter 0.001 out of $\{0.1, 0.01, 0.001\}$ and the shared target-network averaging coefficient $c = 0.995$ in Table 3. For numerical stability of the KL-regularized actor update, the CMDPI actor optimizer uses Adam with weight decay $10^{-4}$. For C-MORL, we tune the fraction of the total 2M-step budget allocated to the initialization stage 0.4 out of $\{0.2, 0.4, 0.5\}$. All other shared optimizer and network hyperparameters in the PreCo-based implementations follow the default settings of the codebase.

### J.2.2 Experimental Protocol for Continuous-Action MO-Gymnasium Tasks

**Environments.** We evaluate our method on eight continuous-control multi-objective MuJoCo environments from MO-Gymnasium (all v5). All tasks are configured with two or three reward dimensions. Specifically, we use `mo-halfcheetah-v5`, `mo-hopper-v5`, `mo-swimmer-v5`, `mo-walker2d-v5`, and truncated-observation variants of Ant and Humanoid, denoted `TruncAnt` and `TruncHumanoid`, following the standard "truncated state" setting used in previous model-based RL works [45, 43, 42]. For the Pareto-front visualization experiments, we use two-objective versions of `mo-hopper-v5` and `TruncAnt` by removing one reward dimension from their original reward vectors.

**Baselines and codebases.** Besides our MF/MB baselines built on SAC/DHMBPO, we run additional state-of-the-art MORL baselines using their publicly available implementations. We note that CAPQL and our CAPQL-MF baseline are conceptually similar (both are preference-conditioned actor-critic with linear scalarization), but differ in codebase and in our additional reward normalization described in Appendix H.

**Training seeds.** For our baselines and proposed method, we run $8$ random seeds for each environment.

**HV reference point.** We set for each environments reference point by taking the minimum expected reward for each dimension over all runs; all over the methods and preference-sweep evaluation steps and seeds.

**Evaluation schedule (preference-sweep evaluation).** During training, we periodically perform *preference-sweep evaluation*. We evaluate every $25{,}000$ real-environment interaction steps by default. Some external implementations cannot be paused at exactly multiples of $25{,}000$ steps; in those cases, we evaluate at the first available checkpoint after each multiple. For plotting sample-efficiency curves, we post-process the logged evaluations by binning them into $25{,}000$-step bins:

**Evaluation preferences.** For each environment with $m = 2$ objectives, we pre-generate 20 approximately equally spaced preference vectors on the simplex using `equally_spaced_weights` from MORL-Baselines [27]. For environments with $m = 3$ objectives, we use $20^{3-1} = 400$ preference vectors on the simplex. At a given training checkpoint and seed, for each preference vector $\omega$, we condition the actor on $\omega$ and roll out the policy. We record the undiscounted episodic return vectors and average them over episodes to obtain an estimate of the expected return vector for that $(\text{checkpoint}, \text{seed})$. EUM is computed by the same procedure as in Appendix J.2.1.

**Counting environment interactions (sample-efficiency).** To compare sample-efficiency fairly across methods, we count *all* real-environment interaction steps used for training, excluding only

Table 3: Hyperparameters for our SAC/DHMBPO-based implementations (CAPQL-MF/MB and PCSAC-MF/MB). From top to bottom: (i) the PCSAC-specific parameter shared by our MF/MB variants, (ii) hyperparameters shared across our MF and MB implementations, and (iii) MB-only hyperparameters shared across our MB implementations.

| Hyper-parameter | Value |
|---|---|
| STCH smoothing $\tau$ | $\{0.1, 0.01, 0.001\}$ |
| Discount factor | 0.995 |
| Seed steps | 5000 |
| Batch size | 256 |
| Update-to-data ratio | 1 |
| Replay buffer size | 1M |
| Learning rate for the actor and critics | $3 \cdot 10^{-4}$ |
| Fixed entropy coefficient $\alpha_{\text{ent}}$ | 0.01 |
| Target-network averaging coefficient $c$ | 0.995 |
| Ensemble size of critic | 5 |
| Length of distribution rollout | 20 |
| Length of training rollout | 5 |
| Ensemble size of model | 8 |
| Optimizer for training model | AdamW [46] |
| Learning rate for model | $1 \cdot 10^{-3}$ |

Table 4: Average ranks on eight discrete-action MO-Gymnasium tasks (lower is better). CMDPI denotes the deep actor–critic implementation. Each entry is the average rank followed by $\pm$ the 95% bootstrap confidence interval from stratified sampling over tasks and eight seeds. Best and second-best results are highlighted in bold and gray, respectively.

| | LS | CAPQL | PreCo | C-MORL | PCSAC | CMDPI |
|---|---|---|---|---|---|---|
| HV | $4.25 \pm 0.36$ | $4.12 \pm 0.33$ | $3.75 \pm 0.54$ | $4.12 \pm 0.69$ | $3.00 \pm 0.72$ | $\mathbf{1.75 \pm 0.33}$ |
| EUM | $3.88 \pm 0.53$ | $3.50 \pm 0.49$ | $3.38 \pm 0.57$ | $3.62 \pm 0.67$ | $3.50 \pm 0.74$ | $\mathbf{3.12 \pm 0.54}$ |
| SP | $\mathbf{2.25 \pm 0.35}$ | $3.75 \pm 0.44$ | $\mathbf{2.25 \pm 0.46}$ | $4.62 \pm 0.69$ | $4.38 \pm 0.62$ | $3.75 \pm 0.47$ |

evaluation rollouts. For model-based methods, imaginary rollouts inside learned models are *not* counted. We also do not count gradient update steps or replay-buffer sampling frequency (i.e., the update-to-data ratio does not change the step count).

**Accounting for additional interaction phases in baselines.** Some baselines use extra real-environment interaction phases beyond standard off-policy training. For GPI-PD [36], we include the real-environment interactions used to identify the optimal preference vector as part of the sample count. For PD-MORL [8], the method relies on a *key preference set* for preference alignment: it defines key preferences as the unit vectors (one per objective) plus the uniform preference (e.g., for $m = 2$: $\{[1, 0], [0.5, 0.5], [0, 1]\}$), trains using fixed key preferences to obtain corresponding solutions, and uses them to normalize the solution space and fit an interpolator that projects preferences for alignment. We follow the official PD-MORL procedure and count the real-environment interactions used in this key-preference phase (200K steps in our setup) as part of the sample count.

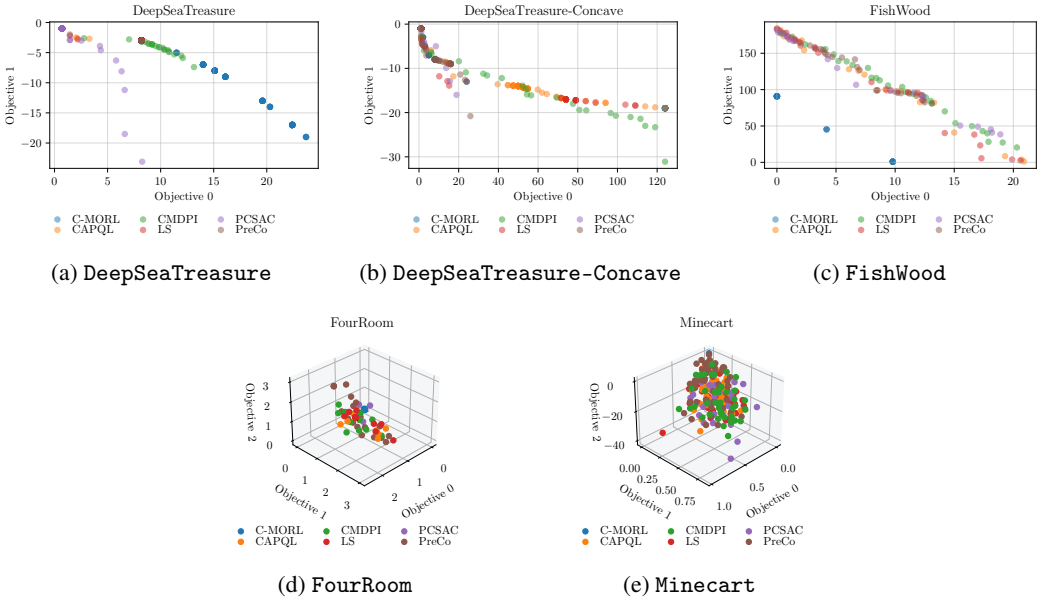
**Hyperparameters.** Table 3 lists the hyperparameters shared across our MF/MB baselines and the proposed method. All unspecified parameters follow DHMBPO [42].

