Soft Forward-Backward Representations for Zero-shot Reinforcement Learning with General Utilities

Marco Bagatella
ETH Zürich
MPI-IS, Tübingen
mbagatella@ethz.ch

Thomas Rupf
ETH Zürich

Georg Martius *
University of Tübingen

Andreas Krause *
ETH Zürich

Abstract

Recent advancements in zero-shot reinforcement learning (RL) have facilitated the extraction of diverse behaviors from unlabeled, offline data sources. In particular, forward-backward algorithms (FB) can retrieve a family of policies that approximately solves any standard RL problem (with additive rewards, linear in the occupancy measure), given sufficient capacity. While retaining zero-shot properties, we tackle the greater problem class of *RL with general utilities*, in which the objective is an arbitrary differentiable function of the occupancy measure. This setting is strictly more expressive, capturing tasks such as distribution matching or pure exploration, which may not be reduced to additive rewards. We show that this additional complexity can be captured by a novel, maximum entropy (*soft*) variant of the forward-backward algorithm, which recovers a family of stochastic policies from offline data. When coupled with zero-order search over compact policy embeddings, this algorithm can sidestep iterative optimization schemes, and optimizes general utilities directly at test-time. Across both didactic and high-dimensional experiments, we demonstrate that our method retains favorable properties of FB algorithms, while also extending their range to more general RL problems.

1 Introduction

Departing from the strict necessity of an explicit reward signal at training time, zero-shot RL has recently received increasing attention. Although several definitions have been brought forward, they largely agree: zero-shot methods may use significant amounts of compute in an initial, unsupervised pretraining phase, but should be capable of producing near-optimal behavior with minimal computation for rewards specified at test-time [Touati et al., 2023, Frans et al., 2024, Sikchi et al., 2025a, Agarwal et al., 2025]. Among these methods, forward-backward representations [Touati and Ollivier, 2021] have been proposed as a zero-shot algorithm for solving *arbitrary* (Markov) reward functions at test-time. Although some have speculated that this family of reward functions is sufficient to extract any behavior of interest [Silver et al., 2021], its expressiveness remains limited [Kumar et al., 2022].

In this work, we turn towards a broader class of RL problems: while Markov rewards lead to a linear objective in the policy's occupancy, we aim to optimize arbitrary differentiable functions of the policy's occupancy, known as General Utilities (GU, Zhang et al., 2020, 2021, Kumar et al., 2022, Barakat et al., 2023). This increased scope includes interesting problem instances, such as pure exploration [Hazan et al., 2019], active learning [Mutny et al., 2023] and learning from observations [Torabi et al., 2019], which are generally beyond the reach of linear RL algorithms (see Section 6 and Appendix B for a broader discussion). Ad-hoc algorithms have been proposed for specific instances of GUs (e.g., Ho and Ermon [2016], Hazan et al. [2019], Ma et al. [2022], Bolland et al. [2024]), but they are not

*Equal senior authorship.

designed to generalize beyond their respective domain. A large body of work has instead proposed principled algorithms for optimizing arbitrary GUs, with particular focus on convex [Zahavy et al., 2021, Mutti et al., 2022] or submodular objectives [De Santi et al., 2024]. Generally, the resulting practical algorithms rely on online learning through semi-gradient methods [Zahavy et al., 2021, De Santi et al., 2024], e.g. by constructing sequences of rewards from the gradient of the objective. While these methods enjoy strong theoretical guarantees in terms of convergence [Zhang et al., 2020, Kumar et al., 2022, Barakat et al., 2025], they scale poorly to high-dimensional settings: they need separate *online* training for each GU, and they have not been shown to scale beyond tabular MDPs and linear policies.

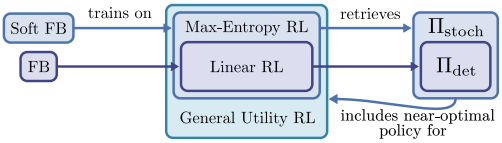


Figure 1: We propose Soft FB, a soft version of the Forward-Backward algorithm which solves maximum entropy RL instances to retrieve a richer set of stochastic policies, and searches them to optimize general utilities at test-time.

In this work, we introduce **S**oft **FB** (SFB), a practical zero-shot algorithm for RL with GUs. Building upon the forward-backward framework [Touati and Ollivier, 2021], we introduce a soft variant [Ziebart et al., 2008, Haarnoja et al., 2018, Hunt et al., 2019] which retrieves a family of stochastic policies, approximating all solutions to maximum entropy RL problems. Interestingly, given sufficient capacity and exact training, we show that the set of policies retrieved through entropy-regularized linear RL contains a near-optimal Markov policy for *each* differentiable GU. Thus, one may optimize GUs zero-shot by simply (i) learning maximum entropy policies and (ii) searching among them to find a good policy for a given GU. The first step bypasses the challenge of parameterizing GU instances. While the second step remains generally intractable, we find that it can be solved approximately and efficiently through zero-order search over low-dimensional policy representations. SoftFB departs significantly from standard policy-gradient approaches for GUs [Zhang et al., 2020, Kumar et al., 2022]. While, it only guarantees the *existence* of a solution among retrieved policies under perfect training, we find that it is better suited for practical settings: it does not require re-training for each GU nor access to online rollouts, and it scales to high-dimensional environments on which optimization of arbitrary GUs has not yet been demonstrated, to the best of our knowledge [Barakat et al., 2023]. As a result, our method can be seen both as a generalization of existing zero-shot frameworks, or as a practical offline algorithm for optimizing GUs.

We demonstrate the flexibility of the method in illustrative continuous environments, in which Soft FB can solve several zero-shot RL problems that are out-of-reach for existing, linear methods. We then extend this evaluation to established, complex zero-shot benchmarks, demonstrating scalability and studying the impact of entropy regularization for standard, linear tasks. Our contributions can be summarized as follows: (i) we propose Soft FB, a *soft* forward-backward algorithm, capable of retrieving stochastic policies and optimizing GUs zero-shot, (ii) we provide formal guarantees for the expressiveness of our method (iii) we present a thorough empirical evaluation of our method, demonstrating its ability to capture a richer class of policies while retaining the desirable properties of forward-backward algorithms.

2 Background

In the context of sequential decision making, a general way to describe an environment is through a reward-free MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, \mu_0, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ are potentially continuous state and actions spaces, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is a stochastic transition kernel, $\mu_0 \in \Delta(\mathcal{S})$ is an initial state distribution, and $\gamma \in (0, 1)$ is a discount factor. Additionally, the agent's behavior may be described by a Markov policy, that is a state-conditional action distribution $\pi : \mathcal{S} \to \Delta(\mathcal{A})$. These simple entities induce a discounted occupancy measure over states-action pairs, also known as the *successor measure* of policy π:

$$M^\pi(s, a, X) = (1-\gamma) \sum_{t \geq 0} \gamma^t \Pr((s_t, a_t) \in X \,|\, s_0 = s, a_0 = a)$$

where $X \subseteq \mathcal{S} \times \mathcal{A}$ and $\Pr(\cdot)$ is the visitation likelihood under policy π and dynamics P. If $\mathcal{S}$ and $\mathcal{A}$ are both discrete, M^π may be directly represented as a $(|\mathcal{S}||\mathcal{A}| \times |\mathcal{S}||\mathcal{A}|)$ stochastic matrix. Intuitively, each element of the matrix would track the discounted cumulative likelihood of visiting a given state-action pair, starting from another state-action pair. While our algorithm generalizes to continuous spaces, we will consider the discrete case in our formal derivations.

We further note that the successor measure might instead track state-only visitations. By overloading $X \in \mathcal{S}$,

$$M^\pi_{\mathcal{S}}(s,a,X) = (1-\gamma)\sum_{t\geq 0}\gamma^t \Pr(s_t \in X \,|\, s_0 = s, a_0 = a) = M^\pi(s,a,X\times\mathcal{A}).$$

Due to a particular choice of parameterization, our practical algorithm will estimate this object directly. However, samples $(s,a) \sim M^\pi$ can be easily obtained by sampling $s \sim M^\pi_{\mathcal{S}}$ and $a \sim \pi(\cdot|s)$, as we describe in Section 4.4. We will finally use M^π and $\mathrm{M}^\pi_{\mathcal{S}}$ to denote successor measures when marginalized over the initial state distribution μ_0, e.g. $\mathrm{M}^\pi(X) = \mathbb{E}_{s_0\sim\mu_0, a_0\sim\pi(\cdot|s_0)} M^\pi(s_0,a_0,X)$.

Many interesting RL problem instances can be directly defined as a function of the successor measure M^π, and classified according to the properties of the function. Standard RL problems can be expressed as a discounted sum of Markov rewards: if the rewards are expressed as a vector R the objective can be simply computed as $J^\pi_{\text{lin}} = \langle \mathrm{M}^\pi, R\rangle$, and this instance is thus referred to as *Linear* RL. Another instance that has received significant attention in recent years is that of *Maximum Entropy* RL [Ziebart et al., 2008, Haarnoja et al., 2018], which adds an entropy term to the linear objective: $J^\pi_{\mathcal{H}} = \langle \mathrm{M}^\pi, R + \mathcal{H}^\pi\rangle$, where $\mathcal{H}^\pi$ is a vector which contains the entropy of the policy $\mathcal{H}^\pi(s,a) = \mathcal{H}(\pi(\cdot|s))$ for each state-action pair $(s,a) \in \mathcal{S}\times\mathcal{A}$. Because of this policy-dependent entropy term, maximum entropy RL is not a linear problem in the occupancy, but rather an instance of *Convex* RL [Mutti et al., 2022]. In general, Convex RL problems can be expressed as $J^\pi_{\text{con}} = f(\mathrm{M}^\pi)$, where $f(\cdot)$ is an arbitrary convex function. Finally, all these instances are encompassed by *General Utility* RL [Zhang et al., 2020], which aims at optimizing a general differentiable scalar objective $J^\pi_{\text{GU}} = f(\mathrm{M}^\pi)$.

3 Forward-Backward Representations and General Utilities

Having established common terminology, we will now analyze the forward-backward algorithm for zero-shot reinforcement learning [Touati and Ollivier, 2021, Blier et al., 2021] and pinpoint its limitations motivating this work. At its core, the FB framework introduces a family of parameterized policies $\{\pi_z\}_{z\in\mathcal{Z}}$ with $\mathcal{Z} := \mathbb{R}^d$, each inducing an occupancy $M^z := M^{\pi_z}$. Crucially, each occupancy undergoes a specific low-rank decomposition

$$M^z = F_z^\top B, \tag{1}$$

where both F_z and B are $(d \times |\mathcal{S}||\mathcal{A}|)$ matrices, and may be called the *forward* and *backward* representation matrices. If the decomposition holds exactly, then for a $|\mathcal{S}||\mathcal{A}|$-dimensional reward vector R, the Q-function for a policy π_z can be simply expressed as

$$Q^z = M^z R = F_z^\top B R \overset{BR=z'}{:=} F_z^\top z', \tag{2}$$

where B can be reinterpreted as a projection from the space of reward vectors to a d-dimensional reward embedding $z' := BR$. Finally, the policy π_z is enforced to be optimal with respect to the discounted sum of rewards encoded by its own reward embedding, that is

$$\pi_z \in \operatorname*{argmax}_{\mathcal{A}} F_z^\top z, \tag{3}$$

leading to the following formal result:

Theorem 3.1. Touati and Ollivier [2021] *For an arbitrary bounded reward vector $R \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, if both Equations 1 and 3 hold for all $z \in \mathcal{Z}$, π_{BR} is optimal with respect to R: $M^{\pi_{BR}}R = \max_\pi M^\pi R$.*

While this theorem guarantees the retrieval of a solution for each *linear* RL problem, the policy extraction objective in Equation 3 (or its empirical counterparts, cf. Appendix E.3) may simply produce deterministic policies. This is sufficient for linear RL instances, as they admit optimal deterministic policies [Sutton and Barto, 1998], but remains generally suboptimal for GUs. This is the main motivating insight: the set of policies retrieved by FB may not contain the solution to non-linear RL problems.

Remark 3.2. Let $\Pi = \{\pi : \mathcal{S} \to \Delta(\mathcal{A})\}$ be the set of all Markov policies. There exist an MDP $\mathcal{M}$ and a scalar function f of occupancy measures and a set of policies $\Pi_z = \{\pi_z\}_{z\in\mathcal{Z}}$ such that (i) π_z satisfies Equations 1 and 3 for all $z \in \mathcal{Z}$, and (ii) $\max_{\pi\in\Pi} f(\mathrm{M}^\pi) > \max_{\pi\in\Pi_z} f(\mathrm{M}^\pi)$.

Counterexample As a straightforward counter-example, it suffices to consider an MDP with a single state $\mathcal{S} = \{s_0\}$ and two actions $\mathcal{A} = \{a_0, a_1\}$, and a convex RL objective such as *pure*

exploration over states and actions, i.e. $J^\pi = f(\mathrm{M}^\pi) = \mathcal{H}(\mathrm{M}^\pi)$, where $\mathcal{H}(\cdot)$ denotes entropy over state-action pairs. For each reward function in $\mathbb{R}^2$, forward and backward representations satisfying both Equations 1 and 3 exist, but may not retrieve the (optimal) uniform policy (see Appendix A.2). Following this motivation, we propose a variant of the FB algorithm which recovers a richer class of policies, and may provably optimize all GUs.

4 *Soft* Forward-Backward Representations

4.1 Core algorithm

As for the linear case, we start by introducing a family of parameterized policies $\{\pi_z\}_{z\in\mathcal{Z}}$, and decompose their occupancy as $M^z = F_z^\top B$, thus enforcing Equation 1 again. We however introduce an entropy regularization term to encourage stochastic behavior [Haarnoja et al., 2018]. The action-state value function for a policy π_z is then expressed as

$$Q^z_{\text{soft}} = M^z(R + \mathcal{H}_{\pi_z}) = F_z^\top BR + M^z\mathcal{H}_{\pi_z} \overset{z'=BR}{=} F_z^\top z' + M^z\mathcal{H}_{\pi_z}, \tag{4}$$

where $\mathcal{H}_{\pi_z} \in \mathbb{R}_+^{|\mathcal{S}||\mathcal{A}|}$ contains the policy's entropy at each state, and is a convex term in the occupancy measure. The entropy-regularized Q-function Q^z_{soft} breaks down into two components: the discounted sum of rewards (estimated as $M^zR \approx F_z^\top z'$), and the discounted sum of entropies $M^z\mathcal{H}_{\pi_z}$. This decomposition of the Q-function will allow these two components to be trained independently through TD learning, as we will see in Section 4.3. Conforming to soft policy improvement [Haarnoja et al., 2018], each policy π_z is then defined with respect to its regularized action-value function:

$$\pi_z \propto \exp(F_z^\top z + M^z\mathcal{H}_{\pi_z}), \tag{5}$$

Equation 1, which remains unchanged, and Equation 5 are the core of the Soft FB algorithm. The introduction of entropy regularization alters the set of policies that are retrieved during training to include all policies with full support. Crucially, as we will show formally in Section 4.2, training on maximum entropy RL instances alone is sufficient to capture ϵ-optimal solutions to general utilities. We will later describe a practical algorithm for estimating soft forward and backward representations from continuous data, in combination with sample-based objectives and function approximation (see Section 4.3).

