From Reward-Free Pretraining to Pareto Fronts: Zero-Shot Multi-Objective Reinforcement Learning

Hicham Azmani
Vrije Universiteit Brussel
Brussels, Belgium
hicham.azmani@vub.be

Ann Nowé
Vrije Universiteit Brussel
Brussels, Belgium
ann.nowe@vub.be

Roxana Rădulescu
Utrecht University
Utrecht, the Netherlands
r.t.radulescu@uu.nl

ABSTRACT

Many real-world sequential decision problems involve multiple, conflicting objectives, yet the desired trade-offs are often unknown a priori and may change as stakeholders iteratively refine their preferences and even the objectives under consideration. In this work, we focus on the latter challenge of handling post hoc changes to objective specification. Many existing multi-objective reinforcement learning (MORL) pipelines, especially decomposition-based approaches, do not accommodate such post hoc changes efficiently and may require repeated retraining. We propose a late-binding paradigm that decouples reward-free pretraining from objective-specific solution-set construction: train a forward-backward (FB) agent on reward-free data, then recover Pareto-optimal trade-offs purely at inference time without retraining or fine-tuning. Our inference-time procedure (i) constructs a grounded candidate pool from the FB representation, (ii) selects a candidate for each preference vector via scalarised rollout evaluation, (iii) evaluates the resulting policy, and (iv) retains the non-dominated subset as an empirical Pareto front approximation. We validate our approach on two MORL benchmarks with known reference sets/fronts, Deep Sea Treasure and Fruit Tree Navigation, demonstrating that a single pre-trained FB agent can recover high-quality Pareto trade-offs across multiple environment variants and objective specifications, including previously unseen layouts and post hoc changes in selected objective subsets. Our results suggest that reward-free pretraining can serve as a practical foundation for flexible, reusable multi-objective decision support.

KEYWORDS

Reinforcement Learning, Multi-objective Reinforcement Learning, Representation Learning

1 INTRODUCTION

Many sequential decision-making problems involve multiple, conflicting objectives. Operating an energy system requires balancing cost, comfort, and emissions [12]; managing a reservoir entails trade-offs among flood risk, irrigation deficits, and ecological constraints [10]. In such settings, there is typically no single policy that simultaneously optimises all objectives. Instead, we would like to compute a set of policies that expose high-quality trade-offs to a decision-maker. In multi-objective reinforcement learning (MORL), several solution concepts capture this idea depending on assumptions about preferences and policy classes [13]. Crucially, in realistic deployments, the challenge is rarely just to compute such a set once, because the objective specification itself is often difficult to pin down. Even in single-objective RL, translating an informal goal into a precise reward is notoriously brittle. In practice, rewards are imperfect proxies: they often omit important constraints and can induce unintended behaviour under optimisation pressure. In particular, agents may exploit loopholes in the specified objective and achieve high reward while violating the designer's intent; this phenomenon is known as reward hacking [16, 18, 19]. This makes reward and constraint design an iterative process of observing behaviour and revising the specification.

Multi-objective problems further amplify this difficulty in two distinct ways. First, when several imperfect objective components must be balanced, preferred trade-offs are rarely known upfront. They may be contested among stakeholders and may only become clear after candidate solutions are evaluated. MORL is therefore frequently used in what Hayes et al. [13] call unknown or dynamic utility settings, where stakeholders cannot fully specify a stable utility function in advance and may revise preferences after inspecting candidate trade-offs. Even when one restricts to linear scalarisations, selecting weights is typically part of an iterative decision-making process rather than a one-shot modelling choice [7, 15]. Second, the specification itself may change post hoc: stakeholders may decide to reweigh objectives, introduce or remove objective dimensions, or focus on a different subset of the available criteria. Taken together, these observations suggest that early binding to a fixed objective specification can be misaligned with deployment realities.

However, many MORL pipelines implicitly assume early binding. A common practical paradigm is decomposition: repeatedly select a preference (e.g., a weight vector over objectives), solve the resulting scalarised single-objective problem using an RL algorithm, evaluate the learned policy, and iterate to improve coverage [4, 13, 23, 28]. While effective when the objective specification is fixed, this structure becomes particularly costly under post hoc changes to objective specification: each update can require re-running preference sweeps, collecting new experience, and retraining policies, sometimes dozens to hundreds of times. When environments are complex or stochastic, or when broad coverage is needed, this re-computation can become a burden.