### J.3 Aggregate-rank summaries

We show aggregate-ranks on discrete-action MO-Gymnasium tasks by Table 4 and one on continuous-action MO-Gymnasium tasks by Table 5.

Table 5: Average ranks on eight continuous-action MO-Gymnasium tasks (lower is better). Model-based methods are evaluated at 500K real-environment steps and model-free methods at 2M steps. Each entry is the average rank followed by $\pm$ the 95% bootstrap confidence interval over eight seeds and tasks.

| | Model-Free Methods | | | | Model-Based Methods | | |
|---|---|---|---|---|---|---|---|
| Metrics | PD-MORL | COLA | CAPQL-MF | PCSAC-MF | GPI-PD | CAPQL-MB | PCSAC-MB |
| HV | $3.25 \pm 0.32$ | $\mathbf{2.00 \pm 0.41}$ | $2.75 \pm 0.38$ | $\mathbf{2.00 \pm 0.33}$ | $3.00 \pm 0.01$ | $1.75 \pm 0.17$ | $\mathbf{1.25 \pm 0.17}$ |
| EUM | $3.25 \pm 0.33$ | $\mathbf{2.00 \pm 0.40}$ | $2.62 \pm 0.36$ | $2.12 \pm 0.37$ | $3.00 \pm 0.01$ | $1.75 \pm 0.17$ | $\mathbf{1.25 \pm 0.17}$ |
| SP | $\mathbf{1.38 \pm 0.18}$ | $2.88 \pm 0.44$ | $3.25 \pm 0.25$ | $2.50 \pm 0.32$ | $\mathbf{1.75 \pm 0.28}$ | $2.00 \pm 0.28$ | $2.25 \pm 0.28$ |




(a) `DeepSeaTreasure` (b) `DeepSeaTreasure-Concave` (c) `FishWood`

(d) `FourRoom` (e) `Minecart`

Figure 3: Final-step return scatter plots for five two- or three-objective tasks.

## J.4 Detailed Results

### J.4.1 Discrete action tasks

Figure 3 present the final-step return scatter plots for five two- or three-objective tasks. In addition, Table 6 reports task-wise metrics for the eight discrete-action MO-Gymnasium benchmarks.

### J.4.2 Continuous action tasks

We show, from Figure 4 to Figure 7, for both of model-free and model-based methods, the sample efficiency curve with respect to the Hypervolume and EUM, and in Figure 8 and Figure 9, final-step return scatters on 2-dimensional six tasks. Additionally, we summarize quantitative metrics at Final step metrics in Table 7.

## J.5 Sensitivity analysis

### J.5.1 Utopia point estimation

Since utopia-point estimation is one of the main practical gaps in the deep setting, we ran an additional sensitivity study by replacing the estimated utopia point $\hat{I}$ with $\lambda\hat{I}$ on 8 discrete-action MO-Gymnasium tasks. This perturbs only the utopia-point estimate, by $\Delta I = (\lambda - 1)\hat{I}$. Across the tested range in Tables 8–9, neither CMDPI nor PCSAC collapses on either side of $\lambda = 1$; the best scale varies by metric. For the exact problem, any over-estimate that remains componentwise above $\mathcal{J}$ satisfies Proposition A.3 and therefore preserves full-front coverage under its polyhedral assumptions,

Table 6: Comparison of final-step EUM, HV, and sparsity on discrete-action tasks from MO-Gymnasium. Entries are means ± standard errors of the mean (SEM) over eight seeds.

| Environments | Metrics | LS | CAPQL | PreCo | C-MORL | PCSAC | CMDPI |
|---|---|---|---|---|---|---|---|
| DST | EUM ↑ ($10^0$) | $2.60 \pm 0.00$ | $2.60 \pm 0.00$ | $2.60 \pm 0.00$ | $\mathbf{6.68 \pm 0.10}$ | $-41.83 \pm 6.10$ | $3.18 \pm 0.10$ |
| | HV ↑ ($10^3$) | $0.80 \pm 0.00$ | $0.80 \pm 0.00$ | $0.80 \pm 0.00$ | $\mathbf{1.99 \pm 0.03}$ | $0.01 \pm 0.01$ | $1.01 \pm 0.04$ |
| | SP ↓ ($10^0$) | $\mathbf{0.00 \pm 0.00}$ | $\mathbf{0.00 \pm 0.00}$ | $\mathbf{0.00 \pm 0.00}$ | $65.69 \pm 4.79$ | $0.27 \pm 0.27$ | $0.29 \pm 0.08$ |
| DST-Concave | EUM ↑ ($10^1$) | $0.33 \pm 0.99$ | $2.05 \pm 0.80$ | $0.08 \pm 0.08$ | $3.36 \pm 0.30$ | $-3.03 \pm 0.92$ | $\mathbf{4.15 \pm 0.48}$ |
| | HV ↑ ($10^3$) | $2.01 \pm 1.28$ | $4.30 \pm 1.55$ | $0.36 \pm 0.26$ | $6.74 \pm 0.51$ | $0.18 \pm 0.15$ | $\mathbf{8.31 \pm 0.84}$ |
| | SP ↓ ($10^{-1}$) | $56.45 \pm 45.70$ | $374.15 \pm 214.38$ | $\mathbf{7.61 \pm 7.61}$ | $12369.38 \pm 1051.80$ | $145.97 \pm 145.97$ | $1260.36 \pm 458.87$ |
| FishWood | EUM ↑ ($10^1$) | $9.16 \pm 0.02$ | $9.16 \pm 0.04$ | $7.93 \pm 0.10$ | $4.59 \pm 0.01$ | $\mathbf{9.22 \pm 0.02}$ | $7.65 \pm 0.48$ |
| | HV ↑ ($10^3$) | $0.56 \pm 0.19$ | $0.35 \pm 0.14$ | $0.77 \pm 0.05$ | $0.05 \pm 0.03$ | $0.94 \pm 0.06$ | $\mathbf{1.56 \pm 0.05}$ |
| | SP ↓ ($10^1$) | $134.80 \pm 60.33$ | $233.61 \pm 118.24$ | $\mathbf{4.08 \pm 1.06}$ | $176.73 \pm 16.74$ | $612.73 \pm 176.80$ | $36.26 \pm 6.93$ |
| Fruit | EUM ↑ ($10^0$) | $4.27 \pm 0.02$ | $4.30 \pm 0.02$ | $4.75 \pm 0.08$ | $4.41 \pm 0.08$ | $\mathbf{4.99 \pm 0.01}$ | $4.85 \pm 0.02$ |
| | HV ↑ ($10^4$) | $0.53 \pm 0.01$ | $0.53 \pm 0.01$ | $1.42 \pm 0.13$ | $0.09 \pm 0.02$ | $\mathbf{1.55 \pm 0.01}$ | $1.52 \pm 0.02$ |
| | SP ↓ ($10^{-4}$) | $7.07 \pm 0.34$ | $11.97 \pm 5.22$ | $6.78 \pm 0.70$ | $194325.24 \pm 10559.86$ | $\mathbf{3.19 \pm 0.39}$ | $4.05 \pm 0.44$ |
| Lunar | EUM ↑ ($10^1$) | $1.21 \pm 0.06$ | $\mathbf{1.24 \pm 0.05}$ | $1.15 \pm 0.05$ | $-0.44 \pm 0.63$ | $-0.11 \pm 0.15$ | $-0.86 \pm 0.08$ |
| | HV ↑ ($10^{10}$) | $0.00 \pm 0.00$ | $0.47 \pm 0.30$ | $0.46 \pm 0.25$ | $2.91 \pm 0.90$ | $\mathbf{3.77 \pm 0.24}$ | $0.78 \pm 0.00$ |
| | SP ↓ ($10^1$) | $\mathbf{1.38 \pm 1.38}$ | $172.98 \pm 124.98$ | $14.97 \pm 3.63$ | $13598.59 \pm 7974.63$ | $44.10 \pm 4.56$ | $18.57 \pm 2.91$ |
| Minecart | EUM ↑ ($10^{-1}$) | $-64.51 \pm 0.58$ | $-64.35 \pm 0.98$ | $\mathbf{-6.81 \pm 0.96}$ | $-9.90 \pm 8.21$ | $-61.68 \pm 2.15$ | $-41.07 \pm 4.67$ |
| | HV ↑ ($10^1$) | $0.00 \pm 0.00$ | $0.00 \pm 0.00$ | $1.75 \pm 0.88$ | $0.00 \pm 0.00$ | $2.43 \pm 0.72$ | $\mathbf{6.51 \pm 0.69}$ |
| | SP ↓ ($10^0$) | $\mathbf{0.00 \pm 0.00}$ | $\mathbf{0.00 \pm 0.00}$ | $0.86 \pm 0.37$ | $\mathbf{0.00 \pm 0.00}$ | $126.73 \pm 61.52$ | $8.50 \pm 3.63$ |
| Reacher | EUM ↑ ($10^1$) | $2.28 \pm 0.07$ | $2.29 \pm 0.07$ | $2.13 \pm 0.00$ | $2.29 \pm 0.02$ | $\mathbf{2.54 \pm 0.03}$ | $2.31 \pm 0.03$ |
| | HV ↑ ($10^6$) | $1.36 \pm 0.19$ | $1.40 \pm 0.19$ | $1.06 \pm 0.01$ | $1.29 \pm 0.03$ | $\mathbf{2.08 \pm 0.07}$ | $1.47 \pm 0.08$ |
| | SP ↓ ($10^{-2}$) | $4.28 \pm 1.97$ | $5.93 \pm 2.34$ | $\mathbf{2.00 \pm 0.50}$ | $9286.84 \pm 1410.67$ | $15.47 \pm 1.89$ | $5.77 \pm 1.61$ |
| FourRoom | EUM ↑ ($10^0$) | $1.96 \pm 0.05$ | $1.96 \pm 0.06$ | $\mathbf{2.15 \pm 0.04}$ | $0.00 \pm 0.00$ | $1.39 \pm 0.09$ | $1.90 \pm 0.15$ |
| | HV ↑ ($10^1$) | $0.74 \pm 0.08$ | $0.72 \pm 0.09$ | $\mathbf{1.02 \pm 0.06}$ | $0.00 \pm 0.00$ | $0.31 \pm 0.06$ | $0.82 \pm 0.14$ |
| | SP ↓ ($10^0$) | $0.03 \pm 0.01$ | $0.03 \pm 0.01$ | $0.09 \pm 0.02$ | $\mathbf{0.00 \pm 0.00}$ | $0.13 \pm 0.03$ | $0.11 \pm 0.03$ |