4.2 Guarantees

Due to the introduction of entropy regularization, we can show that Soft FB retrieves the optimal policy among all Markov policies Π for each maximum entropy RL instance.

Theorem 4.1. *For an arbitrary bounded reward vector $R \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, if both Equations 1 and 5 hold for all $z \in \mathcal{Z}$, π_{BR} is the optimal maximum entropy policy with respect to R: $M^{\pi_{BR}}(R + \mathcal{H}_{\pi_{BR}}) = \max_{\pi\in\Pi} M^\pi(R + \mathcal{H}_\pi)$.*

Moreover, while only trained for solving maximum entropy instances, we can show that the set of policies retrieved by Soft FB includes arbitrarily good solutions to a greater class of problems: namely, it optimizes *any* GU.

Theorem 4.2. *Let f be an arbitrary differentiable function of occupancy measures, and $\tilde{\Pi}_z$ be the set of policies retrieved by Soft FB. If both Equations 1 and 5 hold for all $z \in \mathcal{Z}$, for any $\epsilon > 0$ there exists a reward embedding $z' \in \mathcal{Z}$ such that $\pi_{z'} \in \tilde{\Pi}_z$ and $\max_{\pi\in\Pi} f(\mathrm{M}^\pi) - f(\mathrm{M}^{\pi_{z'}}) < \epsilon$.*

We remark that this is an *existence* result; a good policy is in practice recovered via search and successor-measure estimation, as described in Section 4.4. We refer to Appendix A for proofs of these two statements and further formal remarks. The latter has important practical consequences: while parameterizing GUs is far from straightforward, maximum entropy instances can be easily parameterized, as we discuss in Section 4.3. As a result, *we can directly optimize general utilities at test-time, while only dealing with tractable maximum entropy objectives at training time*.

4.3 Practical algorithm

An algorithmic instantiation of Soft FB in potentially continuous spaces needs to address two points. First, the embedding $z \in \mathcal{Z}$ solving each General RL problem as described so far would lie in $\mathbb{R}^d$,

which is arbitrarily large and impractical to search. We will show that this can be easily addressed through a simple reparameterization. Second, we will introduce sample-based objectives that may be used to train function approximators over continuous state and action spaces, and approximately enforce Equations 1 and 5. A compact description of the full algorithm is reported in Appendix D.

Reparameterization Soft FB reduces policy optimization to search over stochastic policies parameterized by vectors in $\mathcal{Z} = \mathbb{R}^d$. In principle, every stochastic policy may be retrieved for some $z \in \mathcal{Z}$; however, this embedding could lie anywhere in $\mathbb{R}^d$. In fact, deterministic policies are only retrieved by embeddings whose norm approaches infinity (such that $F_z^\top z$ completely outweighs the entropy regularization term in Equation 5). Fortunately, recalling that in maximum entropy RL optimal policies are invariant to reward scaling as long as the entropy coefficient is scaled proportionally, we can find an alternative parameterization for which z may be sampled from a bounded space. For a given policy $\pi_{\bar{z}}$ and an embedding $z \in \mathcal{Z}$, we observe that

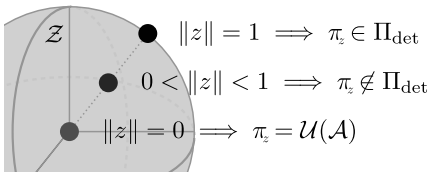


Figure 2: Geometric interpretation of z after reparameterization: the stochasticity of π_z grows with $\|z\|$.

$$Q^{\bar{z}} = F_{\bar{z}}^\top z + M^{\bar{z}}\mathcal{H}_{\pi_{\bar{z}}} \propto \frac{1}{\|z\|+1}(F_{\bar{z}}^\top z + M^{\bar{z}}\mathcal{H}_{\pi_{\bar{z}}}) \overset{z' := \frac{z}{\|z\|+1}}{=} F_{\bar{z}}^\top z' + (1 - \|z'\|)M^{\bar{z}}\mathcal{H}_{\pi_{\bar{z}}}. \quad (6)$$

Therefore, as $0 \leq \|z'\| < 1$, we can now map all maximum entropy RL instances to embeddings $z \in \mathcal{Z}$ sampled from the volume of a d-dimensional hypersphere [2]. This allows an intuitive geometric interpretation: the origin of $\mathbb{R}^d$ is associated with a uniform policy (as z is the null vector, and the Q-function equates the entropy bonus alone), and embeddings on the surface parameterize the family of deterministic policies that standard FB algorithms would return (see Figure 2).

Training objectives A practical instantiation of Soft FB for continuous spaces can leverage the framework proposed by Touati and Ollivier [2021]: we can approximate each entry of M^z by the dot product between a forward and backward embedding, produced by function approximators $F_\theta : \mathcal{S} \times \mathcal{A} \times \mathcal{Z} \to \mathcal{Z}$ and $B_\phi : \mathcal{S} \to \mathcal{Z}$ [3]. The decomposition from Equation 1 is thus generalized as an equality over measures: $M^z(s, a, ds') \approx F_\theta(s, a, z)B_\phi(s')\rho(s')$, where $\rho(\cdot)$ represents the data distribution. We can then adopt a standard objective for training forward and backwards representations, minimizing Bellman residuals by sampling from the data distribution ρ (i.e., an offline dataset of transitions) [Blier et al., 2021]:

$$\mathcal{L}_{\text{FB}}(\theta, \phi) = \mathop{\mathbb{E}}_{\substack{z\sim\mathcal{U}(\mathcal{Z}),\ (s_t,a_t,s_{t+1})\sim\rho \\ s'\sim\rho,\ a_{t+1}\sim\pi_z(s_{t+1})}} \Big[\big(F_\theta(s_t, a_t, z)^\top B_\phi(s') - \gamma \bar{F}(s_{t+1}, a_{t+1}, z)^\top \bar{B}(s')\big)^2 - 2F_\theta(s_t, a_t, z)^\top B_\phi(s_{t+1}) \Big], \quad (7)$$

where $\bar{F}$ and $\bar{B}$ may be target networks [Fujimoto et al., 2018], and forward representations are averaged over twin networks. As in Touati and Ollivier [2021], we employ an auxiliary orthonormalization loss over B_ϕ. In continuous spaces, the policy π_z is also represented through function approximation, and may be expressed as $\pi_\psi(\cdot|s, z)$ [4]. However, the policy's learning rule needs to be altered significantly, in order to account for entropy regularization. First, we'll split the maximum entropy action-state value function from Equation 6 in a reward-based component, and an entropy-based component: $Q^z(s, a) = Q^z_R(s, a) + (1 - \|z\|)Q^z_{\mathcal{H}}(s, a)$. The former is easily computed as $Q^z_R(s, a) \propto F_\theta(s, a, z)^\top z$, while the latter can be estimated by training a parameterized critic $Q_{\mathcal{H},\eta}(s, a, z)$ through the TD objective

$$\mathcal{L}_{\mathcal{H}}(\eta) = \mathop{\mathbb{E}}_{\substack{z\sim\mathcal{U}(\mathcal{Z}) \\ (s_t,a_t,s_{t+1})\sim\rho \\ a_{t+1}\sim\pi_z(s_{t+1})}} \Big[\Big(Q_{\mathcal{H},\eta}(s_t,a_t,z) - \gamma \big(\bar{Q}_{\mathcal{H},\eta}(s_{t+1},a_{t+1},z) - \log \pi_\psi(a_{t+1}|s_{t+1},z) \big) \Big)^2 \Big]. \quad (8)$$

[2] Touati and Ollivier [2021] operate over normalized embeddings z, thus sampling them from the *surface* of the hypersphere.

[3] Following prior work [Touati and Ollivier, 2021], our practical instantiation models the successor measure $M^\pi_{\mathcal{S}}$ over states only.

[4] While Soft FB is compatible with expressive policies with explicit likelihoods (e.g., normalizing flows), in our evaluation π_ψ will be parameterized as a diagonal Gaussian.

In any case, recalling Equations 5 and 6, we update the policy parameters ψ by minimizing the KL divergence to the soft policy implicitly defined by the regularized Q-function [Haarnoja et al., 2018]:

$$\mathcal{L}_\pi(\psi) = \mathop{\mathbb{E}}_{\substack{z\sim\mathcal{U}(\mathcal{Z}) \\ s\sim\rho}} D_{KL}\left(\pi_\psi(\cdot|s,z), \frac{e^{\frac{Q^z(s,\cdot)}{1-\|z\|}}}{Z(s)}\right), \tag{9}$$

with $Z(s) = \int_{\mathcal{A}} \exp\big((1-\|z\|)^{-1} Q^z(s,a)\big) da$. As $Z(s)$ is independent of π_ψ, we can redefine an equivalent loss up to policy-independent factors as

$$\mathcal{L}_\pi(\psi) = \mathop{\mathbb{E}}_{z\sim\mathcal{U}(\mathcal{Z}),\ s\sim\rho,\ a\sim\pi_\psi(\cdot|s,z)} (1-\|z\|)[\log \pi_\psi(a|s,z) - Q_{\mathcal{H},\eta}(s,a,z)] - F_\theta(s,a,z)^\top z, \tag{10}$$

which is optimized with a standard reparameterization trick.

4.4 Inference

Soft FB returns a set of parameterized policies, $\{\pi_z\}_{z\in\mathcal{Z}}$; as shown in Corollary 4.2, this family is rich enough to include a solution for a large class of problems. However, solving a specific downstream task requires searching the policy class $\{\pi_z\}_{z\in\mathcal{Z}}$. Fortunately, policies are parameterized by a low-dimensional embedding $z \in \mathcal{Z}$, and the search process remains arguably efficient. In practice, paralleling Generalized Policy Improvement techniques in Farebrother et al. [2025], we resort to zero-order optimization. Given an objective $J^\pi = f(\mathrm{M}_\pi)$ in its analytical form at inference time, a good policy $\pi_{z^\star} \approx \text{argmax}_\pi f(\mathrm{M}_\pi)$ may be found through offline evaluation:

$$z^\star \approx \mathop{\text{argmax}}_{z\in\mathcal{Z}} f(\hat{\mathrm{M}}_z), \tag{11}$$

where $\hat{\mathrm{M}}_z$ is a sample-based estimate of the measure induced by policy π_z. As $\mathcal{Z}$ is relatively low-dimensional, zero-order, sampling-based optimization methods such as random shooting or CEM [Rubinstein, 1999] can be leveraged to find $z^\star$. Due to its simplicity, this is the solution we adopt in our empirical evaluation. We will consider two ways to recover sample-based estimates $\hat{\mathrm{M}}_z$ of policy dependent measures: implicit and explicit.

Implicit measure model Approximately recovering samples from the measure is possible by sampling states from the pre-training buffer distribution ρ with importance weights $F_z^\top B$: $s \sim \hat{\mathrm{M}}_{\mathcal{S}}^{\pi_z}$ where

$$\hat{\mathrm{M}}_{\mathcal{S}}^{\pi_z}(s) \propto \mathop{\mathbb{E}}_{s_0\sim\mu_0,\ a_0\sim\pi_\psi(s_0,z)} F_\theta(s_0,a_0,z)^\top B_\phi(s)\rho(s), \tag{12}$$

and then drawing actions $a \sim \pi_\psi(\cdot|s,z)$, thus only leveraging networks we previously trained.

Explicit measure model As the importance weights may be inaccurate, we can alternatively train a flow-based generative model of the successor measure, completely offline [Farebrother et al., 2025]. In practice, this may produce more accurate samples, at the cost of increased compute during pre-training. We detail training objectives in Appendix C. The variant of algorithms relying on generative models for inference will be marked by a subscript (e.g., SFB_{flow}).

Beside zero-order optimization, for some specific objectives Soft FB retains closed-form inference. Linear RL problems can be solved by computing $z^\star = BR/\|BR\|$ [Touati and Ollivier, 2021]. Moreover, a closed-form solution is also possible in the case of maximum entropy RL by computing the optimal embedding $z^\star$ for the corresponding linear RL instance, and rescaling it by $(F_z^\top z)/(F_z^\top z + 1)$. Furthermore, all imitation learning approaches described in [Pirotta et al., 2024] remain possible.

5 Experiments

We now complement the formal analysis of Soft FB with a detailed empirical evaluation. We will first present qualitative results in an easily interpretable setting, and then evaluate how Soft FB performs across zero-shot, general utility objectives, including imitation learning, pure exploration and constrained reinforcement learning. Finally, we include an evaluation on standard, high dimensional deep reinforcement learning benchmarks [Yarats et al., 2022]. We report additional empirical results in Appendix E.

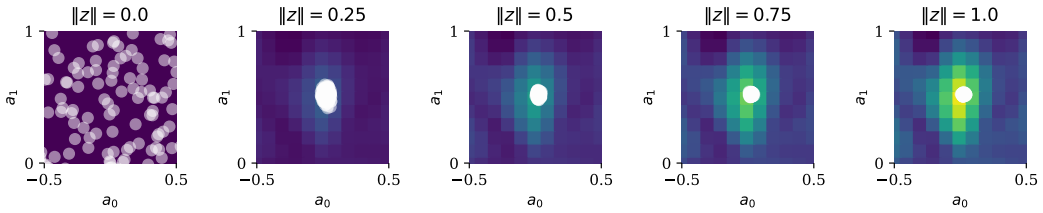


Figure 3: Qualitative evaluation of Soft FB in a didactic environment. White dots are samples from policies π_z over a 2D actions space, and the color map represents learned unregularized Q-values Q_R^z for each action $(F_\theta(s_0, a, z)^\top z)$. From left to right, we infer task embeddings z for a goal-reaching task, and scale them linearly. The policies conditioned on z become more deterministic as its norm increases. The same visualization for FB can be found in Appendix E.3.

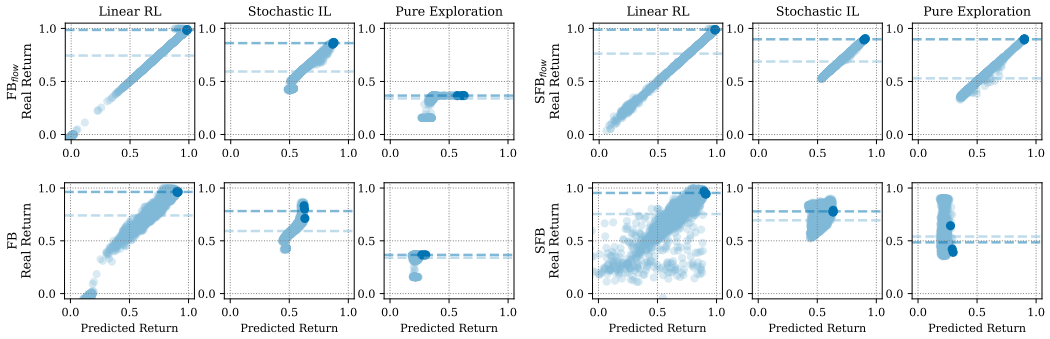


Figure 4: Quantitative results over several General RL objectives in a didactic environment. The x-axis and y-axis represent, respectively, offline performance estimates, and ground-truth performance in the environment. Each dot represents a policy sampled from each method across 3 seeds; for each seed, a darker dot marks the best policy according to offline evaluation. Horizontal lines represent the mean performance over points with the respective color. The policies captured by Soft FB (right) are more expressive, and the top policies affording to offline evaluation outperform, on average, those trained by FB (left). Explicit measure models (top) are more accurate.