These considerations motivate a design principle we call *late binding*: learn an objective-agnostic understanding of environment dynamics first, and bind objectives and preferences only at decision time by selecting among reusable behaviours. Recent progress in reward-free and zero-shot reinforcement learning provides a concrete mechanism for late binding [26, 27]. Rather than learning a policy from scratch for each reward specification, an agent can first learn a reward-free representation from interaction (or offline

Proc.@ of the Adaptive and Learning Agents Workshop (ALA 2026), Aydeniz, Delgrange, Mohammedalamen, Yang (eds.), May 25 – 26, 2026, Paphos, Cyprus, https://alaworkshop2026.github.io/. 2026.

data), and later instantiate task-specific behaviour with lightweight test-time computations.

In particular, the forward-backward (FB) framework learns a low-rank representation of successor measures and induces a family of latent-conditioned policies that can be queried after pretraining [8, 26]. Given a single-objective reward specified at test time, FB constructs a reward embedding and executes the corresponding latent-conditioned policy. While this offers a compelling building block for late binding, MORL decision support requires more than solving for a single reward: it requires recovering sets of non-dominated solutions that can be revisited as preferences or objective specifications change.

This paper takes a first step toward zero-shot Pareto front discovery by treating a pre-trained reward-free FB agent as a fixed policy/representation oracle and performing purely inference-time multi-objective extraction. Concretely, we (i) construct a grounded candidate pool of latents from the FB backward representation, (ii) for each preference vector, select a latent by rollout-based scalarised evaluation among candidates, (iii) evaluate the selected latent-conditioned policy to obtain a vector return, and (iv) retain the non-dominated subset as an empirical Pareto front approximation. This reframes Pareto front discovery from "many expensive RL runs" into an outer-loop selection-and-evaluation problem with respect to a single pre-trained oracle. Our focus in this paper is specifically on post hoc changes in objective specification at evaluation time, rather than on fully general evolving-objective settings. In particular, our experiments study reuse under changes in scalarisation weights, environment variants, and, in Fruit Tree Navigation, changes in which objective dimensions are considered.

Concurrent work [11] connects MORL and reward-free RL by modifying FB training to better cover MORL-relevant regions, injecting preference-weighted reward information to guide exploration and learning. We propose a different interface: treat the MORL family of preference-induced problems as sharing a single underlying reward-free MDP, and separate (i) objective-agnostic pretraining of dynamics-dependent representations from (ii) objective-specific solution-set construction at decision time. Accordingly, we ask a narrower question: can a vanilla reward-free pre-trained FB agent already provide sufficient behavioural coverage to extract useful Pareto sets via purely inference-time latent proposal and selection?

Our *contributions* can be summarised as follows:

- **Inference-time Pareto front extraction from a pre-trained agent.** We show that a reward-free pre-trained FB agent can be treated as a fixed policy/representation oracle and used to recover Pareto-optimal trade-offs via purely inference-time latent proposal and selection, without any retraining or fine-tuning.
- **Objective-agnostic reuse validated against known reference fronts.** Using MORL benchmarks with known reference sets/fronts, we demonstrate that the same pre-trained representation can be reused to recover high-quality Pareto trade-offs under post hoc changes in objective specification, including different objective subsets, by modifying only the evaluation-time definition.

2 BACKGROUND

We summarise the notation and core concepts used throughout the paper: (single-objective) RL, multi-objective RL, and reward-free zero-shot RL via forward-backward (FB) representations. The key bridge is that reward-free methods learn dynamics-dependent representations (and, in FB, a family of latent-conditioned policies) that can be queried after pretraining. Our contribution is to use such a pre-trained policy family as an oracle to recover *sets* of non-dominated solutions at inference time.

2.1 Reinforcement Learning

A Markov decision process (MDP) is a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, r, \gamma \rangle$ with state space $\mathcal{S}$, action space $\mathcal{A}$, transition function $P(s' \mid s, a)$, reward function $r(s, a, s') \in \mathbb{R}$, and discount $\gamma \in [0, 1)$. A stationary policy $\pi(a \mid s)$ induces trajectories $a_t \sim \pi(\cdot \mid s_t)$ and $s_{t+1} \sim P(\cdot \mid s_t, a_t)$. The discounted return is $G^\pi = \sum_{t \geq 0} \gamma^t r(s_t, a_t, s_{t+1})$. The value and action-value functions are

$$V^\pi(s) = \mathbb{E}[G^\pi \mid s_0 = s], \qquad Q^\pi(s, a) = \mathbb{E}[G^\pi \mid s_0 = s, a_0 = a].$$

2.2 Multi-Objective Reinforcement Learning

A multi-objective MDP (MOMDP) is a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, \gamma, \mathbf{r} \rangle$, where $\mathbf{r}(s, a, s') \in \mathbb{R}^d$ is a d-dimensional reward vector. For a policy π, define the discounted return vector and value function

$$\mathbf{G}^\pi = \sum_{t \geq 0} \gamma^t, \mathbf{r}(s_t, a_t, s_{t+1}), \qquad \mathbf{V}^\pi(s) = \mathbb{E}[\mathbf{G}^\pi \mid s_0 = s].$$

A decision maker's preferences are modelled by a utility function $u : \mathbb{R}^d \to \mathbb{R}$ that maps return vectors to a scalar. When u is unknown or may change, we seek a *set* of policies that covers a range of plausible preferences [13].