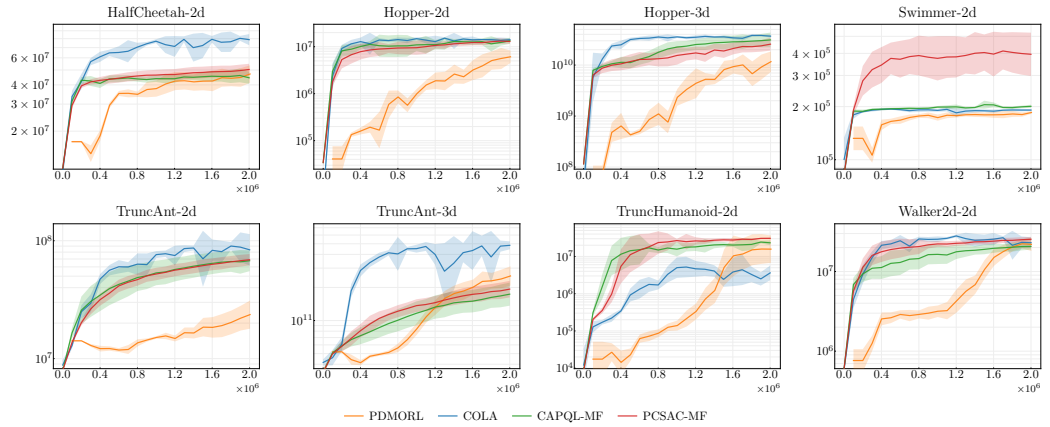



Figure 4: Hypervolume of model-free methods over environment steps on eight continuous action tasks from MO-gymnasium. All statistics are obtained from 8 independent runs and the mean and standard deviation are reported. PD-MORL allocates an initial interaction budget to identify key preferences [8]; hence its curve starts slightly later on the x-axis.

although it may change the preference parameterization. Proposition C.1 is complementary: it controls the exact target under an arbitrary bounded perturbation, whereas these tables measure end-to-end deep learning under a one-dimensional multiplicative perturbation.

For numerical stability, let $f_l$ denote the critic component used in STCH and $z_l = \omega_l(I_l - f_l)/\tau$. Then

$$0 \leq \frac{\partial u}{\partial f_l} = \omega_l \frac{e^{z_l}}{\sum_j e^{z_j}} \leq \omega_l.$$

Crossing $I_l = f_l$ is therefore smooth and cannot by itself make the utility-gradient coefficient diverge; if $I_l < f_l$, the affected exponential is below one before the common softmax normalization. We evaluate log-sum-exp and softmax with the standard maximum shift and smooth the online estimate by exponential averaging with coefficient $\beta = 3 \times 10^{-4}$. These safeguards address numerical exponentiation, but persistent utopia-point estimation error can still bias the actor's scalarization.

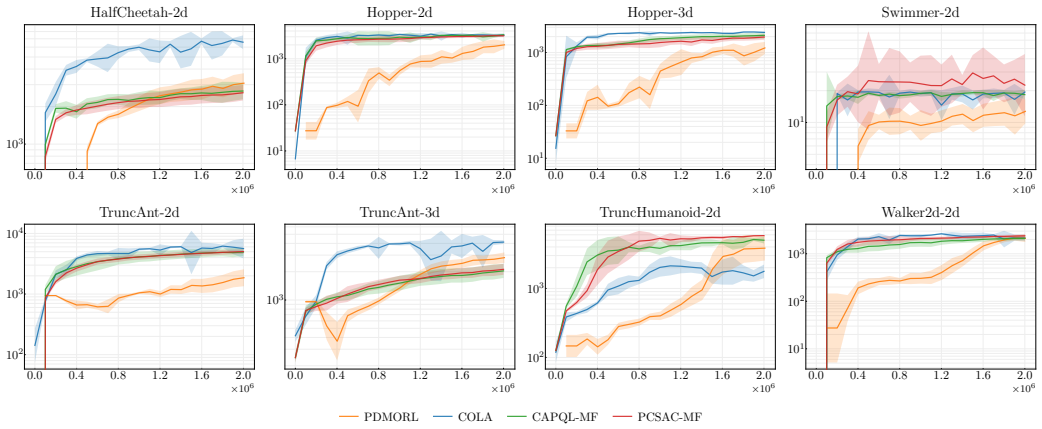



Figure 5: Expected Utility Metric of model-free methods over environment steps on eight continuous action tasks from MO-gymnasium. All statistics are obtained from 8 independent runs and the mean and standard deviation are reported. PD-MORL allocates an initial interaction budget to identify key preferences [8]; hence its curve starts slightly later on the x-axis.