5.1 Qualitative evaluation

Let us consider a simple, continuous environments, with

$$\mathcal{S} = \mathcal{A} = [-1, 1] \times [-1, 1], \ \mu_0 = \text{Dirac}(0, 0) \ \text{ and } \ P(s'|s, a) = \begin{cases} \text{Dirac}(a) & \text{if } s = (0, 0) \\ \text{Dirac}(s) & \text{otherwise.} \end{cases} \tag{13}$$

This didactic environment mimics a bandit-like MDP: the agent stays indefinitely in a state dictated by the action chosen in the initial state. A deterministic policy always moves the agent to the same state, and produces a low-entropy successor measure. While this may be optimal for goal-reaching, stochastic policies are generally needed to maximize other objectives.

We collect a dataset by executing actions uniformly at random, and train Soft FB over this data. We start by defining a simple goal-reaching task as $R(s|g) = \mathbf{1}_{\|s-g\|<0.2}$ and computing the corresponding reward embedding $z^\star = \frac{BR}{\|BR\|}$; for instance, we may fix $g = (0.0, 0.5)$. We may then introduce a controlled amount of entropy regularization through a scaled reward embedding $z = \alpha z^\star$ with $\alpha \in [0, 1]$, and retrieve an optimal policy for all corresponding maximum entropy RL objectives in a zero-shot fashion. If the norm of z is set to 1, the optimal deterministic policy should be retrieved, while very low norms should return near-uniform policies.

We observe this exact behavior in Figure 3. From left to right, we increase the norm of the task embedding, thus decreasing the degree of entropy regularization, and plot samples from $\pi_\psi(\cdot|s_0, z)$, as well as the entropy-*un*regularized action-value function $Q_R(s_0, \cdot, z) = F_\theta(s_0, \cdot, z)^\top z$ as a function of 2D actions. As expected, we observe that policies become more deterministic as the entropy regularization decreases, while still optimizing for the goal-reaching objective. The state-action value function is highest around the goal, and scales linearly in the norm of z by construction.

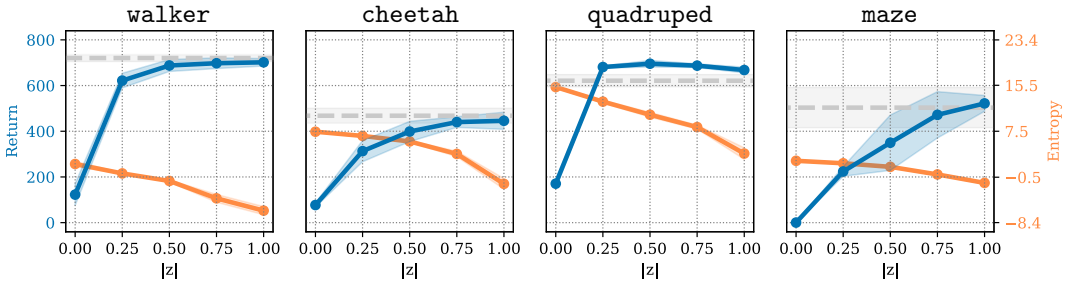


Figure 5: Zero-shot cumulative returns (in blue) and step-wise policy entropy (in orange) of Soft FB for different levels of entropy regularization in DMC, averaged over linear tasks. As entropy regularization decreases, returns generally improve, eventually matching the performance of FB (in grey), or surprisingly exceeding it in `quadruped`. Shaded areas represent 95% CIs over 5 seeds.

5.2 Quantitative evaluation

Within the environment outlined in the previous section, we now evaluate the performance of policies retrieved by Soft FB and FB, respectively. We thus consider a set of different General RL objectives: (i) standard linear RL, (ii) goal-reaching RL, imitation of a (iii) deterministic or (iv) stochastic agent, (v) pure exploration, (vi) constrained RL and (vii) robust RL. Both IL objectives are formulated through minimization of forward KL divergences between successor measures; specifics of each objective can be found in Appendix F. For each objective and algorithm, we search for the optimal policy according to the inference procedures outlined in Section 4.4: we sample 1024 task embeddings z, select the best one according to the measure model (implicit or explicit) and evaluate it in the environment. Table 1 compares the performance of Soft FB and FB, optionally relying on an explicit model for inference.

Table 1: Zero-shot performance over several General Utilities in the didactic environment. For each inference technique, the best policy retrieved by SFB constitutes, on average, a better solution. Results are averaged across 3 seeds, and bold when their 95% confidence intervals intersects with that of the best score.

	FB	FB$_{flow}$	SFB	SFB$_{flow}$
Linear RL	0.96 $\pm.01$	**0.99** $\pm.01$	0.95 $\pm.01$	**0.99** $\pm.01$
Goal-reaching RL	**1.00** $\pm.01$	**1.00** $\pm.01$	**1.00** $\pm.01$	**1.00** $\pm.01$
Deterministic IL	0.78 $\pm.06$	0.86 $\pm.01$	0.78 $\pm.01$	**0.90** $\pm.01$
Stochastic IL	0.67 $\pm.01$	0.77 $\pm.03$	0.79 $\pm.02$	**0.83** $\pm.01$
Pure Exploration	0.37 $\pm.01$	0.37 $\pm.01$	0.49 $\pm.13$	**0.90** $\pm.01$
Constrained RL	0.00 $\pm.01$	0.00 $\pm.01$	0.00 $\pm.01$	**0.65** $\pm.52$
Robust RL	0.01 $\pm.01$	0.79 $\pm.09$	0.39 $\pm.26$	**0.96** $\pm.02$
Average	0.54	0.68	0.63	**0.89**

As expected, the performance gap between FB and Soft FB on objectives admitting a deterministic solution is minimal (the first three); however, when deterministic policies are not sufficient, Soft FB achieves better performance. Furthermore, we observe that relying on an explicit measure models for policy evaluation increases performance, confirming the effectiveness of flow modeling [Farebrother et al., 2025] in stochastic settings. This is further confirmed in Figure 4, which relates offline performance estimates to ground-truth performance in the environment in three representative tasks. These two are more strongly correlated[5] when the former is estimated through an explicit measure model (SFB$_{flow}$, FB$_{flow}$); SFB$_{flow}$ performs the best as it also retrieves a richer class of policies (wider spread on the y-axis).

5.3 High-dimensional evaluation

So far, evaluation has largely focused on a continuous, yet low-dimensional setting, in which we demonstrated the expressiveness of policies retrieved by Soft FB. We now extend our evaluation to standard, high-dimensional continuous control benchmarks, in order to demonstrate scalability. We consider the Deepmind Control Suite (DMC, Tassa et al., 2018), which includes tasks across four embodiments, and follow the established evaluation protocol in [Touati et al., 2023] relying on exploratory data [Yarats et al., 2022]. We first consider classic linear objectives: Figure 5 tracks the return of policies trained with Soft FB and their entropy when conditioned on scaled task embedding z obtained by linear regression (i.e., the stan-

[5]We report rank-based correlation coefficients and numerical results in Appendix E.1.

dard task inference procedure for Forward-Backward algorithms). As expected, decreasing the vector norm induces more stochastic behavior, which results in performance degradation. However, we notice that, for near-zero entropy regularization, performance matches or exceeds that of policies trained by FB, confirming that the richness of policies does not come at the cost of performance. As standard objectives in DMC are exclusively linear, we additionally define several GUs: pure exploration (i.e. entropy maximization) and imitation of a deterministic or stochastic expert (i.e. KL minimization) over state and state-action measures, demonstrating walking or reaching behavior, as described in Appendix F.2. Appendix E.4 reports an in-depth description of these objectives, and a nuanced discussion of results, including additional baselines. For brevity, we also compare the performance of Soft FB and FB with explicit measure models in Table 2. While performance is explainably similar for linear tasks, we find that a significant gap appears as soon as a deterministic optimal policy does not exist, confirming the effectiveness of Soft FB in producing a richer class of policies, with strong performance beyond linear rewards. In particular, we find that the stochastic policies of SoftFB are capable of "wriggling" more, and thus achieve better exploration. For imitation objectives, while both FB and SoftFB can follow the demonstration, SoftFB reduces the tracking error on average by not committing to a single deterministic action.

Table 2: Zero-shot performance on GUs in DMC. We consider entropy maximization and KL minimization with respect to a deterministic or stochastic expert (i.e. an optimal policy for a linear task, e.g. `walker-walk`, with injected Gaussian noise in the stochastic case). Scores are averages over 5 seeds, with 95% confidence intervals and bold numbers signaling overlaps.

		FB$_{flow}$	Soft FB$_{flow}$
walker	$\mathcal{H}(\mathrm{M}^{\pi}_{\mathcal{S}})$	12.56 $\pm.91$	**13.97** $\pm.25$
	$\mathcal{H}(\mathrm{M}^{\pi})$	9.86 $\pm.10$	**18.01** $\pm.03$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{stoch}}_{\mathcal{S}})$	**-5.35** $\pm.24$	**-5.25** $\pm.14$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{stoch}})$	-9.40 $\pm.17$	**-1.48** $\pm.11$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{det}}_{\mathcal{S}})$	**-5.69** $\pm.27$	**-5.69** $\pm.02$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{det}})$	-9.43 $\pm.18$	**-1.53** $\pm.09$
cheetah	$\mathcal{H}(\mathrm{M}^{\pi}_{\mathcal{S}})$	11.74 $\pm.43$	**13.63** $\pm.10$
	$\mathcal{H}(\mathrm{M}^{\pi})$	11.68 $\pm.89$	**17.83** $\pm.07$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{stoch}}_{\mathcal{S}})$	-5.56 $\pm.31$	**-4.56** $\pm.31$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{stoch}})$	-7.25 $\pm.76$	**-1.02** $\pm.08$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{det}}_{\mathcal{S}})$	-5.45 $\pm.65$	**-4.38** $\pm.18$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{det}})$	-7.55 $\pm.79$	**-1.23** $\pm.08$
quadruped	$\mathcal{H}(\mathrm{M}^{\pi}_{\mathcal{S}})$	13.37 $\pm.43$	**14.43** $\pm.24$
	$\mathcal{H}(\mathrm{M}^{\pi})$	10.11 $\pm.08$	**19.10** $\pm.01$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{stoch}}_{\mathcal{S}})$	-5.44 $\pm.28$	**-4.64** $\pm.20$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{stoch}})$	-9.92 $\pm.16$	**-0.76** $\pm.04$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{det}}_{\mathcal{S}})$	-5.61 $\pm.30$	**-4.52** $\pm.16$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{det}})$	-9.94 $\pm.20$	**-0.79** $\pm.02$
maze	$\mathcal{H}(\mathrm{M}^{\pi}_{\mathcal{S}})$	**10.83** $\pm.56$	**11.22** $\pm.70$
	$\mathcal{H}(\mathrm{M}^{\pi})$	10.52 $\pm.66$	**15.54** $\pm.16$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{stoch}}_{\mathcal{S}})$	**-6.49** $\pm.36$	**-5.25** $\pm.90$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{stoch}})$	-6.45 $\pm.61$	**-1.39** $\pm.31$
	$-KL(\mathrm{M}^{\pi}_{\mathcal{S}};\mathrm{M}^{\pi^{det}}_{\mathcal{S}})$	-6.57 $\pm.61$	**-5.07** $\pm.69$
	$-KL(\mathrm{M}^{\pi};\mathrm{M}^{\pi^{det}})$	-7.27 $\pm.58$	**-2.51** $\pm.35$

6 Related Works

We present an essential discussion of the literature here, and extend it in Appendix B.

Zero-shot Reinforcement Learning In its purest instantiation, reinforcement learning is centered on optimizing a single, scalar reward function [Sutton and Barto, 1998]. Despite its flexibility [Silver et al., 2021], this formulation does not adapt to changes in the objective without re-training. The successor feature framework [Dayan, 1993, Barreto et al., 2017] tackles this problem by specifying or learning policy representations that enable efficient evaluation and optimization; in this context, Hunt et al. [2019] formulate a related entropy-regularized variant, but remain focused on linear rewards. Forward-backward representations [Blier et al., 2021, Touati and Ollivier, 2021, Touati et al., 2023] leverage low-rank approximation of occupancy measures to retrieve suitable representations for any reward function, and have been scaled to high-dimensional environments [Tirinzoni et al., 2024, Li et al., 2026]. However, forward-backward algorithms remain constrained to linear RL problems and, practically, deterministic policies, which becomes a limitation for more complex objectives, involving multimodal data distributions or exploration objectives. Our work directly addresses this limitation.

Non-linear Reinforcement Learning Much of the existing RL machinery builds upon the assumption that the objective may be broken down in additive terms, each of which may be traced back to a single state-action pair [Sutton and Barto, 1998]. This is referred to as RL with additive rewards, or Linear RL. Convex RL [Zahavy et al., 2021, Geist et al., 2022, Mutti et al., 2022] encompasses a richer class of objective, such as pure exploration [Hazan et al., 2019], active learning in MDPs [Mutny et al., 2023] and distribution matching [Kostrikov et al., 2020, Rupf et al., 2025]. A further generalization

leads to RL with General Utilities [Zhang et al., 2020], for which we provide more related works in Appendix B due to space constraints. Solutions to Convex or General Utilities may be found by iterative procedures [Geist et al., 2022], which involve a low-level MDP-solving routine, or adversarial optimization schemes [Zahavy et al., 2021]. This generally produces mixture, non-Markovian policies, while Soft FB returns a single, Markov policy. Zero-shot methods have been applied to non-linear problems before [Pirotta et al., 2024], but are restricted to imitation learning and only take non-additivity into account during inference. To the best of our knowledge, our work is the first in exploring zero-shot solutions to arbitrary General RL problems in a principled and scalable way.

7 Conclusion

At its core, this work proposes a novel extension of zero-shot reinforcement learning beyond linear rewards. We introduce a soft forward-backward algorithm, which leverages a simple entropy regularization mechanism to capture stochastic behaviors in a dynamical system. At inference, the space of behaviors can be searched efficiently to retrieve an approximately optimal policy for an arbitrary GU.

Limitations and Future Work Theorem 4.2 – the core of our contribution – is an existence result: under exact optimization and sufficient latent dimension, the family of retrieved policies includes a near-optimal one. This does not guarantee that search will find that policy, which practically depends on test-time search, and particularly on measure estimation. This limitation is visible for instance in Figure 5: as the assumptions for Theorem 4.2 are in practice violated, SoftFB does not exactly retrieve all full-support policies, and it does not reach the maximum return theoretically possible in DMC. Furthermore, the formal guarantees require infinite-dimensional task representations and expressive actors: in practical settings, a narrower set of policies will be retrieved. Formally studying the properties of this set represents an important direction for future work. Accurate search over policies relies on precise modeling of successor measures, as evident in Table 1: zero-order search is non-trivial, and depends on the quality of the measure model. While in practice a dedicated generative model suffices, it also increases the computational costs during training. Furthermore, while we found simple zero-order optimization to be sufficient for search among policies, more involved optimization techniques may further scale this approach to higher-dimensional representation spaces. A further interesting extension would generalize our framework to arbitrary successor-feature-based algorithms.