Pareto dominance and Pareto front. For $\mathbf{v}, \mathbf{v}' \in \mathbb{R}^d$, we say $\mathbf{v}$ Pareto-dominates $\mathbf{v}'$ if

$$\mathbf{v} \succ_P \mathbf{v}' \iff (\forall i,\ v_i \geq v'_i) \ \wedge \ (\exists i,\ v_i > v'_i).$$

The Pareto front is the set of achievable non-dominated value vectors (and the corresponding Pareto-optimal policies). In practice, algorithms often learn a finite Pareto coverage set (PCS) that approximates this front [13]. In this paper, we obtain an empirical approximation by evaluating a collection of policies and retaining the non-dominated subset of observed returns.

Linear scalarisation and convex coverage set. A common special case is a positive linear utility

$$u(\mathbf{v}) = \mathbf{w}^\top \mathbf{v}, \quad \mathbf{w} \in \Delta^{d-1},$$

where $\Delta^{d-1} = \mathbf{w} \in \mathbb{R}^d_+ : \sum_i w_i = 1$ is the probability simplex. The relevant solution concept is the convex coverage set (CCS): a set that contains at least one optimal policy for every $\mathbf{w} \in \Delta^{d-1}$ [13].

2.3 Reward-Free Zero-Shot Reinforcement Learning

Zero-shot RL aims to produce near-optimal behaviour for a reward specified *after* a reward-free pretraining phase, without additional learning, planning, or fine-tuning at test time [27]. Formally, we consider a reward-free MDP $\langle \mathcal{S}, \mathcal{A}, P, \gamma \rangle$ and learn a compact representation from transitions (s_t, a_t, s_{t+1}) such that, once a bounded

reward function r is specified, the agent can compute a policy using only simple computations based on the learned representation and the reward [27].

Successor measures. A policy π induces a discounted occupancy over future state-action pairs. In continuous spaces, this is expressed via the successor measure [26]:

$$M^{\pi}(s_0, a_0, X) \;:=\; \sum_{t \geq 0} \gamma^t \Pr\big((s_t, a_t) \in X \mid s_0, a_0, \pi\big),$$

for measurable sets $X \subseteq \mathcal{S} \times \mathcal{A}$. This object captures dynamics independently of the reward; given M^{π}, the return under any bounded reward can be computed by integration against the occupancy [26].

Forward-Backward (FB) representations. The FB framework learns a tractable, low-rank model of successor measures in a latent space $Z \simeq \mathbb{R}^{d_z}$ by training a *forward* map $F : \mathcal{S} \times \mathcal{A} \times Z \to Z$ and a *backward* map $B : \mathcal{S} \times \mathcal{A} \to Z$ such that, for a suitable data distribution ρ,

$$M^{\pi_z}(s_0, a_0, \mathrm{d}s, \mathrm{d}a) \;\approx\; F(s_0, a_0, z)^{\top} B(s, a), \rho(\mathrm{d}s, \mathrm{d}a),$$

with $\pi_z(s) \in \arg\max_{a \in \mathcal{A}} F(s, a, z)^{\top} z$, and where $z \in Z$ indexes a family of latent-conditioned policies [26]. Given a bounded reward r, FB forms a reward embedding (up to ρ)

$$z_r \;:=\; \mathbb{E} * (s, a) \sim \rho\big[r(s, a), B(s, a)\big],$$

and executes $\pi * z_r$. In MORL, a preference $\mathbf{w}$ induces a scalar reward $r_{\mathbf{w}}$ (e.g., via linear scalarization). Our work studies how to cover many such preferences by selecting multiple latents purely at inference time and retaining the non-dominated set of observed returns.

Idealized guarantee. If the FB factorisation holds exactly, then for any bounded reward function r, the induced policy π_{z_r} is optimal for r, and the optimal action-value function satisfies

$$Q_r^{\star}(s, a) \;=\; F(s, a, z_r)^{\top} z_r.$$

[26]. In practice, approximate FB representations yield approximately optimal policies, with sub-optimality controlled by representation error [26].