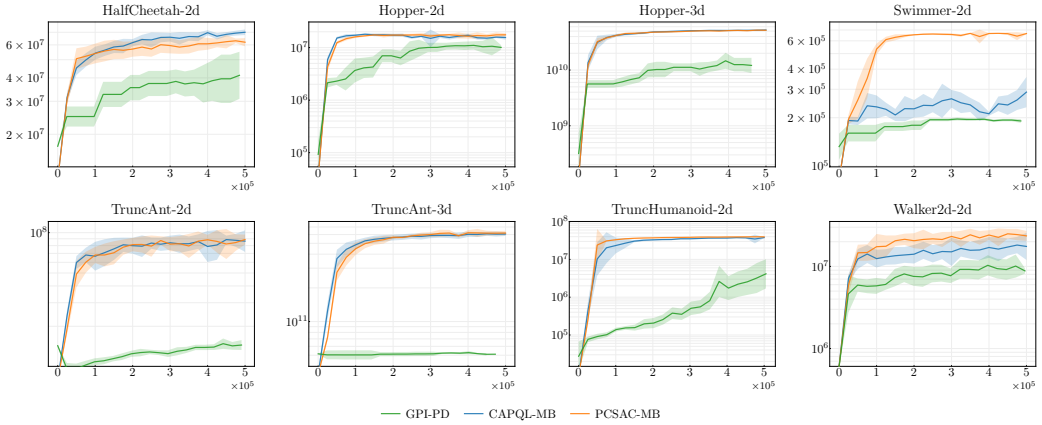



Figure 6: Hypervolume of model-based methods over environment steps on eight continuous action tasks from MO-gymnasium. All statistics are obtained from 8 independent runs and the mean and standard deviation are reported.

### J.5.2 Relative smoothness constant

The theoretical CMDPI scale $\alpha_{\mathrm{KL}} = L_{\mathrm{rel}}(1-\gamma)$ and the deep PCSAC entropy coefficient $\alpha_{\mathrm{ent}}$ are distinct. We sweep each method's corresponding practical coefficient over $\{0.01, 0.1, 1, 10, 100\}$ and find no common monotone degradation across HV, EUM, and SP (Tables 10–11). These are empirical sensitivity checks, not validation of the worst-case relative-smoothness constant in Proposition E.1.

## J.6 Diagnostic with KL in the critic target

We tested whether replacing PCSAC's uniform-reference entropy term by a previous-policy term gives a usable continuous-control approximation. The diagnostic variant, *PCSAC-KL*, uses the analytic KL divergence between the Gaussian actor and an exponentially averaged reference actor, broadcasts a fixed-coefficient KL penalty into the vector bootstrapped critic target, and otherwise follows PCSAC. To distinguish this diagnostic coefficient from the theoretical CMDPI scale $\alpha_{\mathrm{KL}} = L_{\mathrm{rel}}(1-\gamma)$, denote it by $\lambda_{\mathrm{KL}}$.

With STCH smoothing fixed at $\tau = 0.01$, we swept $\lambda_{\mathrm{KL}} \in \{10^0, 10^2, 10^4, 10^6, 10^8\}$ on the six two-objective continuous-control tasks. Every run with $\lambda_{\mathrm{KL}} \leq 10^2$ terminated with a non-finite loss

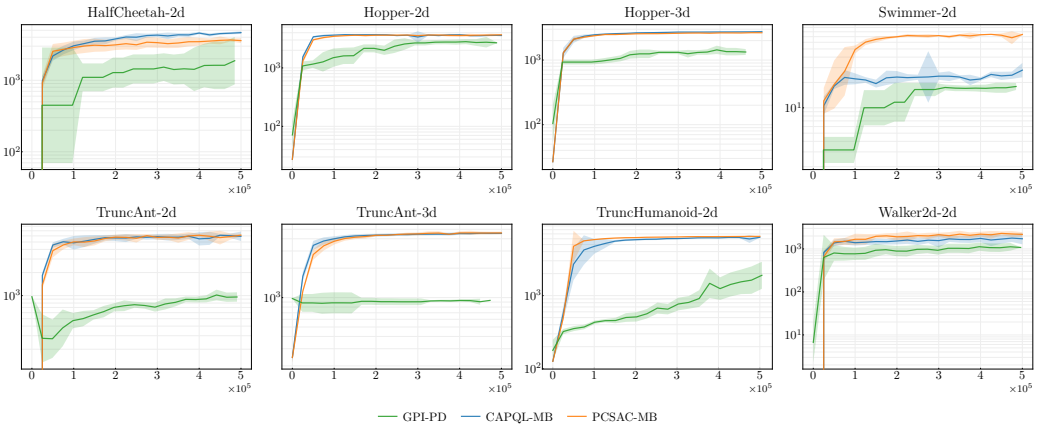



Figure 7: Expected Utility Metric of model-based methods over environment steps on eight continuous action tasks from MO-gymnasium. All statistics are obtained from 8 independent runs and the mean and standard deviation are reported.

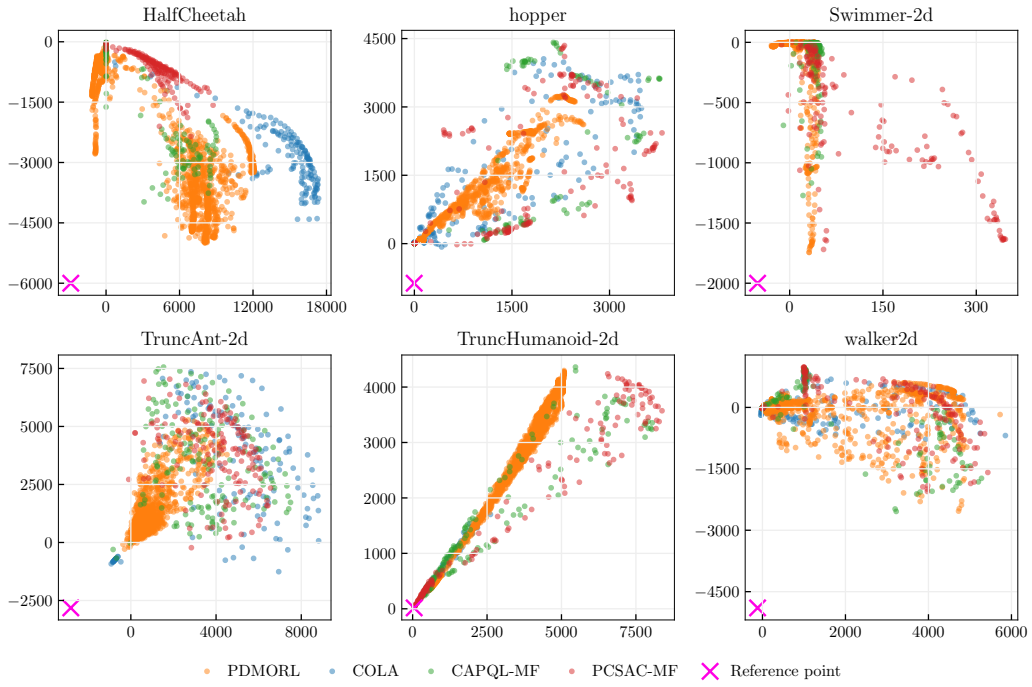



Figure 8: Final-step (2M steps) sum-of-reward-vectors scatter plots for model-free methods on 2-dimensional tasks.

(`inf` or `NaN`). Runs with $\lambda_{\mathrm{KL}} \geq 10^4$ completed, but learning slowed as the coefficient increased. On HalfCheetah at $\lambda_{\mathrm{KL}} = 10^8$, the forward-reward component remained zero after 2M environment steps in all four seeds, whereas the corresponding PCSAC runs exceeded $10^4$.

At the largest failing coefficient, $\lambda_{\mathrm{KL}} = 10^2$, the first observed non-finite quantity occurred at roughly 400K of 2M steps: the analytic KL for one replay transition became `inf`, the squared critic loss then became `inf`, the critic weights became `NaN`, and the actor subsequently became `NaN`. The minibatch-mean KL showed no preceding spike. The observed order supports the interpretation of a localized Gaussian-parameter outlier amplified by squared TD regression and bootstrapping, rather than a global policy jump. Because the KL is analytic, this is not Monte Carlo variance of a KL estimator; these observations also do not identify insufficient policy expressivity as the cause.