Soft FB introduces a first-of-its-kind extension of zero-shot RL beyond linear rewards. We hope that this work represents a step forward in bringing fundamental works on general utilities closer to application in practical settings.

Acknowledgments and Disclosure of Funding

We would like to thank Andrea Tirinzoni, Núria Armengol Urpí, Marin Vlastelica, Pavel Kolev, Yifan Hu and Ehsan Sharifian for the fruitful discussions and valuable feedback. Marco Bagatella is supported by the Max Planck ETH Center for Learning Systems. This project was supported in part by the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40 180545.

References

Siddhant Agarwal, Harshit Sikchi, Peter Stone, and Amy Zhang. Proto successor measure: Representing the space of all possible solutions of reinforcement learning. In *ICML*, 2025.

Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. Solving rubik's cube with a robot hand. In *arXiv preprint arXiv:1910.07113*, 2019.

Marcin Andrychowicz, Dwight Crow, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *NeurIPS*, 2017.

Marco Bagatella, Matteo Pirotta, Ahmed Touati, Alessandro Lazaric, and Andrea Tirinzoni. Td-jepa: Latent-predictive representations for zero-shot reinforcement learning. In *ICLR*, 2026.

Anas Barakat, Ilyas Fatkhullin, and Niao He. Reinforcement learning with general utilities: Simpler variance reduction and large state-action space. In *ICML*, 2023.

Anas Barakat, Souradip Chakraborty, Peihong Yu, Pratap Tokekar, and Amrit Singh Bedi. On the global optimality of policy gradient methods in general utility reinforcement learning. In *NeurIPS*, 2025.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In *NeurIPS*, 2017.

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

Adrien Bolland, Gaspard Lambrechts, and Damien Ernst. Off-policy maximum entropy rl with future state and action visitation measures. In *arXiv preprint arXiv:2412.06655*, 2024.

Edoardo Cetin, Ahmed Touati, and Yann Ollivier. Finer behavioral foundation models via auto-regressive features and advantage weighting. In *arXiv preprint arXiv:2412.04368*, 2024.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. In *Neural computation*, 1993.

Riccardo De Santi, Manish Prajapat, and Andreas Krause. Global reinforcement learning: Beyond linear and convex rewards via submodular semi-gradient methods. In *ICML*, 2024.

Alejandro Escontrela, Xue Bin Peng, Wenhao Yu, Tingnan Zhang, Atil Iscen, Ken Goldberg, and Pieter Abbeel. Adversarial motion priors make good substitutes for complex reward functions. In *IROS*, 2022.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. In *NeurIPS*, 2022.

Jesse Farebrother, Matteo Pirotta, Andrea Tirinzoni, Rémi Munos, Alessandro Lazaric, and Ahmed Touati. Temporal difference flows. In *ICML*, 2025.

Kevin Frans, Seohong Park, Pieter Abbeel, and Sergey Levine. Unsupervised zero-shot reinforcement learning via functional reward encodings. In *ICML*, 2024.

Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *ICML*, 2018.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *ICML*, 2019.

Matthieu Geist, Julien Pérolat, Mathieu Laurière, Romuald Elie, Sarah Perrin, Olivier Bachem, Rémi Munos, and Olivier Pietquin. Concave utility reinforcement learning: The mean-field game viewpoint. In *AAMAS*, 2022.

Dibya Ghosh, Chethan Anand Bhateja, and Sergey Levine. Reinforcement learning from passive data via latent intentions. In *ICML*, 2023.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *ICML*, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018.

Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *ICML*, 2019.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *NeurIPS*, 2016.

Audrey Huang and Nan Jiang. Occupancy-based policy gradient: Estimation, convergence, and optimality. In *NeurIPS*, 2024.

Jonathan Hunt, Andre Barreto, Timothy Lillicrap, and Nicolas Heess. Composing entropic policies using divergence correction. In *ICML*, 2019.

Hisham Husain, Kamil Ciosek, and Ryota Tomioka. Regularized policies are reward robust. In *AISTATS*, 2021.

Arnav Kumar Jain, Lucas Lehnert, Irina Rish, and Glen Berseth. Maximum state entropy exploration using predecessor and successor representations. In *NeurIPS*, 2023.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. In *Machine learning*, 2002.

Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. In *ICLR*, 2020.

Navdeep Kumar, Kaixin Wang, Kfir Levy, and Shie Mannor. Policy gradient for reinforcement learning with general utilities. In *arXiv preprint arXiv:2210.00991*, 2022.

Joonho Lee, Lukas Schroth, Victor Klemm, Marko Bjelonic, Alexander Reske, and Marco Hutter. Evaluation of constrained reinforcement learning algorithms for legged locomotion. In *IROS*, 2024.

Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. In *arXiv preprint arXiv:1906.05274*, 2020.

Yitang Li, Zhengyi Luo, Tonghe Zhang, Cunxi Dai, Anssi Kanervisto, Andrea Tirinzoni, Haoyang Weng, Kris Kitani, Mateusz Guzek, Ahmed Touati, Alessandro Lazaric, Matteo Pirotta, and Guanya Shi. BFM-zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning. In *ICLR*, 2026.

Jason Yecheng Ma, Jason Yan, Dinesh Jayaraman, and Osbert Bastani. Offline goal-conditioned reinforcement learning via f-advantage regression. In *NeurIPS*, 2022.

George Mathew, Suresh Kannan, Amit Surana, Sanjay Bajekal, and Konda R. Chevva. Experimental implementation of spectral multiscale coverage and search algorithms for autonomous uavs. In *AIAA*, 2013.

Marius Memmel, Andrew Wagenmaker, Chuning Zhu, Patrick Yin, Dieter Fox, and Abhishek Gupta. Asid: Active exploration for system identification in robotic manipulation. In *ICLR*, 2024.

Mojmir Mutny, Tadeusz Janik, and Andreas Krause. Active exploration via experiment design in markov chains. In *AISTATS*, 2023.

Mirco Mutti, Riccardo De Santi, Piersilvio De Bartolomeis, and Marcello Restelli. Challenging common assumptions in convex reinforcement learning. In *NeurIPS*, 2022.

Mirco Mutti, Riccardo De Santi, Piersilvio De Bartolomeis, and Marcello Restelli. Convex reinforcement learning in finite trials. In *JMLR*, 2023.

Matteo Pirotta, Andrea Tirinzoni, Ahmed Touati, Alessandro Lazaric, and Yann Ollivier. Fast imitation via behavior foundation models. In *ICLR*, 2024.

Vitchyr H. Pong, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine. Skew-fit: State-covering self-supervised reinforcement learning. In *ICML*, 2020.

Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. In *Methodology and Computing in Applied Probability*, 1999.

Thomas Rupf, Marco Bagatella, Nico Guertler, Jonas Frey, and Georg Martius. Zero-shot offline imitation learning via optimal transport. In *ICML*, 2025.

Thomas Rupf, Marco Bagatella, Marin Vlastelica, and Andreas Krause. Optimistic task inference for behavior foundation models. In *ICLR*, 2026.

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *ICML*, 2015.

Harshit Sikchi, Amy Zhang, and Scott Niekum. Imitation from arbitrary experience: A dual unification of reinforcement and imitation learning methods. In *Workshop on Reincarnating Reinforcement Learning at ICLR*, 2023.

Harshit Sikchi, Siddhant Agarwal, Pranaya Jajoo, Samyak Parajuli, Caleb Chuck, Max Rudolph, Peter Stone, Amy Zhang, and Scott Niekum. Rl zero: zero-shot language to behaviors without any supervision. In *7th Robot Learning Workshop: Towards Robots with Human-Level Abilities @ ICLR*, 2025a.

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

David Silver, Satinder Singh, Doina Precup, and Richard S Sutton. Reward is enough. In *Artificial intelligence*. Elsevier, 2021.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT Press, 1998.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. In *arXiv preprint arXiv:1801.00690*, 2018.

Stephen Tian, Suraj Nair, Frederik Ebert, Sudeep Dasari, Benjamin Eysenbach, Chelsea Finn, and Sergey Levine. Model-based visual planning with self-supervised functional distances. In *ICLR*, 2021.

Andrea Tirinzoni, Ahmed Touati, Jesse Farebrother, Mateusz Guzek, Anssi Kanervisto, Yingchen Xu, Alessandro Lazaric, and Matteo Pirotta. Zero-shot whole-body humanoid control via behavioral foundation models. In *NeurIPS Workshop on Open-World Agents*, 2024.

Faraz Torabi, Garrett Warnell, and Peter Stone. Recent advances in imitation learning from observation. In *arXiv preprint arXiv:1905.13566*, 2019.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. In *NeurIPS*, 2021.

Ahmed Touati, Jérémy Rapin, and Yann Ollivier. Does zero-shot reinforcement learning exist? In *ICLR*, 2023.

Núria Armengol Urpí, Marin Vlastelica, Georg Martius, and Stelian Coros. Epistemically-guided forward-backward exploration. In *RLC*, 2025.

Qing Wang, Sanjeev R Kulkarni, and Sergio Verdú. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. In *IEEE Transactions on Information Theory*, 2009.

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

Raymond Yu, William Huey, Mustafa Mukadam, Anusha Nagabandi, and Abhishek Gupta. Support-constrained rl enables real-world policy improvement without real-world experience. In *arXiv preprint arXiv:2606.27475*, 2026.

Tom Zahavy, Brendan O'Donoghue, Guillaume Desjardins, and Satinder Singh. Reward is enough for convex mdps. In *NeurIPS*, 2021.

Junyu Zhang, Alec Koppel, Amrit Singh Bedi, Csaba Szepesvari, and Mengdi Wang. Variational policy gradient method for reinforcement learning with general utilities. In *NeurIPS*, 2020.

Junyu Zhang, Chengzhuo Ni, Csaba Szepesvari, and Mengdi Wang. On the convergence and sample efficiency of variance-reduced policy gradient method. In *NeurIPS*, 2021.

Chongyi Zheng, Seohong Park, Sergey Levine, and Benjamin Eysenbach. Intention-conditioned flow occupancy models. In *ICLR*, 2026.

Brian D. Ziebart, Andrew L. Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI-08)*, pages 1433–1438. AAAI Press, 2008.

A Theoretical results and proofs

This section starts by reporting proofs for the main formal results. We later present some counterexamples, examples, and additional insights in smoothness and interpolation properties that arise with Soft FB's entropy regularization.

A.1 Proofs

This section contains proofs for Theorems 4.1 and 4.2; the latter is preceded by a useful intermediate Lemma.

Theorem 4.1. *For an arbitrary bounded reward vector $R \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, if both Equations 1 and 5 hold for all $z \in \mathcal{Z}$, π_{BR} is the optimal maximum entropy policy with respect to R: $M^{\pi_{BR}}(R + \mathcal{H}_{\pi_{BR}}) = \max_{\pi \in \Pi} M^{\pi}(R + \mathcal{H}_{\pi})$.*

Proof. This result can be proven through a direct generalization of the results in Haarnoja et al. [2018] and Touati and Ollivier [2021]. Starting from Equation 5, setting $z = BR$, we can observe that

$$\pi_z \propto \exp(F_z^\top z + M^z \mathcal{H}_{\pi_z}) \tag{14}$$

$$\overset{z=BR}{=} \exp(F_z^\top BR + M^z \mathcal{H}_{\pi_z}) \tag{15}$$

$$\overset{Eq.1}{=} \exp\big(M^z(R + \mathcal{H}_{\pi_z})\big) \tag{16}$$

$$= \exp(Q^z_{\text{soft}}). \tag{17}$$

Up to normalization constants, the policy satisfies the optimality criterion from Theorem 1 in Haarnoja et al. [2017], and is thus the optimal maximum entropy policy, i.e. $\pi_z = \text{argmax}_\pi \, M^\pi(R + \mathcal{H}_\pi)$. □

Lemma A.1. *Let $\tilde{\Pi}_z$ be the set of policies retrieved by Soft FB and $\pi \in \Pi$ be an arbitrary Markov policy. If both Equations 1 and 3 hold for all $z \in \mathcal{Z}$, and $\pi(a|s) > 0$ for every $(s, a) \in \mathcal{S} \times \mathcal{A}$ (i.e., π has complete support), then $\pi \in \tilde{\Pi}_z$.*

Proof. We will show that a maximum entropy problem admitting π as its optimal policy can be constructed, which will imply that π is part of the set of solutions to entropy-regularized instances retrieved by Soft FB by Theorem 4.1. Let $R(s, a) = \log \pi(a|s)$: since π has complete support, $R(s, a)$ is bounded. Let us now consider its corresponding maximum entropy objective:

$$J^{\pi'}_{\mathcal{H}}(R) = \mathop{\mathbb{E}}_{\mu_0, P, \pi'} \sum_{t=0}^{\infty} \gamma^t \Big(R(s_t, a_t) + \mathcal{H}\big(\pi'(\cdot|s_t)\big) \Big) \tag{18}$$

$$= \mathop{\mathbb{E}}_{\mu_0, P, \pi'} \sum_{t=0}^{\infty} \gamma^t \mathop{\mathbb{E}}_{a_t \sim \pi'(\cdot|s_t)} \Big(\log \pi(a_t|s_t) - \log \pi'(a_t|s_t) \Big) \tag{19}$$

$$= \mathop{\mathbb{E}}_{\mu_0, P, \pi'} \sum_{t=0}^{\infty} \gamma^t \Big(- D_{\text{KL}}\big(\pi'(\cdot|s_t) || \pi(\cdot|s_t)\big) \Big) \leq 0. \tag{20}$$

This objective is an expected, discounted sum of negative KL distances, and therefore non-positive. As $D_{\text{KL}}(p||p) = 0$, $J^{\pi}_{\mathcal{H}}(R^\star) = 0$, and π is the optimal policy for the maximum entropy objective with reward R. Theorem 4.1 guarantees that there exist a reward embedding $z \in \mathcal{Z}$ recovering a solution π_z for each maximum entropy problem; therefore, there exist a reward embedding $z \in \mathcal{Z}$ such that $\pi_z = \text{argmax}_{\pi' \in \Pi} J^{\pi'}_{\mathcal{H}}(R) = \pi$, and $\pi \in \tilde{\Pi}_z$. □

Theorem 4.2. *Let f be an arbitrary differentiable function of occupancy measures, and $\tilde{\Pi}_z$ be the set of policies retrieved by Soft FB. If both Equations 1 and 5 hold for all $z \in \mathcal{Z}$, for any $\epsilon > 0$ there exists a reward embedding $z' \in \mathcal{Z}$ such that $\pi_{z'} \in \tilde{\Pi}_z$ and $\max_{\pi \in \Pi} f(\mathrm{M}^\pi) - f(\mathrm{M}^{\pi_{z'}}) < \epsilon$.*

Proof. This proof will rely on the construction of a sequence of policies. We will first show that each policy is retrieved by Soft FB, and then that this sequence gets arbitrarily close to the optimum.