3 RELATED WORK

3.1 Reward-free and Zero-shot Reinforcement Learning

Reward-free and zero-shot RL studies how to pretrain agents without committing to a downstream reward, and later instantiate task-specific behaviour with minimal computation. A prominent line builds on successor representations: successor features (SF) and their universal variants combine a task description with efficient policy evaluation and reuse, often via generalised policy improvement (GPI) [9]. These approaches are attractive when rewards can be expressed (exactly or approximately) as linear combinations of features, and they provide a clean mechanism for transfer to unseen tasks.

Successor-measure approaches remove the need to predefine reward features by directly learning occupancies. Forward-backward (FB) representations learn low-rank models of successor measures and induce latent-conditioned policies that can be queried at test time using a reward embedding [26, 27]. Touati et al. [27] provides a systematic evaluation of SF variants versus FB, highlighting the sensitivity of SF to the choice of base features and showing strong, consistent empirical performance for FB under suitable data coverage. Several recent works study limitations of successor-measure zero-shot RL and propose remedies, including conservative variants for low-diversity offline data and exploration-augmented objectives to improve coverage in online unsupervised settings [14, 25]. Beyond FB/SF, Proto Successor Measure Agarwal et al. [2] proposes a basis-function view of behaviour: it learns policy-independent bases for successor measures, enabling downstream behaviours to be expressed by combining these bases and providing a different approach to zero-shot task solving from reward-free data [2].

3.2 Multi-objective Reinforcement Learning and Coverage-set Construction

MORL methods differ primarily in (i) whether they aim to learn a single policy for a fixed utility function or a set of policies covering a range of preferences, and (ii) what assumptions are made about the utility function and admissible policy class. For a broader overview of solution concepts, algorithm families, and evaluation metrics, we refer the reader to the practical guide of Hayes et al. [13].

A common approach constructs a coverage set by repeatedly solving scalarized (or otherwise constrained) single-objective subproblems and using an outer loop to select new preferences to query [13]. Recent work refines this paradigm with principled preference selection and reuse via generalised policy improvement (GPI), improving sample efficiency when building convex coverage sets under linear scalarization [4]. Beyond the convex case, IPRO provides a decomposition method aimed at unveiling non-convex Pareto fronts (notably for deterministic policies) by iteratively solving constrained single-objective problems, with convergence guarantees and explicit bounds on remaining approximation error [23]. A complementary direction generalises learning across preferences by training a single network conditioned on a preference or target return. Pareto Conditioned Networks (PCN) [22] learn a single policy conditioned on a desired multi-objective return, enabling direct execution of behaviours corresponding to different Pareto-efficient trade-offs and avoiding assumptions of convexity that arise with purely linear-scalarization-based methods.

MORL and multi-task RL connections. A related perspective comes from work that connects MORL to multi-task transfer RL. Alegre et al. [4] shows that, under linear scalarization, multi-objective action-value functions can be viewed as successor-feature (SF) decompositions when reward vectors are interpreted as linear reward features, and that learning a CCS in this setting is closely related to optimal policy transfer across tasks that are linear combinations of features [4, 5]. In our work, we operate at the level of successor *measures* rather than fixed successor features: FB can be seen as learning a low-rank successor-measure representation and recovers SF-style methods as a special case when rewards are restricted to be linear in user-provided features [26].

Another relevant direction for handling minimal reward specifications is Lexicographic RL, which focuses on satisfying objectives in a predefined order of importance rather than requiring explicit weights [24]. This is a distinct way of specifying preferences from

the weighted trade-off setting considered in our work, but is relevant as an alternative approach when hierarchical objective priorities are known in advance.

Reward-free viewpoints on MORL. Concurrent work proposes MORL-FB, which adapts FB training to the multi-objective setting via preference-guided exploration and additional learning signals [11]. In contrast, our work keeps reward-free pretraining unchanged and studies purely inference-time solution-set extraction from a fixed pre-trained policy/representation oracle; see Section 1 for a detailed comparison.

4 INFERENCE-TIME PARETO FRONT EXTRACTION FROM A REWARD-FREE FB AGENT

A recurring theme in MORL is that multi-objective solution sets can often be obtained by *reusing* single-objective machinery: one formulates an outer-loop procedure that proposes scalarized or constrained subproblems, and solves each using a standard RL solver. This decomposition perspective is attractive because it inherits the strengths and guarantees of the underlying single-objective method [23], and it connects naturally to transfer-style views in which different preferences correspond to different tasks/policies [3].