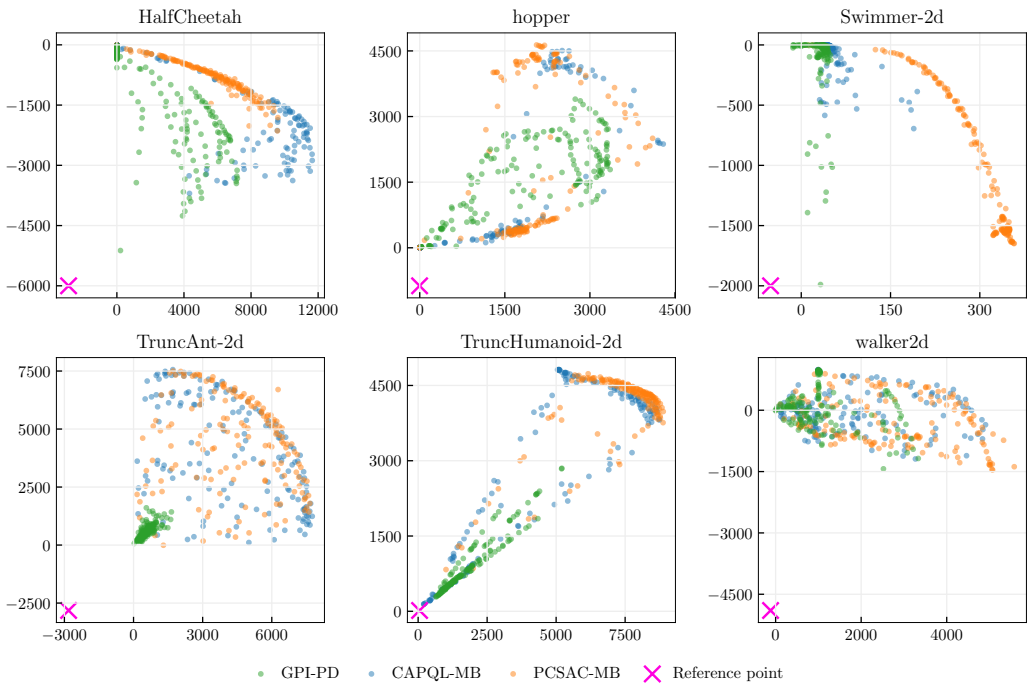



Figure 9: Final-step (500K steps) sum-of-reward-vectors scatter plots for model-based methods on 2-dimensional tasks.

Table 7: Comparison of final-step EUM, HV, and sparsity on continuous-action tasks from MO-Gymnasium. Entries are means ± standard errors of the mean (SEM) over eight seeds.

| | | PD-MORL | COLA | CAPQL-MF | PCSAC-MF | GPI-PD | CAPQL-MB | PCSAC-MB |
|---|---|---|---|---|---|---|---|---|
| HalfCheetah | EUM ↑ ($10^3$) | $3.39 \pm 0.32$ | $\mathbf{7.02 \pm 0.08}$ | $2.66 \pm 0.20$ | $2.57 \pm 0.20$ | $1.74 \pm 0.27$ | $4.64 \pm 0.10$ | $3.63 \pm 0.13$ |
| | HV ↑ ($10^7$) | $5.43 \pm 0.33$ | $\mathbf{8.19 \pm 0.10}$ | $4.43 \pm 0.20$ | $5.03 \pm 0.23$ | $4.01 \pm 0.28$ | $7.00 \pm 0.12$ | $6.25 \pm 0.15$ |
| | SP ↓ ($10^6$) | $0.59 \pm 0.04$ | $5.63 \pm 1.91$ | $5.48 \pm 1.27$ | $\mathbf{0.28 \pm 0.06}$ | $1.03 \pm 0.26$ | $4.91 \pm 1.06$ | $0.35 \pm 0.03$ |
| Hopper-2d | EUM ↑ ($10^3$) | $2.07 \pm 0.18$ | $2.99 \pm 0.35$ | $3.32 \pm 0.09$ | $3.15 \pm 0.10$ | $2.80 \pm 0.09$ | $3.59 \pm 0.06$ | $\mathbf{3.67 \pm 0.06}$ |
| | HV ↑ ($10^7$) | $0.62 \pm 0.09$ | $1.27 \pm 0.18$ | $1.35 \pm 0.10$ | $1.29 \pm 0.07$ | $1.10 \pm 0.06$ | $1.56 \pm 0.08$ | $\mathbf{1.74 \pm 0.07}$ |
| | SP ↓ ($10^6$) | $\mathbf{0.01 \pm 0.00}$ | $0.22 \pm 0.10$ | $1.13 \pm 0.48$ | $0.48 \pm 0.20$ | $0.20 \pm 0.11$ | $0.32 \pm 0.19$ | $1.00 \pm 0.19$ |
| Hopper-3d | EUM ↑ ($10^3$) | $1.26 \pm 0.20$ | $2.24 \pm 0.15$ | $2.09 \pm 0.12$ | $1.96 \pm 0.12$ | $1.31 \pm 0.10$ | $\mathbf{2.74 \pm 0.02}$ | $2.63 \pm 0.01$ |
| | HV ↑ ($10^{10}$) | $1.27 \pm 0.29$ | $3.28 \pm 0.38$ | $3.08 \pm 0.32$ | $2.60 \pm 0.34$ | $1.21 \pm 0.20$ | $\mathbf{5.23 \pm 0.06}$ | $5.14 \pm 0.04$ |
| | SP ↓ ($10^6$) | $0.03 \pm 0.01$ | $1.87 \pm 1.70$ | $0.05 \pm 0.01$ | $0.09 \pm 0.02$ | $0.02 \pm 0.01$ | $0.02 \pm 0.00$ | $\mathbf{0.01 \pm 0.00}$ |
| Swimmer | EUM ↑ ($10^1$) | $1.40 \pm 0.19$ | $1.85 \pm 0.06$ | $1.88 \pm 0.07$ | $2.29 \pm 0.85$ | $1.76 \pm 0.06$ | $2.80 \pm 0.32$ | $\mathbf{7.51 \pm 0.05}$ |
| | HV ↑ ($10^5$) | $1.89 \pm 0.02$ | $1.91 \pm 0.02$ | $2.01 \pm 0.02$ | $3.92 \pm 0.56$ | $1.92 \pm 0.02$ | $2.88 \pm 0.35$ | $\mathbf{6.67 \pm 0.04}$ |
| | SP ↓ ($10^4$) | $0.14 \pm 0.06$ | $\mathbf{0.01 \pm 0.00}$ | $3.14 \pm 1.64$ | $5.25 \pm 2.14$ | $0.82 \pm 0.78$ | $0.34 \pm 0.12$ | $2.04 \pm 0.09$ |
| TruncAnt-2d | EUM ↑ ($10^3$) | $2.11 \pm 0.42$ | $5.42 \pm 0.88$ | $4.94 \pm 0.36$ | $5.06 \pm 0.21$ | $1.03 \pm 0.09$ | $5.93 \pm 0.34$ | $\mathbf{6.17 \pm 0.15}$ |
| | HV ↑ ($10^7$) | $2.67 \pm 0.51$ | $8.14 \pm 1.15$ | $6.78 \pm 0.68$ | $6.96 \pm 0.38$ | $1.50 \pm 0.07$ | $8.64 \pm 0.63$ | $\mathbf{8.97 \pm 0.35}$ |
| | SP ↓ ($10^6$) | $0.34 \pm 0.09$ | $0.86 \pm 0.21$ | $0.79 \pm 0.16$ | $0.44 \pm 0.06$ | $\mathbf{0.14 \pm 0.09}$ | $2.10 \pm 1.04$ | $3.61 \pm 2.97$ |
| TruncAnt-3d | EUM ↑ ($10^3$) | $2.92 \pm 0.17$ | $4.21 \pm 0.06$ | $2.04 \pm 0.19$ | $2.13 \pm 0.17$ | $0.93 \pm 0.01$ | $4.53 \pm 0.05$ | $\mathbf{4.54 \pm 0.07}$ |
| | HV ↑ ($10^{11}$) | $2.72 \pm 0.24$ | $5.00 \pm 0.14$ | $1.75 \pm 0.23$ | $1.96 \pm 0.22$ | $0.52 \pm 0.01$ | $6.06 \pm 0.12$ | $\mathbf{6.10 \pm 0.13}$ |
| | SP ↓ ($10^5$) | $\mathbf{0.02 \pm 0.02}$ | $7.45 \pm 0.80$ | $0.57 \pm 0.11$ | $0.29 \pm 0.07$ | $5.34 \pm 2.27$ | $0.06 \pm 0.01$ | $0.15 \pm 0.05$ |
| TruncHumanoid | EUM ↑ ($10^3$) | $4.00 \pm 0.44$ | $1.70 \pm 0.34$ | $4.93 \pm 0.46$ | $5.85 \pm 0.05$ | $2.17 \pm 0.38$ | $6.28 \pm 0.04$ | $\mathbf{6.46 \pm 0.02}$ |
| | HV ↑ ($10^7$) | $1.69 \pm 0.24$ | $0.35 \pm 0.13$ | $2.32 \pm 0.30$ | $3.09 \pm 0.06$ | $0.50 \pm 0.16$ | $3.78 \pm 0.04$ | $\mathbf{3.95 \pm 0.03}$ |
| | SP ↓ ($10^5$) | $\mathbf{0.00 \pm 0.00}$ | $\mathbf{0.00 \pm 0.00}$ | $1.42 \pm 0.78$ | $0.82 \pm 0.20$ | $\mathbf{0.00 \pm 0.00}$ | $2.76 \pm 0.44$ | $0.52 \pm 0.08$ |
| Walker2d | EUM ↑ ($10^3$) | $2.31 \pm 0.14$ | $2.37 \pm 0.13$ | $2.11 \pm 0.09$ | $\mathbf{2.39 \pm 0.08}$ | $1.12 \pm 0.09$ | $1.73 \pm 0.24$ | $2.08 \pm 0.18$ |
| | HV ↑ ($10^7$) | $2.41 \pm 0.14$ | $2.47 \pm 0.13$ | $2.08 \pm 0.09$ | $\mathbf{2.54 \pm 0.09}$ | $1.01 \pm 0.15$ | $1.75 \pm 0.29$ | $2.23 \pm 0.22$ |
| | SP ↓ ($10^6$) | $\mathbf{0.25 \pm 0.19}$ | $1.49 \pm 0.64$ | $1.75 \pm 0.29$ | $0.99 \pm 0.38$ | $\mathbf{0.25 \pm 0.12}$ | $0.42 \pm 0.10$ | $0.58 \pm 0.20$ |