Let us consider an arbitrary differentiable objective f and its optimal policy $\pi^\star = \text{argmax}_{\pi \in \Pi} f(\mathrm{M}^\pi)$. Let $\bar{\pi} = \mathcal{U}(\mathcal{A})$ denote the uniform policy (i.e., the policy choosing each action with equal probability). We can construct a sequence of increasingly stochastic policies as a linear interpolation of the two through a parameter $\alpha \in (0, 1]$:

$$\pi_\alpha(a|s) := (1-\alpha)\pi^\star(a|s) + \alpha\bar{\pi}(a|s). \tag{21}$$

For all $\alpha > 0$ and $(s, a) \in \mathcal{S} \times \mathcal{A}$, $\pi_\alpha(a|s) > \alpha\bar{\pi}(a|s) > 0$. Thus, each policy π_α has full support. Through Lemma A.1 we can conclude that $\pi_\alpha \in \tilde{\Pi}_z$, that is the set of policies retrieved by Soft FB. We will now show that, as $\alpha \to 0$, π_α is approximately close to the optimum for f.

We can first bound the distance in action space to the optimal policy $\pi^\star$ for each $s \in \mathcal{S}$, that is

$$\|\pi^\star(\cdot|s) - \pi_\alpha(\cdot|s)\|_1 = \sum_{a \in \mathcal{A}} |\pi^\star(a|s) - (1-\alpha)\pi^\star(a|s) - \alpha\bar{\pi}(a|s)| \tag{22}$$

$$= \sum_{a \in \mathcal{A}} \alpha|\pi^\star(a|s) - \bar{\pi}(a|s)| \tag{23}$$

$$\leq 2\alpha, \tag{24}$$

where the last inequality follows from the fact that $\pi^\star$ and $\bar{\pi}$ are probability distributions, and thus bounded in $[0, 1]$. We can now apply the Simulation Lemma (Kearns and Singh [2002], cf. Appendix A.5) to bound the distance between marginalized successor measures of the two policies:

$$\|\mathrm{M}^{\pi^\star} - \mathrm{M}^{\pi_\alpha}\|_1 \leq \frac{1}{1-\gamma} \max_{s \in \mathcal{S}} \|\pi^\star(\cdot|s) - \pi_\alpha(\cdot|s)\|_1 \leq \frac{2\alpha}{1-\gamma} \tag{25}$$

The last step requires bounding the difference $f(\mathrm{M}^\pi) - f(\mathrm{M}^{\pi_\alpha})$, which is possible through uniform continuity. Let us consider the set of all valid occupancy measures $\mathcal{K} = \{M^\pi\}_{\pi \in \Pi} \subset \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$. Each element of $\mathcal{K}$ must respect the following linear constraints: $M(s, a) \geq 0$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$ (i.e., probabilities must be non-negative), (ii) $\sum_{s,a \in \mathcal{S} \times \mathcal{A}} M(s, a) = 1$ (i.e., M is a valid probability distribution), and (iii) $\sum_a \mathrm{M}(s, a) = (1-\gamma)\mu_0(s) + \gamma \sum_{s',a'} P(s|s', a')\mathrm{M}(s', a')$ (i.e., Bellman flow constraints respecting the dynamics of the MDP). $\mathcal{K}$ is thus closed and bounded, and by consequence a compact set. Furthermore, differentiability of f implies continuity of f: by the Heine Cantor Theorem (cf. Appendix A.5), a continuous function on a compact set is uniformly continuous. This is equivalent to the existence of a continuous, monotonic function $\omega_f : [0, \infty) \to [0, \infty)$ with $\lim_{\delta \to 0} \omega_f(\delta) = 0$ (i.e., a modulus of continuity) such that

$$|f(\mathrm{M}) - f(\mathrm{M}')| \leq \omega_f(\|\mathrm{M} - \mathrm{M}'\|_1) \tag{26}$$

Combining the uniform continuity property in Equation 26 with the bound on measures in Equation 25, we finally have

$$|f(\mathrm{M}^{\pi^\star}) - f(\mathrm{M}^{\pi_\alpha})| \leq \omega_f(\|\mathrm{M}^{\pi^\star} - \mathrm{M}^{\pi_\alpha}\|_1) \leq \omega_f\left(\frac{2\alpha}{1-\gamma}\right) \tag{27}$$

Since $\omega_f(x) \to 0$ as $x \to 0$, for any precision ϵ, one can find a sufficiently small α such that $\omega_f(2\alpha(1-\gamma)^{-1}) < \epsilon$, and thus π_α achieves an arbitrarily close objective value to the optimum of f. As $\pi_\alpha \in \tilde{\Pi}_z$, we can conclude that there exist a policy among those retrieved by Soft FB that optimizes f arbitrarily well. □

A.2 Extended counterexample

Section 3 anticipates a counterexample to clarify why policies retrieves by FB may fail to optimize a given GU. We will now formalize this counterexample in detail, while providing a visual description in Figure 6.

Let us consider an MDP $\mathcal{M}$ with a single state $\mathcal{S} = \{s_0\}$ and two actions $\mathcal{A} = \{a_0, a_1\}$. Let us also consider representations of size $d = 2$ (i.e., $\mathcal{Z} = \mathbb{R}^2$). Let us now set the backward representation matrix as the identity matrix (i.e., $B = I$). By doing so, we can take (the transposes of) policy-conditional successor measure matrices M_z as forward representation matrices (i.e., $F_z^\top = M_z$), and trivially satisfy Equation 1 as $F_z^\top B = M_z I = M_z$ for any policy π_z. Furthermore, setting B to the identity matrix projects all rewards functions $R \in \mathbb{R}^2$ to identical reward embeddings $z = BR = R$.

It remains to show that there is a family of policies that satisfies Equation 3 as well. We thus choose the family of z-parameterized policies $\Pi_z = \{\pi_z\}_{z \in \mathcal{Z}}$, where the policy parameterized by z picks the action indexed by the largest of the two elements of z: $\pi_z(\cdot) = a_0$ if $z_0 > z_1$, else a_1. Intuitively, as $z = R$, these policies are simply choosing the action associated with the largest reward. In order to check whether these policies satisfy Equation 3, we need to calculate its right hand-side; as $F_z^\top = M_z$, it suffices to compute the successor measure for all policies. Let us start by considering $z_0 > z_1$: in this case $\pi_z(\cdot) = a_0$. When starting from (s, a_0), the other state-action pair is never visited: $M_z(s, a_0, s, a_0) = 1$. When starting from (s, a_1), this state-action pair is only visited at the beginning of the trajectory, and never again: $M_z(s, a_1, s, a_1) = (1-\gamma) := C$ with $0 \leq C < 1$. By considering that M_z is a stochastic matrix, and similarly working through the case in which $z_0 \leq z_1$, we have

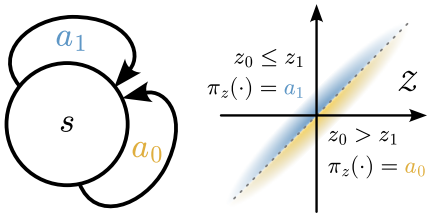


Figure 6: The simple MDP referred to for the counterexample (left); we consider a 2-dimensional representation space $\mathcal{Z}$ (right), and policies π_z that output the first or second action depending on the semi-plane their parameter z belongs to (right).

$$F_z^\top = M_z = \begin{bmatrix} 1 & 0 \\ 1-C & C \end{bmatrix} \text{ if } z_0 > z_1, \text{ else } \begin{bmatrix} C & 1-C \\ 0 & 1 \end{bmatrix}. \tag{28}$$

We can now verify directly that Equation 3 holds. In the case in which $z_0 > z_1$, we have

$$F_z^\top z = \begin{bmatrix} 1 & 0 \\ 1-C & C \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \end{bmatrix} = \begin{bmatrix} z_0 \\ (1-C)z_0 + Cz_1 \end{bmatrix}. \tag{29}$$

Since $(1-C)z_0 + Cz_1 \overset{z_1 < z_0}{\leq} (1-C)z_0 + Cz_0 = z_0$, then the policy optimizes its own Q-function: $\pi_z(\cdot) = a_0 = \text{argmax}_{\mathcal{A}} F_z^\top z$. Similarly, in the case in which $z_0 \leq z_1$, we have

$$F_z^\top z = \begin{bmatrix} C & 1-C \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z_0 \\ z_1 \end{bmatrix} = \begin{bmatrix} Cz_0 + (1-C)z_1 \\ z_1 \end{bmatrix}. \tag{30}$$

This time, $Cz_0 + (1-C)z_1 \overset{z_1 \geq z_0}{\leq} C(z_1) + (1-C)z_1 = z_1$, and the policy still optimizes its own Q-function: $\pi_z(\cdot) = a_1 = \text{argmax}_{\mathcal{A}} F_z^\top z$. Since these two cases ($z_0 > z_1$ and $z_0 \leq z_1$) cover all policies $\pi_z \in \Pi_z$, Equation 3 holds.

All policies π_z are strictly deterministic. We can now consider a convex RL objective such as *pure exploration* over states and actions, i.e. $J^\pi = f(\mathrm{M}^\pi) = \mathcal{H}(\mathrm{M}^\pi)$, where $\mathcal{H}(\cdot)$ denotes Shannon entropy over state-action pairs. For this objective, all policies in Π_z are actually *minimizers*: $J^{\pi_z} = 0$ for all $z \in \mathcal{Z}$. The optimal policy is uniform over the two actions (i.e., $\pi^\star(\cdot) = \mathcal{U}(\mathcal{A})$), achieves $J^{\pi^\star} = \log 2$ and does not belong to Π_z. This thus represents an instance in which the set of policies retrieved by FB may not include the optimal policy for some GU.

A.3 Didactic Example

This section derives a possible policy set retrieved by SoftFB in the MDP described in Figure 6 for illustrative purposes.

The MDP in question has one state s and two actions $[a_0, a_1]$. For each $z \in \mathbb{R}^2$, let us consider the policy $\pi_z := [x, 1-x]^\top$, and compute its occupancy measure as

$$M^z = (1-\gamma)I + \gamma \mathbf{1}\pi_z^\top. \tag{31}$$

We will set $B := I \in \mathbb{R}^{2\times 2}$, and ensure Equation 1 holds by setting $F_z^\top := M^z$. The exponent in Equation 5 simplifies as

$$F_z^\top z + M^z \mathcal{H}_z = (1-\gamma)z + (\gamma \pi_z^\top z + h(x))\mathbf{1}, \tag{32}$$

where $h(x) = -x \log x - (1-x)\log(1-x)$. Equation 5 states that

$$\pi \propto \exp((1-\gamma)z + (\gamma \pi_z^\top z + h(x))\mathbf{1}) = C \exp((1-\gamma)z), \tag{33}$$

since the second part is constant (C) across the policy vector. We have then that

$$\pi_z = \frac{1}{e^{(1-\gamma)z_0} + e^{(1-\gamma)z_1}} [e^{(1-\gamma)z_0}, e^{(1-\gamma)z_1}]^\top. \tag{34}$$

By substituting π_z into $F_z^\top$ we complete the example.

A.4 Additional remarks

One well-known property of entropy regularization in RL is that of inducing smoothness in the mapping from rewards to optimal behaviors [Geist et al., 2019, Husain et al., 2021]. This property is reflected in Soft FB, as we discuss in this section.

Remark A.2. Let us assume that Equations 1 and 5 hold, and that backward representations B are full-rank. Let us now consider the map $g : \mathcal{Z} \to \mathcal{M}$ from the reward representation space $\mathcal{Z}$ to the space of feasible occupancy measures $\mathcal{M}$, such that $g(z) = M^{\pi_z}$. The map g is $\mathcal{C}^\infty$ (smooth).

Proof. The map g is a composition of several functions: we will show that each of these functions is, in turn, smooth.

- As established in Theorem 4.1, each policy $\pi_z \in \tilde{\Pi}_z$ is the regularized optimal policy for reward R if $z = BR$. We can thus project the reward representation z to its reward as $R = (B^\top B)^{-1} B^\top z$, where $B^\top B$ is invertible as B is full rank. This map $z \mapsto R$ is linear and thus also smooth in z.
- Let us define the optimal entropy-regularized state-action value function for reward R: $Q^\star = \max_{\pi \in \Pi} M^\pi (R + \mathcal{H}_\pi)$. This is the unique fixed point of the Soft Bellmann optimality operator $\mathcal{T}_r$. We may then define its root finding function as $G(Q, r) = Q - \mathcal{T}_r(Q) = 0$. Component-wise for each state-action pair (s, a) it takes the form

$$G_{(s,a)}(Q, r) = Q(s, a) - \left(r(s, a) + \gamma \sum_{s'} P(s'|s, a) \log \sum_{a'} \exp Q(s', a') \right) \tag{35}$$

  which, as a composition of linear function and the LogSumExp function, is itself smooth. In order to apply the Implicit Function Theorem, we examine the Jacobian of G w.r.t. Q. The partial derivative takes the form

$$\frac{\partial G_{(s,a)}}{\partial Q_{(s',a')}} = \delta_{(s,a)=(s',a')} - \gamma P(s'|s, a) \pi_{\text{soft}}(a'|s') \;\text{ where }\; \pi_{\text{soft}}(a'|s') = \frac{\exp Q(s', a')}{\sum_b \exp Q(s', b)}. \tag{36}$$

  In matrix form, this is $J_Q = \mathrm{I} - \gamma P^{\pi_{\text{soft}}}$, which has eigenvalues with norm of at least $1 - \gamma > 0$, making it non-singular. We can thus apply the Implicit Function Theorem, and confirm that the map $R \mapsto Q^\star$ is unique and smooth.
- By Theorem 4.1 $\pi_z = \text{softmax}(Q^\star)$: this map is smooth in $Q^\star$ because of the smoothness of the softmax operator.
- Finally we have $M^z = (\mathrm{I} - \gamma P^{\pi_z})^{-1}$. First, we notice that $\mathrm{I} - \gamma P^{\pi_z}$ is smooth as a composition of smooth functions, and second, that it is non-singular, as we showed for J_Q above. As such, the matrix inverse is also smooth, making $\pi_z \mapsto M^z$ a smooth function.

As g is a composition of smooth functions ($z \mapsto R \mapsto Q^\star \mapsto \pi_z \mapsto M^z$), we can conclude that it is also smooth. □

This property allows smooth interpolation of behaviors; however we note that the solutions of linearly interpolated task vectors do not necessarily lie on a line in the space of successor measures. In contrast, the remark above does not hold for FB in general: repurposing the counterexample in Section A.2, and in particular Figure 6, we can see that changes in z results in a sharp change in π_z and M^z when $z_1 = z_2$.

A.5 Useful Results

This section refreshes some known results leveraged in the proofs.