The diagnostic used the same Gaussian and clipped log-standard-deviation parameterization as the PCSAC base implementation.

For comparison, the worst-case $L_{\text{rel}}$ scale is of order $10^{11}$ in this setup, above the largest tested diagnostic coefficient. Proposition E.1 is tabular, however, so this comparison is descriptive and does not prescribe $\lambda_{\text{KL}}$ under function approximation. The sweep shows that no tested fixed coefficient both avoided non-finite value learning and learned at a comparable pace; it does not show that previous-policy regularization is intrinsically unstable when confined to a policy-improvement step.

Table 8: Sensitivity to the utopia-point scale for CMDPI. Entries are average ranks over the eight tasks (lower is better for all metrics), with ± 95% bootstrap confidence intervals over eight seeds; ranks compare the five parameter settings within each method.

| value | 0.1 | 0.5 | 1.0 | 2.0 | 5.0 |
|---|---|---|---|---|---|
| EUM | $3.4 \pm 0.34$ | $2.5 \pm 0.42$ | $3.0 \pm 0.43$ | $2.8 \pm 0.30$ | $3.4 \pm 0.31$ |
| HV | $3.2 \pm 0.48$ | $3.4 \pm 0.30$ | $2.9 \pm 0.43$ | $2.2 \pm 0.49$ | $3.2 \pm 0.31$ |
| SP | $3.2 \pm 0.47$ | $2.6 \pm 0.48$ | $3.8 \pm 0.30$ | $2.0 \pm 0.35$ | $3.4 \pm 0.38$ |

Table 9: Sensitivity to the utopia-point scale for PCSAC. Entries are average ranks over the eight tasks (lower is better for all metrics), with ± 95% bootstrap confidence intervals over eight seeds; ranks compare the five parameter settings within each method.

| value | 0.1 | 0.5 | 1.0 | 2.0 | 5.0 |
|---|---|---|---|---|---|
| EUM | $3.9 \pm 0.45$ | $3.6 \pm 0.55$ | $2.4 \pm 0.27$ | $2.4 \pm 0.34$ | $2.8 \pm 0.54$ |
| HV | $2.5 \pm 0.39$ | $3.2 \pm 0.68$ | $2.6 \pm 0.27$ | $2.9 \pm 0.40$ | $3.7 \pm 0.57$ |
| SP | $3.4 \pm 0.57$ | $3.9 \pm 0.39$ | $3.2 \pm 0.59$ | $2.6 \pm 0.35$ | $1.9 \pm 0.37$ |

### J.7 MPO-based continuous-control variants

We extend MPO [47] with a preference-conditioned actor $\pi_\theta(a|s,\omega)$ and vector critic $\mathbf{Q}_\phi(s,a,\omega) \in \mathbb{R}^m$. In its non-parametric E-step, the scalar score is

$$q_u(s,a,\omega) = u(\mathbf{Q}_\phi(s,a,\omega),\omega), \qquad w_j = \frac{\exp(q_u(s,a_j,\omega)/\eta)}{\sum_b \exp(q_u(s,a_b,\omega)/\eta)},$$

for actions sampled from the reference actor and the dual-optimized temperature $\eta > 0$. We use $u(\mathbf{z},\omega) = \omega^\top \mathbf{z}$ for LS-MPO and STCH for CMDPI-MPO. Within each MF/MB pair, only this scalarization differs; networks and other components are identical. Exponentially averaged reference networks, dual optimization, M-steps, and remaining MPO settings follow the same single-objective implementation.

Policy evaluation is unregularized; KL enters policy improvement only. This follows the type-2 distinction of Geist et al. [16, Sec. 5], who classify MPO as an MD-MPI type-2 method. Our critic-local nonlinear score is a heuristic, not the gradient-weighted reward of exact CMDPI; its tabular guarantees do not transfer.

We use the same eight tasks, eight seeds, and MF/MB budgets as in Appendix J.2.2, with common hyperparameters across tasks and STCH smoothing $\tau = 0.01$. Metrics follow the same protocol, except that HV reference points are recomputed after including all MPO variants. Consequently, HV values and ranks belong to this augmented comparison and need not match the main experiment. Tables 12–13 report means and 95% bootstrap confidence intervals.

Neither MF nor MB exhibited the PCSAC-KL-like collapse. Performance gains over LS-MPO are not uniform: on Swimmer, CMDPI-MPO improves mean HV in both settings and mean EUM in MB, while MF EUM is slightly lower (3.69 versus 3.79, in units of $10^1$). These results support feasibility of the MPO-based variant, not uniform superiority or the tabular convergence guarantee.