Theorem A.3 (Simulation Lemma for Discounted Successor Measures (adapted from Kearns and Singh [2002]))**.** *Let π, π' be two stationary Markov policies in an MDP $\mathcal{M}$ and $\mathrm{M}^\pi(X) = (1 - \gamma) \sum_{t \geq 0} \gamma^t \Pr((s_t, a_t) \in X \mid \mu_0, P, \pi)$ be the marginalized state-action successor measure for π. Then,*

$$\|\mathrm{M}^{\pi} - \mathrm{M}^{\pi'}\|_1 \leq \frac{1}{1-\gamma} \max_{s \in \mathcal{S}} \|\pi(\cdot \mid s) - \pi'(\cdot \mid s)\|_1.$$

Proof. Following the Bellman flow constraints, we start by fixing an initial state $s \in \mathcal{S}$ and considering the state-action successor measure $\mathrm{M}^{\pi}_{s} := (1-\gamma) \sum_{t \geq 0} \gamma^t \Pr((s_t, a_t) \in X \mid s_0 = s, P, \pi)$ starting from s, as well as the state-only successor measure $\mathrm{M}^{\pi}_{\mathcal{S},s} := (1-\gamma) \sum_{t \geq 0} \gamma^t \Pr(s_t \in X \mid s_0 = s, P, \pi)$:

$$\|\mathrm{M}^{\pi}_{s} - \mathrm{M}^{\pi'}_{s}\|_1 = \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left| \mathrm{M}^{\pi}_{\mathcal{S},s}(s') \pi(a|s') - \mathrm{M}^{\pi'}_{\mathcal{S},s}(s') \pi'(a|s') \right| \tag{37}$$

$$= \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left| \mathrm{M}^{\pi}_{\mathcal{S},s}(s') \pi(a|s') - \mathrm{M}^{\pi}_{\mathcal{S},s}(s') \pi'(a|s') + \mathrm{M}^{\pi}_{\mathcal{S},s}(s') \pi'(a|s') - \mathrm{M}^{\pi'}_{\mathcal{S},s}(s') \pi'(a|s') \right| \tag{38}$$

$$\leq \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \mathrm{M}^{\pi}_{\mathcal{S},s}(s') \left| \pi(a|s') - \pi'(a|s') \right| + \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left| \mathrm{M}^{\pi}_{\mathcal{S},s}(s') - \mathrm{M}^{\pi'}_{\mathcal{S},s}(s') \right| \pi'(a|s') \tag{39}$$

$$= \sum_{s' \in \mathcal{S}} \mathrm{M}^{\pi}_{\mathcal{S},s}(s') \|\pi(\cdot|s') - \pi'(\cdot|s')\|_1 + \sum_{s' \in \mathcal{S}} \left| \mathrm{M}^{\pi}_{\mathcal{S},s}(s') - \mathrm{M}^{\pi'}_{\mathcal{S},s}(s') \right| \sum_{a \in \mathcal{A}} \pi'(a|s') \tag{40}$$

$$\leq \max_{s' \in \mathcal{S}} \|\pi(\cdot|s') - \pi'(\cdot|s')\|_1 + \|\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s}\|_1. \tag{41}$$

We can now focus on bounding the second term:

$$\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s} = \left((1-\gamma) e_s^{\top} + \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} P^{\pi}\right) - \left((1-\gamma) e_s^{\top} + \gamma \mathrm{M}^{\pi'}_{\mathcal{S},s} P^{\pi'}\right) \tag{42}$$

$$= \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} P^{\pi} - \gamma \mathrm{M}^{\pi'}_{\mathcal{S},s} P^{\pi'} \tag{43}$$

$$= \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} P^{\pi} - \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} P^{\pi'} + \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} P^{\pi'} - \gamma \mathrm{M}^{\pi'}_{\mathcal{S},s} P^{\pi'} \tag{44}$$

$$= \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} (P^{\pi} - P^{\pi'}) + \gamma (\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s}) P^{\pi'} \tag{45}$$

We then rearrange the terms as

$$\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s} - \gamma (\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s}) P^{\pi'} = \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} (P^{\pi} - P^{\pi'}) \tag{46}$$

$$(\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s})(I - \gamma P^{\pi'}) = \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} (P^{\pi} - P^{\pi'}) \tag{47}$$

$$\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s} = \gamma \mathrm{M}^{\pi}_{\mathcal{S},s} (P^{\pi} - P^{\pi'})(I - \gamma P^{\pi'})^{-1}. \tag{48}$$

We can then take the L_1 norm and apply the standard norm inequality:

$$\|\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s}\|_1 \leq \gamma \|\mathrm{M}^{\pi}_{\mathcal{S},s}\|_1 \|P^{\pi} - P^{\pi'}\|_{\infty} \|(I - \gamma P^{\pi'})^{-1}\|_{\infty}. \tag{49}$$

We then note that $\|\mathrm{M}^{\pi}_{\mathcal{S},s}\|_1 = 1$, as it is a probability distribution, that $\|(I - \gamma P^{\pi'})^{-1}\|_{\infty} = \sum_{t \geq 0} \gamma^t = \frac{1}{1-\gamma}$ and that $\|P^{\pi} - P^{\pi'}\|_{\infty} \leq \sup_{x \in \mathcal{S}} \|\pi(\cdot \mid x) - \pi'(\cdot \mid x)\|_1$ as the underlying dynamics are the same. Therefore,

$$\|\mathrm{M}^{\pi}_{\mathcal{S},s} - \mathrm{M}^{\pi'}_{\mathcal{S},s}\|_1 \leq \frac{\gamma}{1-\gamma} \max_{s' \in \mathcal{S}} \|\pi(\cdot \mid s') - \pi'(\cdot \mid s')\|_1, \tag{50}$$

and combining this with the previous bound for state-action measures we have

$$\|\mathrm{M}_s^{\pi} - \mathrm{M}_s^{\pi'}\|_1 \leq \max_{s' \in \mathcal{S}} \|\pi(\cdot \mid s') - \pi'(\cdot \mid s')\|_1 + \frac{\gamma}{1-\gamma} \max_{s' \in \mathcal{S}} \|\pi(\cdot \mid s') - \pi'(\cdot \mid s')\|_1 \tag{51}$$

$$= \frac{1}{1-\gamma} \max_{s' \in \mathcal{S}} \|\pi(\cdot \mid s') - \pi'(\cdot \mid s')\|_1. \tag{52}$$

We finally note that M^{π} is an expectation of M_s^{π} over $\mathcal{S}$, and thus

$$\|\mathrm{M}^{\pi} - \mathrm{M}^{\pi'}\|_1 = \|\mathbb{E}_{s \sim \mu_0}[\mathrm{M}_s^{\pi} - \mathrm{M}_s^{\pi'}]\|_1 \tag{53}$$

$$\leq \mathbb{E}_{s \sim \mu_0} \|\mathrm{M}_s^{\pi} - \mathrm{M}_s^{\pi'}\|_1 \tag{54}$$

$$\leq \frac{1}{1-\gamma} \max_{s' \in \mathcal{S}} \|\pi(\cdot \mid s') - \pi'(\cdot \mid s')\|_1. \tag{55}$$

□

Theorem A.4 (Heine-Cantor Theorem)**.** *If $K \subset \mathbb{R}^n$ is compact and $f : K \to \mathbb{R}^m$ is continuous, then f is uniformly continuous on K, i.e.*

$$\forall \varepsilon > 0 \; \exists \delta > 0 \; \forall x, y \in K, \quad \|x - y\| < \delta \implies \|f(x) - f(y)\| < \varepsilon.$$

B Further Related Work

Zero-shot Reinforcement Learning Goal-conditioned policies [Andrychowicz et al., 2017, Eysenbach et al., 2022] and universal value functions approximators [Schaul et al., 2015] represent a further approach to extract flexible multi-task policies with minimal supervision. Such methods normally specialize to specific, parameterized reward classes (e.g., Dirac [Tian et al., 2021]), or combinations thereof [Frans et al., 2024], and can train conditional policies through self-supervision [Ghosh et al., 2023], with impressive empirical results [Akkaya et al., 2019]. Successor-feature-based methods [Dayan, 1993, Barreto et al., 2017] try to solve a potentially broader class of function, containing all scalars that are linear in the features. Such algorithms, and in particular those relying on forward-backward representations, were gradually expanded for imitation learning [Pirotta et al., 2024], self-supervised exploration [Urpí et al., 2025]. Recent works have further explored alternative parameterizations [Cetin et al., 2024, Bagatella et al., 2026], more capable generative models for occupancies [Farebrother et al., 2025], fast online exploration or adaptation [Sikchi et al., 2025b, Urpí et al., 2025, Rupf et al., 2026].

Non-linear Reinforcement Learning RL with General Utilities [Zhang et al., 2020] may not be directly solved through standard dynamic programming or actor-critic approaches, as the key assumption behind TD learning is violated. On the other hand, policy gradient theorems may still be derived [Zhang et al., 2020, 2021, Kumar et al., 2022, Barakat et al., 2023]. Specific algorithms for subclasses of convex RL problems (such as maximum entropy RL [Haarnoja et al., 2018] or imitation learning [Kostrikov et al., 2020]) can be derived, for instance by building upon duality [Sikchi et al., 2023], but do not generalize to arbitrary objectives. Recent works have also focused on non-linear objectives displaying set properties [De Santi et al., 2024], while a parallel line of work has departed from optimizing for expected occupancy to consider finite trial objectives [Mutti et al., 2023, Jain et al., 2023]. While theoretical analysis of algorithms for non-linear RL has received significant attention [Huang and Jiang, 2024, Barakat et al., 2025], their scalability to high-dimensional problem has not been extensively tested. Similarly, most algorithms deal with the standard online setting, and solve for one objective at a time. We remark that general utilities remain relevant in the offline setting we operate in. For instance, if we want to collect additional data for adaptation after offline pre-training, a great candidate for data collection would be the policy achieving the best coverage over the state space, i.e. maximizing a pure exploration objective.

Practical Applications of General Utilities Pure exploration objectives have been used for efficient data collection and fast downstream adaptation on a D'Claw robotic platform [Lee et al., 2020] and a door opening task [Pong et al., 2020]; non-linear exploration objectives have also been used for system identification of a manipulation robot [Memmel et al., 2024] and coverage and search missions with drones [Mathew et al., 2013]. Humanoids and quadrupeds imitating mocap data do

not have access to action labels: in this case, the problem is often formulated as a general utility, namely distribution matching [Escontrela et al., 2022, Li et al., 2026], and solved through generative adversarial schemes specialized to imitation learning. Constrained and behavior-regularized RL objectives are also instances of general utilities which may not be generally reduced to linear rewards, and have been applied for quadruped locomotion [Lee et al., 2024] or robotic manipulation, both when training a policy [Yu et al., 2026] or a reward function [Ma et al., 2022].

C Explicit Measure Models

The inference procedure of Soft FB requires accurate measure estimates. As we mention in Section 4.4, while a rough estimate can be extracted from an implicit measure model (i.e., relying on forward and backward representations directly for importance sampling), we find that training an explicit measure model results in much more reliable estimates. Following recent work on geometric horizon models [Farebrother et al., 2025, Zheng et al., 2026], we train a generative model of z-conditional successor measures M^z through a temporal-difference, conditional flow-matching objective; we briefly report the core idea as follows. We will omit the conditioning on z for notational simplicity.

Let us consider a time-dependent probability path $m_t : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ for $t \in [0, 1]$ describing a process transporting samples from an initial distribution $m_0 = \mathcal{N}(0, I)$ to $m_1 = M^z$. This process is described by a vector field $v_t : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathcal{S}$, which may be integrated to produce the flow $\psi_t : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ governing the probability path:

$$\frac{\mathrm{d}}{\mathrm{d}t}\psi_t(x \mid s, a) = v_t(\psi_t(x \mid s, a) \mid s, a),$$

$$\psi_0(x \mid s, a) = x \iff \psi_t(x \mid s, a) = x + \int_0^t v_\tau(\psi_\tau(x \mid s, a) \mid s, a)\,\mathrm{d}\tau.$$

If $X_t := \psi_t(X_0 \mid S, A) \sim m_t(\cdot \mid S, A)$ for $X_0 \sim m_0$, v_t is said to generate m_t; estimation of v_t is facilitated by introducing a conditioning (e.g., $Z = X_1$) and choosing a Gaussian conditional probability path $p_{t,Z} := \mathcal{N}(\cdot|tX_1, (1-t)^2 I)$, which grants a closed form for a *conditional* vector field $u_{t,X_1}(x) := (X_1 - x)/(1 - t)$. Furthermore, off-policy learning is possible by bootstrapping (i.e., using the estimated vector field to sample from measures conditional on the next state), and variance reduction can be carried out by directly matching the the estimated vector field. This results in the TD^2-CFM objective for training a parameterized estimator of the vector field $\tilde{v}_{t,\theta} : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathcal{S}$, which we report:

$$\vec{\mathcal{L}}(\theta) = \mathbb{E}_{\rho,t,Z,\vec{X}_t}\left[\left\|\tilde{v}_t(\vec{X}_t \mid S, A; \theta) - \vec{u}_{t|Z}(\vec{X}_t \mid Z)\right\|^2\right],$$

$$\text{where } Z = X_1 \sim P(\cdot \mid S, A),\ \vec{X}_t \sim p_{t|Z}(\cdot \mid Z),$$

$$\widetilde{\mathcal{L}}(\theta) = \mathbb{E}_{\rho,t,\widetilde{X}_t}\left[\left\|\tilde{v}_t(\widetilde{X}_t \mid S, A; \theta) - \bar{\tilde{v}}_t(\widetilde{X}_t \mid S', \pi_z(S'))\right\|^2\right],$$

$$\text{where } X_0 \sim p_0,\ S' \sim P(\cdot \mid S, A),\ \widetilde{X}_t = \widetilde{\psi}_t(X_0 \mid S', \pi_z(S')),$$

$$\mathcal{L}_{\text{TD}^2\text{-CFM}}(\theta) = (1 - \gamma)\vec{\mathcal{L}}(\theta) + \gamma\widetilde{\mathcal{L}}(\theta),$$

where ρ represents the empirical distribution of states in the same dataset used for training Soft FB. We refer to Farebrother et al. [2025] for a detailed discussion, and for implementation details, which we closely follow.

D Algorithmic Description

Algorithm 1 describes the practical version of SoftFB introduced in Section 4.3 in its entirety.

E Additional Experimental Results

E.1 Correlation

The policy search procedure outlined in Section 4.4 relies on offline performance estimates, which are only effective if they correlate with ground truth performance in the environment. We find that

Algorithm 1: Practical Soft FB for Zero-shot RL with GUs

Input: Number of gradient updates N, test-time budget K, dataset $\mathcal{D}$, batch size B
Randomly initialized policy π_ψ, representations F_θ, B_ϕ and entropy critic $Q_{\mathcal{H},\eta}$
(Optional) Randomly initialized explicit measure model $\tilde{v}$

`// Pre-training phase`
Initialize target networks for F_θ, $Q_{\mathcal{H},\eta}$ and, optionally, $\tilde{v}$;
for $i = 1$ **to** N **do**
  Sample training batch $(s, a, s')_{i=1}^B$ from $\mathcal{D}$;
  Sample embeddings $z_{i=1}^B \sim \mathcal{U}(\mathcal{Z})$;
  Sample bootstrap actions $a'_i \sim \pi_\psi(\cdot|s'_i, z_i)$;
  Update θ, ϕ by minimizing Eq. 7 (optionally with regularization, cf. Appendix F);
  Update η by minimizing Eq. 8;
  Update ψ by minimizing Eq. 9;
  Update $\tilde{v}$ as described in Appendix C (optionally);
  Update target networks through Polyak averaging;

`// Test-time (e.g., optimizing through random shooting)`
for *each GU objective* f **do**
  Receive initial state distribution μ_0;
  Sample K candidate embeddings $z_{i=1}^K$;
  Estimate successor measures $\{\mathrm{M}_{z_i}\}_{i=1}^K$ through Eq. 12 or $\tilde{v}$;
  Evaluate successor measures $\hat{f}_i = f(\mathrm{M}_{z_i})$;
  Find best sample $i^\star = \mathrm{argmax}_i \hat{f}_i$ and yield downstream policy $\pi_\psi(z_{i^\star})$;

the degree of correlation depends on the objective and on the technique used for sampling from the measure: explicit models (denoted with, e.g., SFB$_{flow}$) generally result in more reliable policy evaluation. For completeness, we compute Spearman's rank correlation coefficients across the results from Table 1 in Table 3.