Table 10: Sensitivity to the practical KL/prox coefficient $\alpha_{\mathrm{KL}}$ for CMDPI. Entries are average ranks over the eight tasks (lower is better for all metrics), with ± 95% bootstrap confidence intervals over eight seeds; ranks compare the five parameter settings within each method.

| value | 0.01 | 0.1 | 1.0 | 10.0 | 100.0 |
|---|---|---|---|---|---|
| EUM | $3.6 \pm 0.39$ | $2.3 \pm 0.52$ | $2.1 \pm 0.25$ | $3.0 \pm 0.42$ | $4.0 \pm 0.43$ |
| HV | $4.2 \pm 0.35$ | $2.6 \pm 0.51$ | $1.9 \pm 0.38$ | $2.9 \pm 0.33$ | $3.5 \pm 0.38$ |
| SP | $3.4 \pm 0.54$ | $3.6 \pm 0.46$ | $2.9 \pm 0.40$ | $2.8 \pm 0.23$ | $2.4 \pm 0.49$ |

Table 11: Sensitivity to the entropy coefficient $\alpha_{\text{ent}}$ for PCSAC. Entries are average ranks over the eight tasks (lower is better for all metrics), with $\pm$ 95% bootstrap confidence intervals over eight seeds; ranks compare the five parameter settings within each method.

| value | 0.01 | 0.1 | 1.0 | 10.0 | 100.0 |
|---|---|---|---|---|---|
| EUM | $2.7 \pm 0.45$ | $2.2 \pm 0.47$ | $2.4 \pm 0.30$ | $4.4 \pm 0.33$ | $3.3 \pm 0.47$ |
| HV | $3.2 \pm 0.53$ | $2.4 \pm 0.54$ | $2.6 \pm 0.54$ | $3.4 \pm 0.45$ | $3.4 \pm 0.38$ |
| SP | $3.1 \pm 0.47$ | $3.1 \pm 0.34$ | $3.7 \pm 0.52$ | $2.4 \pm 0.29$ | $2.8 \pm 0.38$ |

### J.8 Compute Resources

Experiments were conducted on an internal compute cluster; a typical node has an Intel Xeon Gold 6138 CPU, an NVIDIA GeForce RTX 2080 Ti GPU, and about 360 GB RAM. Each continuous-control run uses 1 CPU core, 1 GPU, and approximately 2 GB of main memory.

**Continuous-Action Tasks** For continuous control tasks (8 tasks), each (task, seed) run takes approximately 24 hours. With 8 seeds, this corresponds to about 192 GPU hours per task and 1,536 GPU hours per method across all tasks. Across 7 methods, this yields approximately 10,752 GPU hours. Including implementation iterations and hyperparameter tuning (approximately 2–3 rounds), the total cost is roughly 20,000–30,000 GPU hours.

In practice, we parallelize multiple seeds on a single GPU when possible, reducing the effective wall-clock time to approximately one-third to one-half of the total GPU hours, i.e., roughly 7,000–15,000 hours.

**Discrete-Action Tasks** For discrete tasks (8 tasks), experiments are conducted on CPU only, with each (task, seed) run taking approximately 30 minutes. With 8 seeds, this corresponds to about 4 CPU hours per task and 32 CPU hours per method. Across 6 methods, this yields approximately 192 CPU hours. Including implementation iterations and hyperparameter tuning (approximately 2–3 rounds), the total cost is approximately 400–600 CPU hours, which remains negligible compared to continuous-control experiments.

### J.9 Existing assets and licenses

Table 14 lists the existing assets used in this work together with their licenses. Our continuous-action implementation builds on the official DHMBPO implementation, and our discrete-action implementation builds on the official PreCo implementation. The official PreCo and C-MORL repositories do not specify a license at the time of writing; we used them for research purposes only and do not redistribute them or code derived from them. Accordingly, our public repository (`https://github.com/4kubo/cmdpi-morl`, MIT License) contains the code for the tabular and continuous-action experiments, and we will release the discrete-action code once a license for the PreCo implementation is provided.

## K Limitations

Our results have distinct objective-space, tabular-algorithmic, and deep-implementation scopes.

(i) Proposition 2.1, Theorem 2.3, and Corollary F.1 require a compact convex attainable set. Proposition 2.2 and the resulting full-front coverage additionally use polyhedrality and a componentwise upper utopia point. By contrast, the CMDPI convergence analysis remains tabular: it assumes finite $\mathcal{S}$ and $\mathcal{A}$, $p_0 > 0$, and mirror descent over the occupancy-measure polytope $\mathcal{M}$. We provide neither a convergence theorem under function approximation nor a finite-sample error-propagation analysis.

(ii) Uniqueness of the objective-space target $\mathbf{J}^\omega$ implies neither uniqueness of an attaining policy nor convergence to a unique limiting policy selector. Exact tabular CMDPI with a fixed continuous full-support initialization does define an algorithm-induced policy $\pi_k^\omega$ at each finite iteration, and Appendix G proves its continuity in $\omega$. This finite-iterate fact motivates a single conditioned actor; it does not theoretically guarantee the deep implementation.

Table 12: MPO variants on eight continuous-action tasks. LS and CMDPI denote LS-MPO and CMDPI-MPO; entries are means $\pm$ 95% bootstrap confidence intervals over eight seeds. EUM/HV: higher is better; SP: lower is better. Values are rounded at the indicated scales; $0.00$ need not mean zero.

| | | MF (2M steps) | | MB (500K steps) | |
|---|---|---|---|---|---|
| Task | Metric (scale) | LS | CMDPI | LS | CMDPI |
| HalfCheetahd | EUM ($10^3$) | $5.46 \pm 0.03$ | $5.30 \pm 0.07$ | $4.80 \pm 0.06$ | $4.61 \pm 0.08$ |
| | HV ($10^7$) | $7.36 \pm 0.03$ | $7.66 \pm 0.09$ | $7.29 \pm 0.09$ | $7.14 \pm 0.08$ |
| | SP ($10^6$) | $8.26 \pm 0.29$ | $1.96 \pm 0.09$ | $2.68 \pm 0.22$ | $1.55 \pm 0.04$ |
| Hopper-2d | EUM ($10^3$) | $3.73 \pm 0.03$ | $3.42 \pm 0.08$ | $3.80 \pm 0.02$ | $3.71 \pm 0.02$ |
| | HV ($10^7$) | $1.77 \pm 0.02$ | $1.60 \pm 0.07$ | $1.83 \pm 0.02$ | $1.79 \pm 0.01$ |
| | SP ($10^6$) | $0.03 \pm 0.02$ | $0.17 \pm 0.04$ | $0.01 \pm 0.00$ | $0.06 \pm 0.02$ |
| Hopper-3d | EUM ($10^3$) | $2.65 \pm 0.02$ | $2.31 \pm 0.01$ | $2.66 \pm 0.01$ | $2.62 \pm 0.02$ |
| | HV ($10^{10}$) | $4.77 \pm 0.06$ | $3.86 \pm 0.07$ | $4.90 \pm 0.03$ | $4.90 \pm 0.07$ |
| | SP ($10^6$) | $0.02 \pm 0.00$ | $0.02 \pm 0.01$ | $0.01 \pm 0.00$ | $0.00 \pm 0.00$ |
| Swimmer | EUM ($10^1$) | $3.79 \pm 0.33$ | $3.69 \pm 0.57$ | $4.87 \pm 0.51$ | $6.11 \pm 0.24$ |
| | HV ($10^5$) | $3.52 \pm 0.30$ | $4.38 \pm 0.44$ | $4.76 \pm 0.46$ | $6.24 \pm 0.05$ |
| | SP ($10^4$) | $0.19 \pm 0.04$ | $4.68 \pm 0.65$ | $1.07 \pm 0.43$ | $1.60 \pm 0.10$ |
| TruncAnt-2d | EUM ($10^3$) | $3.43 \pm 0.72$ | $4.87 \pm 0.12$ | $5.53 \pm 0.15$ | $5.41 \pm 0.19$ |
| | HV ($10^7$) | $4.73 \pm 1.01$ | $6.67 \pm 0.22$ | $7.57 \pm 0.28$ | $7.45 \pm 0.37$ |
| | SP ($10^6$) | $0.12 \pm 0.04$ | $0.68 \pm 0.07$ | $3.35 \pm 0.76$ | $3.53 \pm 1.03$ |
| TruncAnt-3d | EUM ($10^3$) | $1.52 \pm 0.06$ | $1.07 \pm 0.06$ | $3.91 \pm 0.07$ | $3.49 \pm 0.04$ |
| | HV ($10^{11}$) | $1.41 \pm 0.05$ | $1.02 \pm 0.05$ | $4.66 \pm 0.17$ | $4.03 \pm 0.06$ |
| | SP ($10^5$) | $0.15 \pm 0.01$ | $0.22 \pm 0.03$ | $0.16 \pm 0.02$ | $0.18 \pm 0.03$ |
| TruncHumanoid | EUM ($10^3$) | $4.44 \pm 0.14$ | $4.61 \pm 0.13$ | $5.74 \pm 0.11$ | $5.93 \pm 0.04$ |
| | HV ($10^7$) | $1.95 \pm 0.12$ | $2.10 \pm 0.11$ | $3.25 \pm 0.11$ | $3.45 \pm 0.04$ |
| | SP ($10^5$) | $0.02 \pm 0.01$ | $0.40 \pm 0.20$ | $0.20 \pm 0.03$ | $0.89 \pm 0.07$ |
| Walker2d | EUM ($10^3$) | $2.50 \pm 0.04$ | $2.14 \pm 0.04$ | $1.80 \pm 0.20$ | $2.07 \pm 0.08$ |
| | HV ($10^7$) | $2.65 \pm 0.04$ | $2.26 \pm 0.07$ | $1.76 \pm 0.25$ | $2.19 \pm 0.12$ |
| | SP ($10^6$) | $0.38 \pm 0.33$ | $2.68 \pm 0.31$ | $0.12 \pm 0.03$ | $0.13 \pm 0.03$ |