Table 3: Spearman's rank correlation coefficients between offline policy estimates and ground-truth performance in the didactic environment.

	FB	FB$_{flow}$	SFB	SFB$_{flow}$
Linear RL	$0.98_{\pm.01}$	$\mathbf{1.00}_{\pm.01}$	$0.83_{\pm.02}$	$\mathbf{1.00}_{\pm.01}$
Goal-reaching RL	$0.90_{\pm.01}$	$\mathbf{1.00}_{\pm.01}$	$0.86_{\pm.03}$	$\mathbf{1.00}_{\pm.01}$
Deterministic IL	$0.91_{\pm.02}$	$\mathbf{0.97}_{\pm.02}$	$0.41_{\pm.03}$	$\mathbf{1.00}_{\pm.01}$
Stochastic IL	$0.90_{\pm.04}$	$0.94_{\pm.03}$	$0.34_{\pm.05}$	$\mathbf{1.00}_{\pm.01}$
Pure Exploration	$0.09_{\pm.14}$	$0.53_{\pm.15}$	$0.01_{\pm.04}$	$\mathbf{1.00}_{\pm.01}$
Constrained RL	$0.13_{\pm.03}$	$0.76_{\pm.15}$	$0.40_{\pm.03}$	$\mathbf{0.96}_{\pm.03}$
Robust RL	$0.19_{\pm.03}$	$0.96_{\pm.02}$	$0.47_{\pm.02}$	$\mathbf{1.00}_{\pm.01}$
Average	0.58	0.88	0.48	**0.99**

E.2 Linear RL on DMC

We report numerical results from Figure 5 (i.e., the evaluation of linear objectives in DMC) in Table 4.

E.3 Visualization of Action Samples

While Soft FB retrieves stochastic policies, the standard forward-backward objectives do not explicitly prevent policies to collapse towards determinism. While set of policies retrieved by FB may, in principle, also include stochastic policies, we find that this is, in practice, not the case. As a didactic example, we repeat the visualization in Figure 3, this time sampling actions from policies trained with FB. These samples are displayed in Figure 7, which confirms that retrieved policies are deterministic.

Table 4: Episodic returns and step-wise action entropy across different environments as a function of $\|z\|$, averaged over tasks and reported from Figure 5.

$\|z\|$	walker		cheetah		quadruped		maze	
	Return	Entropy	Return	Entropy	Return	Entropy	Return	Entropy
0.00	123 ± 24	3.5 ± 0.5	77 ± 5	14.5 ± 0.2	171 ± 10	29.8 ± 0.2	0 ± 0	4.6 ± 0.1
0.25	622 ± 16	0.3 ± 0.1	313 ± 24	13.1 ± 0.1	681 ± 2	24.8 ± 0.3	224 ± 11	3.8 ± 0.2
0.50	688 ± 13	-2.3 ± 0.2	399 ± 23	11.2 ± 0.2	696 ± 7	20.3 ± 0.4	350 ± 62	2.6 ± 0.2
0.75	698 ± 12	-8.2 ± 0.7	440 ± 12	7.0 ± 0.5	687 ± 4	16.1 ± 0.3	472 ± 52	-0.0 ± 0.2
1.00	702 ± 8	-12.4 ± 0.7	446 ± 19	-3.2 ± 0.9	668 ± 6	7.1 ± 1.0	522 ± 18	-2.9 ± 0.2

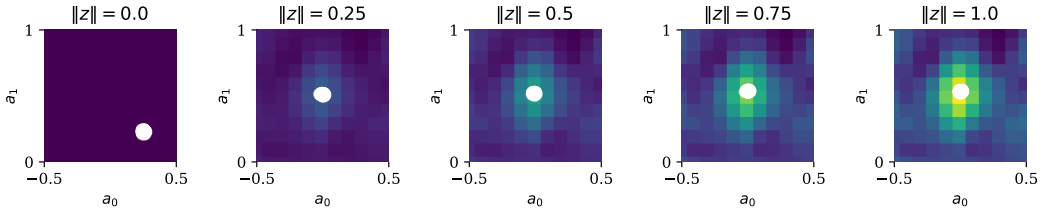


Figure 7: Qualitative evaluation of FB in a didactic environment. White dots are samples from policies π_z over a 2D actions space, and the color map represents learned unregularized Q-values Q_R^z for each action $(F_\theta(s_0, a, z)^\top z)$. From left to right, we infer task embeddings z solving a goal-reaching task, and scale them linearly. In this case, all retrieved policies are near-deterministic; interestingly, FB often outputs optimal actions for embeddings of norm $\|z\| < 1$, despite not being explicitly trained on them.

E.4 Extended results on DMC

This Section extends the experimental results summarized in Table 2 in several ways.

First, we include two additional objectives (J_{robust} and $J_{\text{constrained}}$, thus evaluating the same family of non-linear objectives considered in Table 1. When interpreting results, we remark that the domains considered are not generally designed for non-linear objectives; as dynamics are unknown, it is not clear whether some of these objectives may still admit deterministic optimal policies: for instance, high state entropy may be achievable in `maze` by a deterministic policy which covers the room in a non-overlapping zig-zag pattern. We find that Soft FB is consistently the strongest method for the two objectives that do not admit an optimal deterministic policy, namely maximum entropy over states and actions ($\mathcal{H}(\mathrm{M}^\pi)$) and imitation of a stochastic policy ($-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{stoch}})$). Perhaps surprisingly, Soft FB also performs well in imitating a deterministic policy: while KL divergence to the expert is minimized by a deterministic policy, we hypothesize that finding a stochastic policy with a low divergence is generally easier. For other objectives (e.g., state-only entropy maximization, constrained or robust objectives), we find that the best policy captured by FB may perform as well, confirming that deterministic policies may be sufficient for nonlinear objectives in specific MDPs.

Second, Table 5 includes the performance of FB and Soft FB with implicit measure models (2nd and 4th columns, respectively). We find that implicit measure models also perform relatively well, and not significantly worse than explicit measure models in most tasks when combined with Soft FB. We believe that with further compute allocation, explicit measure models can be further improved, as discussed in Appendix F.

Third, we report the performance of a random policy sampled from those trained by FB for each objective and domain (first column of Table 5). While this simple baseline is surpassed by Soft FB$_{flow}$ in each instance, the performance difference is not significant in a few cases. This suggests that, for specific objectives, Soft FB may in practice not recover a rich enough class of policies, such that the best of its policies may not be much better than the average FB policy (we remark that the guarantees in Section 4.2 only hold with exact minimization of the objectives, and for sufficiently high-dimensional representations). Furthermore, the accuracy of the inference procedure remains limited by the quality of measure models, which are also learned, and therefore imperfect. This is in particular the case for higher-dimensional domains such as `quadruped`.

Table 5: Zero-shot performance on general utilities in DMC. We extend Table 2 by including three additional methods and two objectives. Scores are averages over 5 seeds, with 95% confidence intervals and bold numbers signaling overlaps.

		FB$_{rand}$	FB	FB$_{flow}$	Soft FB	Soft FB$_{flow}$
walker	$\mathcal{H}(\mathrm{M}^\pi_\mathcal{S})$	**12.99** $\pm.94$	**12.82** $\pm.04$	12.56 $\pm.91$	**13.91** $\pm.20$	**13.97** $\pm.25$
	$\mathcal{H}(\mathrm{M}^\pi)$	9.90 $\pm.11$	9.77 $\pm.09$	9.86 $\pm.10$	**18.00** $\pm.02$	**18.01** $\pm.03$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; M^{\pi_{stoch}}_\mathcal{S})$	**-6.08** $\pm.89$	-6.91 $\pm.03$	**-5.35** $\pm.24$	**-5.11** $\pm.27$	**-5.25** $\pm.14$
	$-KL(\mathrm{M}^\pi; M^{\pi_{stoch}})$	-9.64 $\pm.20$	-9.56 $\pm.23$	-9.40 $\pm.17$	**-1.39** $\pm.05$	**-1.48** $\pm.11$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; M^{\pi_{det}}_\mathcal{S})$	**-6.18** $\pm.05$	-6.87 $\pm.97$	-5.69 $\pm.27$	**-5.02** $\pm.16$	**-5.69** $\pm.02$
	$-KL(\mathrm{M}^\pi; M^{\pi_{det}})$	-9.66 $\pm.15$	-9.62 $\pm.20$	-9.43 $\pm.18$	**-1.48** $\pm.08$	**-1.53** $\pm.09$
	$J_{\text{robust}}(\mathrm{M}^\pi)$	0.39 $\pm.04$	**0.51** $\pm.08$	**0.51** $\pm.03$	**0.56** $\pm.12$	**0.44** $\pm.07$
	$J_{\text{constrained}}(\mathrm{M}^\pi)$	**0.13** $\pm.22$	**0.13** $\pm.30$	**0.23** $\pm.32$	**-0.10** $\pm.25$	**0.17** $\pm.19$
cheetah	$\mathcal{H}(\mathrm{M}^\pi_\mathcal{S})$	11.57 $\pm.43$	12.66 $\pm.15$	11.74 $\pm.43$	**13.69** $\pm.15$	**13.63** $\pm.10$
	$\mathcal{H}(\mathrm{M}^\pi)$	9.96 $\pm.21$	10.59 $\pm.88$	11.68 $\pm.89$	**17.85** $\pm.04$	**17.83** $\pm.07$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; \mathrm{M}^{\pi_{stoch}}_\mathcal{S})$	-6.54 $\pm.68$	-5.58 $\pm.20$	-5.56 $\pm.31$	**-4.36** $\pm.15$	**-4.56** $\pm.31$
	$-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{stoch}})$	-8.64 $\pm.45$	-7.90 $\pm.90$	-7.25 $\pm.76$	**-1.11** $\pm.05$	**-1.02** $\pm.08$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; \mathrm{M}^{\pi_{det}}_\mathcal{S})$	-6.52 $\pm.78$	-5.58 $\pm.28$	-5.45 $\pm.65$	**-4.35** $\pm.27$	**-4.38** $\pm.18$
	$-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{det}})$	-8.92 $\pm.40$	-8.14 $\pm.85$	-7.55 $\pm.79$	**-1.19** $\pm.05$	**-1.23** $\pm.08$
	$J_{\text{robust}}(\mathrm{M}^\pi)$	**0.19** $\pm.07$	**0.22** $\pm.04$	**0.24** $\pm.06$	**0.24** $\pm.04$	**0.21** $\pm.01$
	$J_{\text{constrained}}(\mathrm{M}^\pi)$	**0.11** $\pm.08$	**0.18** $\pm.09$	**0.14** $\pm.18$	**0.19** $\pm.10$	**0.23** $\pm.07$
quadruped	$\mathcal{H}(\mathrm{M}^\pi_\mathcal{S})$	**13.84** $\pm.49$	**13.67** $\pm.74$	13.37 $\pm.43$	**14.48** $\pm.23$	**14.43** $\pm.24$
	$\mathcal{H}(\mathrm{M}^\pi)$	10.19 $\pm.08$	10.18 $\pm.14$	10.11 $\pm.08$	**19.09** $\pm.03$	**19.10** $\pm.01$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; \mathrm{M}^{\pi_{stoch}}_\mathcal{S})$	-5.54 $\pm.47$	-5.99 $\pm.60$	-5.44 $\pm.28$	**-4.95** $\pm.35$	**-4.64** $\pm.20$
	$-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{stoch}})$	-9.93 $\pm.08$	-9.87 $\pm.18$	-9.92 $\pm.16$	**-0.75** $\pm.02$	**-0.76** $\pm.04$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; \mathrm{M}^{\pi_{det}}_\mathcal{S})$	-5.64 $\pm.53$	-6.07 $\pm.74$	-5.61 $\pm.30$	-4.96 $\pm.14$	**-4.52** $\pm.16$
	$-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{det}})$	-9.95 $\pm.07$	-9.99 $\pm.22$	-9.94 $\pm.20$	**-0.79** $\pm.02$	**-0.79** $\pm.02$
	$J_{\text{robust}}(\mathrm{M}^\pi)$	0.07 $\pm.01$	**0.14** $\pm.02$	0.09 $\pm.03$	**0.17** $\pm.04$	0.08 $\pm.01$
	$J_{\text{constrained}}(\mathrm{M}^\pi)$	**0.03** $\pm.03$	**0.02** $\pm.04$	**0.04** $\pm.06$	**0.06** $\pm.04$	**0.07** $\pm.06$
maze	$\mathcal{H}(\mathrm{M}^\pi_\mathcal{S})$	**10.31** $\pm.34$	**10.68** $\pm.21$	**10.83** $\pm.56$	**10.62** $\pm.66$	**11.22** $\pm.70$
	$\mathcal{H}(\mathrm{M}^\pi)$	10.00 $\pm.48$	10.96 $\pm.70$	10.52 $\pm.66$	**15.67** $\pm.08$	**15.54** $\pm.16$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; \mathrm{M}^{\pi_{stoch}}_\mathcal{S})$	-6.32 $\pm.62$	**-6.15** $\pm.67$	-6.49 $\pm.36$	**-5.23** $\pm.31$	**-5.25** $\pm.90$
	$-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{stoch}})$	-7.22 $\pm.34$	-6.61 $\pm.14$	-6.45 $\pm.61$	**-1.08** $\pm.21$	**-1.39** $\pm.31$
	$-KL(\mathrm{M}^\pi_\mathcal{S}; \mathrm{M}^{\pi_{det}}_\mathcal{S})$	-6.62 $\pm.54$	-6.42 $\pm.58$	-6.57 $\pm.61$	**-5.76** $\pm.32$	**-5.07** $\pm.69$
	$-KL(\mathrm{M}^\pi; \mathrm{M}^{\pi_{det}})$	-8.67 $\pm.54$	-7.27 $\pm.96$	-7.27 $\pm.58$	**-2.37** $\pm.27$	**-2.51** $\pm.35$
	$J_{\text{robust}}(\mathrm{M}^\pi)$	0.17 $\pm.10$	**0.44** $\pm.06$	**0.34** $\pm.19$	0.33 $\pm.13$	**0.52** $\pm.05$
	$J_{\text{constrained}}(\mathrm{M}^\pi)$	-0.01 $\pm.11$	**0.46** $\pm.22$	**0.53** $\pm.24$	**0.31** $\pm.22$	**0.57** $\pm.14$

Fourth and last, we briefly discuss our results in pinpointing and evaluating algorithmic *toplines*. A direct source of comparison would be principled algorithms for GUs, which often come with strong theoretical guarantees (e.g. convergence) for arbitrary objectives [Zhang et al., 2020, Kumar et al., 2022, Barakat et al., 2023, 2025]. Being single-task and online algorithms, we would expect them to outperform Soft FB when allowed privileged access to the environment. However, these algorithms have not been shown to scale beyond tabular MDPs or linear function approximation [Barakat et al., 2023]. When evaluated in high-dimensional environments (i.e., DMC), we did not find these policy-gradient-based methods to perform well.

E.5 Entropy-averse evaluation

While some of the objectives in Section 5.1 are included specifically to stress whether SoftFB captures stochastic policies, the the goal-reaching evaluation in Table 1 is arguably opposed to the bias of SoftFB, by making any degree of stochasticity unnecessary and potentially counterproductive. This evaluation requires the agent to select actions that are within a given radius of the goal; a stochastic policy would fail to consistently select those actions, and thus underperform. The fact that SoftFB scores well confirms that it also recovers near-deterministic policies capable of precise control. In this Section, we further stress-test SoftFB on heavily entropy-opposed objectives. In particular, we evaluate variants of the goal-reaching tasks in which we shrink the radius of the goal area by controlling ρ in the reward $R(x, y) = \mathbf{1}_{\sqrt{(x-x_p)^2+(y-y_p)^2}<\rho}$. By default, $(x_g, y_g) = (0.0, 0.5)$ and

Table 6: Goal-reaching evaluation from Table 1, repeated for different radii ρ. Scores are averaged over 3 seeds with 95% confidence intervals.

	Search		Closed-form	
	FB_{flow}	$\text{SoftFB}_{\text{flow}}$	FB_{flow}	$\text{SoftFB}_{\text{flow}}$
GCRL ($\rho = 0.1$)	1.0 ± 0.01	1.0 ± 0.01	1.0 ± 0.01	1.0 ± 0.01
GCRL ($\rho = 0.05$)	1.0 ± 0.01	0.96 ± 0.04	1.0 ± 0.01	1.0 ± 0.01
GCRL ($\rho = 0.02$)	0.62 ± 0.42	0.28 ± 0.09	0.69 ± 0.31	0.59 ± 0.46
GCRL ($\rho = 0.01$)	0.17 ± 0.15	0.08 ± 0.03	0.10 ± 0.08	0.24 ± 0.20
GCRL ($\rho = 0.005$)	0.03 ± 0.02	0.02 ± 0.01	0.02 ± 0.02	0.06 ± 0.06
GCRL ($\rho = 0.002$)	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.01 ± 0.01

$\rho = 0.2$. As we decrease the radius, we expect SoftFB's performance to decrease faster, as any degree of stochasticity is eventually penalized. We evaluate FB and SoftFB (in their more accurate variant, using explicit measure models) by finding the best policy through zero-order search and through the closed form inference procedure in Table 6.

As expected, we find that the performance of both methods decays as the size of the goal shrinks. Under search, FB leads for tight radii (≈ 0.02). SoftFB remains generally competitive *despite the adversarial evaluation*, and closes the gap under closed-form inference. This confirms its ability to retrieve near-deterministic policies, and that the initial disadvantage only arises due to having to search through a richer family of policies, which is not necessary when knowing that the objective is entropy-averse.

F Implementation Details

Our practical implementation of Soft FB builds upon the vanilla FB algorithm introduced in Touati and Ollivier [2021] and open-sourced in Tirinzoni et al. [2024]. Architecturally, the sole modification lies in changing the policy's parameterization: from a Gaussian with a fixed standard deviation of $\sigma = 0.3$ to a squashed Gaussian with learnable standard deviation. This modification is applied to all of the algorithms we evaluate. All networks are MLPs with embedding layers; width and depth are reported in Table 7.

Unlike FB, Soft FB trains an action-state value function to estimate discounted sums of entropy terms (see Equation 8), which shares the same optimization hyperparameters and the same architecture as the forward map F_θ, but outputs scalar values instead of embeddings. As the forward map, the critic also relies on twin networks, and on a target network updated at the same pace. While the discounted sum of entropies may also be estimated implicitly through forward and backward representations, we found that training a separate critic resulted in more reliable estimates.

When explicit measure models are used for inference, a flow-matching generative model is trained in parallel with Soft FB. For this purpose, we closely follow the implementation details described in [Farebrother et al., 2025]. Due to limited computational budget, we train the model for fewer gradient steps (3M instead of 8M). While this was sufficient for policy selection, allocating further resource to measure estimation may bring additional performance gains.

Our implementation of Soft FB inherits all of FB's hyperparameters, without any additional tuning; we present the main hyperparameters in Table 7, and we refer to our codebase for specific details. Soft FB does not introduce any additional hyperparameter: the standard entropy regularization coefficient common in entropy-regularized RL [Haarnoja et al., 2017] is not needed, as Soft FB trains on all levels of entropy regularization, as discussed in Section 4.3. Additionally scaling the entropy term by a fixed coefficient is possible, and may lead to better performance for tasks in which high- or low-entropy policies are favored.

Our implementation is open-sourced at the project's repository; further details on implementation and evaluation are organized by environment in the following sections. Each experimental run requires between 3 and 72 hours on a single GPU (e.g. RTX 4090 or equivalent), depending on the complexity of the environment. The total compute cost of this work hovers in the order of $\sim 10^5$ GPU-hours.

Table 7: Hyperparameter Configuration

Parameter	Value
F_θ - Learning rate	1×10^{-4}
F_θ - Width	1024
F_θ - Depth	4
B_ϕ - Learning rate	1×10^{-4}
B_ϕ - Width	256
B_ϕ - Depth	3
π_ψ - Learning rate	1×10^{-4}
π_ψ - Width	1024
π_ψ - Depth	4
$Q_{\mathcal{H},\eta}$ - Learning rate	1×10^{-4}
$Q_{\mathcal{H},\eta}$ - Width	1024
$Q_{\mathcal{H},\eta}$ - Depth	4
Optimizer	Adam [Kingma and Ba, 2015]
Dimensionality of z	50
Polyak coefficient τ	0.01
Orthonormality regularization coefficient	1.0
Goal sampling ratio	0.5
Q-value aggregation	`mean`
Batch size	1024

F.1 Low-dimensional Evaluation

The didactic MDP described in Section 5.1 has a discount factor of 0.5; due to its simplicity, agents are trained for $5 \cdot 10^4$ gradient steps. The task inference procedure is carried out as described in Section 4.4 with 1024 uniformly sampled reward embeddings as candidates. In the case of standard FB, they are always sampled from the surface of an hypersphere, as the agent was not trained for embeddings lying within its volume. For each z, 1024 samples are drawn from the z-conditional successor measure $\hat{M}^{\pi_z}$, estimated through either implicit or explicit measure models. These samples are then evaluated according the objective, which are presented in full in Table 8; the highest-scoring sample is provided as a conditioning to the policy, which is then evaluated in the environment for 1024 episodes to produce the scores reported in Table 1.

Objectives were designed to be easily interpretable: the linear task encourages states on a unit circle around the origin, the goal-conditioned task involves landing in proximity of a certain state, while the deterministic and stochastic imitation learning tasks feature an expert which always visits the same state, or visits states uniformly on a line, respectively. The pure exploration objective simply maximizes state coverage, while the robust objective considers the minimum between a goal conditioned objective and its complement, and the constrained objective maximizes a goal-conditioned objective within a given range. For objectives requiring estimation of entropies or Kullback-Leibler divergence, we employ a standard nearest-neighbor-based estimator [Wang et al., 2009], with $k = 3$. For each objective, we normalized scores between minima and maxima (theoretical if available, or estimated from the policies with lowest performance), as described in the last two columns of Table 8. Across this work, confidence intervals are computed assuming scores are normally distributed.

Table 8: Objectives considered in the quantitative evaluation in low-dimensional settings (see Figure 4). $s = (x, y)$ denotes the two coordinates of each state $s \in \mathcal{S}$.

Task	Objective	`min`	`max`
Linear RL	$\max_\pi \langle \mathrm{M}^\pi_\mathcal{S}, R \rangle$ with $R(x,y) = -((x^2+y^2)-1)^2$	≈ 0	1
Goal-conditioned RL	$\max_\pi \langle \mathrm{M}^\pi_\mathcal{S}, R_{(0.0,0.5)}(x,y) \rangle$ with $R_{(x_g,y_g)}(x,y) = \mathbf{1}_{\left((x-x_g)^2+(y-y_g)^2\right)^{\frac{1}{2}}<0.2}$	0	1
Deterministic IL	$\max_\pi -D_{\mathrm{KL}}(\mathrm{M}^\pi_\mathcal{S} \| \mathrm{M}^\star_\mathcal{S})$ with $\mathrm{M}^\star_\mathcal{S} = \mathbf{1}_{s=(0.0,0.5)}$	≈ -15	0
Stochastic IL	$\max_\pi -D_{\mathrm{KL}}(\mathrm{M}^\pi_\mathcal{S} \| \mathrm{M}^\star_\mathcal{S})$ with $\mathrm{M}^\star_\mathcal{S} = \mathcal{U}(\{x \in [-0.5, 0.5], y = 0]\})$	≈ -15	0
Pure Exploration	$\max_\pi \mathcal{H}(\mathrm{M}^\pi_\mathcal{S})$	≈ 8	≈ 16
Robust MDP	$\max_\pi \min\left(\langle \mathrm{M}^\pi_\mathcal{S}, R_{(0.0,0.5)}(x,y) \rangle, \langle \mathrm{M}^\pi_\mathcal{S}, 1 - R_{(0.0,0.5)}(x,y) \rangle\right)$	0	0.5
Constrained RL	$\max_\pi \langle \mathrm{M}^\pi_\mathcal{S}, R_{(0,0.5)}(x,y) \rangle$ s.t. $\langle \mathrm{M}^\pi_\mathcal{S}, R_{(0.,0.5)}(x,y) \rangle < 0.9$	0	0.9

F.2 High-dimensional evaluation

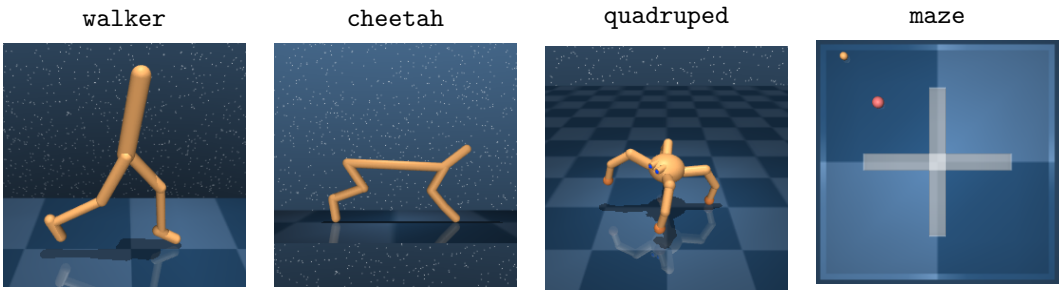


Figure 8: The four domains considered from ExORL [Yarats et al., 2022] and the Deepmind Control Suite [Tassa et al., 2018].

High-dimensional experiments revolve around locomotion and navigation tasks from the Deepmind Control Suite [Tassa et al., 2018] and use ExORL data [Yarats et al., 2022]. All evaluation parameters, including linear reward functions, are taken from Touati and Ollivier [2021]: algorithms are trained until convergence (for 3M gradient steps) with a discount factor of 0.98 (0.99 for `maze`). The inference procedure for General Utilities differs minimally from the didactic environment (detailed previously): it evaluates 1024 candidates for z, but considers slightly more samples from the estimated successor measure (2048). The chosen policy is then evaluated through a Monte Carlo estimator: we execute the policy for 10 episodes and resample visited state-action pairs according to a geometric distribution parameterized by $1 - \gamma$.

Objectives involving entropy or Kullback-Leibler divergences also rely on a K-nearest-neighbor-based estimator, with $k = 3$ [Wang et al., 2009]. For imitation learning objectives, the expert is a deterministic optimal policy for one of the tasks defined by the suite (`walk` for all domains, except for `maze`, which demonstrates `reach_bottom_left`); the stochastic variants simply inject Gaussian noise ($\sigma = 1.0$) to expert actions. The robust and constrained objectives are less direct in their definition. For simplicity, we consider the velocities of the first two degrees of freedoms for all embodiments (e.g., going right/left and up/down for `maze`), and refer to the function extracting them from a state as v_x and v_y. The robust objective is simply the minimum between the discounted averages of the two absolute velocities, i.e. $J_{\text{robust}}(\mathrm{M}^\pi) = \min(\langle \mathrm{M}^\pi, R_x \rangle, \langle \mathrm{M}^\pi, R_y \rangle)$, where $R_x(s) = \text{abs}(v_x(s))$ and $R_y(s) = \text{abs}(v_y(s))$. The constrained objective encourages high velocity in the first degree of freedom, while ensuring the velocity is not always positive: $J_{\text{constrained}}(\mathrm{M}^\pi) = \langle \mathrm{M}^\pi, R \rangle$ constrained by $\langle \mathrm{M}^\pi, R_c \rangle > 0$ with $R(s) = v_x(s)$ and $R_c(s) = \mathbf{1}_{v_x(s)<0}$.

As in this case minimum and maximum performance are not easily estimated, we do not normalize scores.

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: As claimed in the abstract and in Section 1, our main contribution is a method for zero-shot RL with general utilities. Theoretically, we show that an entropy-regularized variant of forward-backward representations may capture all solutions to general utilities. Practically, we show that this variant matches the base algorithm on linear objectives, and retrieves better solutions for general utilities.

   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 discuss limitation (both theoretical and practical) in Section 7.

   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: The main theoretical results (Theorems 4.1 and 4.2) report each assumption, and their proofs are found in Appendix A. All core formal results and formulas are cross-referenced.

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: We disclose information for reproducing our results in Section 5, and details in Appendix F. Code is available on the anonymous website.

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: We provide essential code on the anonymous website.

   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 main implementation details are presented in Appendix F, with further information available through the codebase.

   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 numerical results are accompanied by 1.96-sigma confidence intervals.

   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 report compute resources in Appendix F.

   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: We comply to the NeurIPS Code of Ethics.

   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: [N/A]

    Justification: This paper presents an algorithm for zero-shot optimization of general objectives in MDPs. As such, its potential impact falls within the societal consequences of machine learning and robotics at large.

    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: We do not believe our work poses such risks.

    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: The two main assets we use are the codebase from Tirinzoni et al. [2024], which adopts an Attribution-NonCommercial 4.0 International license, and the datasets from Yarats et al. [2022], which adopts an MIT license.

    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: Our code release is documented through its README.

    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: We do not include crodwsourcing experiments.

    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: We do not involve human subjects.

    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 method development did not involve LLMs.

    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.