(iii) The deep actor uses critic-local utility gradients and therefore optimizes a surrogate rather than the tabular objective $u(\mathbf{J}(\pi), \omega)$. Critic/entropy evaluation error, nonlinear aggregation, and replay/initial-distribution mismatch remain even apart from ordinary stochastic optimization error. The continuous-action experiments evaluate deep approximations of CMDPI, not an exact continuous-action implementation of the tabular recursion; Appendices J.6 and J.7 report the KL-in-target diagnostic and MPO-based variants. Consequently, the tabular convergence and coverage theorems do not transfer to PCSAC.

(iv) The utopia point is estimated online in the deep experiments. Appendix C proves Lipschitz sensitivity only for the exact STCH target and does not imply robustness of a learned policy. Proposition 2.1 is utopia-point independent, and in two objectives the full-front coverage result is also utopia-point independent. Moreover, as $\omega_i \to 0$, the gradient signal for objective $i$ vanishes and continuity is not asserted on the simplex boundary.

(v) For a restricted policy class with attainable set $\mathcal{J}_\Theta$, the global approximation gap $u(\mathbf{J}^\omega, \omega) - \max_{\mathbf{J} \in \mathcal{J}_\Theta} u(\mathbf{J}, \omega)$ generally remains. If $\mathcal{J}_\Theta$ is nonconvex, the global uniqueness, coverage, and convergence conclusions do not carry over; under additional regularity, one can at most appeal to parameter-space Pareto stationarity as in Lin et al. [19]. Empirically, our conclusions are limited to

Table 13: Average ranks from the augmented comparison (lower is better), with $\pm$ 95% bootstrap confidence intervals over tasks and seeds. Only MPO columns are shown; ranks are among six MF methods or five MB methods, respectively. These are not pairwise MPO ranks.

| | MF | | MB | |
|---|---|---|---|---|
| Metric | LS-MPO | CMDPI-MPO | LS-MPO | CMDPI-MPO |
| HV | $2.88 \pm 0.56$ | $3.12 \pm 0.55$ | $2.62 \pm 0.39$ | $2.88 \pm 0.33$ |
| EUM | $2.50 \pm 0.59$ | $3.38 \pm 0.53$ | $2.50 \pm 0.38$ | $3.00 \pm 0.32$ |

Table 14: Existing assets used in this work.

| Asset | Use in this work | License |
|---|---|---|
| MO-Gymnasium [27] `https://github.com/Farama-Foundation/MO-Gymnasium` | Benchmark environments (all tasks) | MIT |
| MuJoCo [48] `https://github.com/google-deepmind/mujoco` | Physics simulator (continuous-action tasks) | Apache-2.0 |
| MORL-Baselines [27] `https://github.com/LucasAlegre/morl-baselines` | GPI-PD baseline [36]; preference generation (`equally_spaced_weights`) | MIT |
| PyTorch [49] `https://github.com/pytorch/pytorch` | Deep learning framework | BSD-3-Clause |
| DHMBPO [42] `https://github.com/4kubo/erl_lib` | Base of our continuous-action implementation (MF/MB) | MIT |
| PreCo [34] `https://github.com/YYCAAA/ICML25PCRL` | Base of our discrete-action implementation (not released; see text); LS, CAPQL, and PreCo baselines | Not specified |
| C-MORL [9] `https://github.com/RuohLiuq/C-MORL` | Baseline (discrete-action tasks) | Not specified |
| PD-MORL [8] `https://github.com/tbasaklar/PDMORL-Preference-Driven-Multi-Objective-Reinforcement-Learning-Algorithm` | Baseline (continuous-action tasks) | MIT |
| COLA [35] `https://github.com/yeshenpy/COLA` | Baseline (continuous-action tasks) | MIT |

the reported benchmarks, seeds, and training budgets, and the task-wise tables show no uniform gain across metrics or tasks.

(vi) The smoothness analysis is specialized to the STCH utility used here; STCH itself is generally not strictly concave. We do not establish analogous CMDPI constants or behavior for other nonlinear utility classes.

## L Broader Impact

This work studies preference-conditioned multi-objective reinforcement learning from a methodological perspective. Potential positive impacts include improved decision-making in systems that must balance multiple objectives, such as robotics, autonomous systems, and resource management, where trade-offs between competing criteria are essential.

Potential negative impacts arise when such systems are deployed in real-world settings involving safety- or fairness-critical decisions. In particular, biased or mis-specified preferences may lead to systematically suboptimal or unfair outcomes, even when the algorithm functions as intended. In addition, increased flexibility in representing trade-offs could be misused to justify undesirable decision policies under certain preference configurations.

As this work is primarily foundational and does not introduce new datasets or deployable systems, direct risks are limited. Nevertheless, careful consideration of preference design, evaluation protocols, and deployment constraints is necessary when applying such methods in practice.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the main contributions and scope, and the claims are supported by the theoretical and empirical results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly state the key assumptions underlying the derivation and theoretical development of the proposed method, and consistently scope our evaluation and claims under these assumptions. We further provide a dedicated "Limitations" section in the appendix to clearly articulate the current limitations and boundaries of our approach.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theoretical results clearly state their assumptions, and complete proofs are provided in the appendix with appropriate references from the main text.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies experimental setups, models, and hyperparameters, and describes evaluation protocols sufficiently to enable reproduction of the results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification:
   Code and instructions for reproducing the tabular and continuous-action experiments are publicly available at `https://github.com/4kubo/cmdpi-morl` under the MIT License (Section 4). The code for the discrete-action experiments is built on the official PreCo implementation, which does not specify a license; we therefore withhold its release until a license is provided (Appendix J.9). Implementation details and experimental protocols for all experiments, including the discrete-action ones, are given in Appendices H and J.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the experimental setup, evaluation protocol, and key hyperparameters needed to interpret the results. Additional implementation and training details are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All experiments use eight random seeds, and the reported uncertainty captures variability across seeds. Tables 4, 5, and 8–13 report 95% bootstrap confidence intervals, Tables 6 and 7 report the standard error of the mean of the task-wise final metrics, and

Figures 4–7 show the mean with the standard deviation as shaded regions; each table and figure caption states which statistic is used.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide details on the computational resources used (e.g., hardware, memory, and runtime) in the appendix. This information is sufficient to reproduce the reported experiments and estimate the computational cost.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research was conducted in accordance with the NeurIPS Code of Ethics. No ethical concerns were identified in the methodology, experiments, or reporting.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We provide a dedicated "Broader Impact" in the appendix that discusses both potential positive and negative societal impacts of our work.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not involve releasing data or models with a high risk of misuse, so safeguards are not applicable.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All existing assets (environments, libraries, and codebases) are cited, and Appendix J.9 lists each of them with its URL and license. Our continuous-action implementation builds on the official DHMBPO implementation and our discrete-action implementation on the official PreCo implementation. Because the PreCo and C-MORL implementations do not specify a license, we state this explicitly, used them only for research purposes, and do not redistribute them or code derived from them; in particular, our discrete-action code is withheld until a license is provided (Appendix J.9).

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The MIT-licensed code for the tabular and continuous-action experiments is released alongside instructions for reproducing those experiments.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve research with human subjects or crowdsourcing, so IRB approval is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core methodology of this paper does not involve LLMs as an important or non-standard component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.