In this work, we push the decomposition idea further by replacing repeated single-objective training with reward-free *pretraining* plus zero-shot adaptation. While many MORL methods treat each preference (or scalarization) as a separate MDP instance to be solved with a single-objective RL algorithm, we view them as lightweight variants of a single reward-free MDP: the transition dynamics are unchanged, only the evaluation utility differs. This lets us pretrain an FB agent once and, at inference time, select an appropriate latent-conditioned policy for any preference vector, thereby obtaining a Pareto front approximation without retraining.

Problem setting. Let the environment produce a vector reward $\mathbf{r}_t \in \mathbb{R}^d$ and consider linear preferences $w \in \Delta^{d-1}$ with scalarized return

$$J_w(\pi) \;=\; \mathbb{E}\left[\sum_{t=0}^{T-1} w^\top \mathbf{r}_t\right].$$

We assume access to a *reward-free* pre-trained forward-backward (FB) agent inducing a family of latent-conditioned policies $\{\pi_z\}_{z\in\mathbb{R}^{d_z}}$. Our goal is to approximate the Pareto front by selecting, for each preference w, a latent $z(w)$ that yields a high scalarized return, evaluating the corresponding policy to obtain a vector return, and retaining the non-dominated set. Our pipeline is entirely *inference-time*: (i) construct a grounded candidate set of latents, (ii) select a latent for each preference via scalarized evaluation, (iii) evaluate the resulting policy to obtain a vector return, and (iv) compute the non-dominated subset.

FB pretraining (DQN-style temporal-difference learning). In all experiments, the pre-trained checkpoint is obtained by running the standard reward-free FB algorithm with a DQN-style training loop [26, 27]. FB learns a forward network $F_\theta(s,a,z) \in \mathbb{R}^{d_z}$ and a backward network $B_\omega(\cdot) \in \mathbb{R}^{d_z}$, together defining a low-rank model of successor measures under the latent-conditioned greedy policy

$$\pi_z(s) \in \arg\max_{a\in\mathcal{A}} F_\theta(s,a,z)^\top z.$$

Pretraining is reward-free and off-policy from a replay buffer of transitions (s_t, a_t, s_{t+1}), using target networks updated slowly as in DQN. At each update, a latent z is sampled (typically from a normalised Gaussian), and the FB TD objective enforces Bellman consistency of the successor-measure factorisation. In practice, exploration during pretraining follows an ϵ-greedy policy w.r.t. the induced score $F_\theta(s,a,z)^\top z$ for randomly sampled z, collecting diverse reward-free experience. Once pretraining is complete, FB can be queried for a downstream scalar reward r by forming a reward embedding $z_r \approx \mathbb{E}_{(s,a)\sim\rho}[r(s,a)\,B_\omega(s,a)]$ and executing π_{z_r}, without additional learning or planning.

(i) Grounded latent candidates. The main practical challenge is inferring a suitable latent from *reward preferences* alone. Directly proposing latents from a simple prior and selecting by $w^\top$ return can lead to severe under-coverage of the Pareto front, because many proposals correspond to behaviours not supported by the learned representation. We therefore restrict inference to latents grounded in the backward representation: collect observations (or states) $\{o_i\}_{i=1}^N$ from reward-free rollouts and build

$$\mathcal{Z} \;=\; \{\, B(o_i) \;:\; i = 1,\dots,N \,\},$$

(optionally normalised), capturing latent directions associated with reachable experience.[1] In practice, $\mathcal{Z}$ can be augmented with lightweight proposals such as normalised Gaussian samples.

(ii) Preference-conditioned latent selection. For each w, we estimate the expected vector return $\hat{\mathbf{R}}(z)$ of π_z using rollouts and select

$$z(w) \;=\; \arg\max_{z\in\mathcal{Z}} \; w^\top \hat{\mathbf{R}}(z).$$

We then run additional evaluation rollouts under $\pi_{z(w)}$ and record the resulting vector return $\mathbf{R}(w)$. To reduce selection noise, we average over multiple evaluation episodes per candidate and optionally use a two-stage procedure (a short-horizon prescore followed by long-horizon refinement of the top-k candidates), using shared randomness across candidates to ensure fair comparison.

(iii) Pareto front approximation. Given the set of returns $\{\mathbf{R}(w)\}$ across preferences, we compute the non-dominated subset to obtain an empirical Pareto front approximation.

Remark: goals vs. reward preferences. When an explicit goal state g is available, FB provides a direct latent via $z = B(g)$. In MORL, however, preferences are specified over reward components, and multiple trajectories (or terminal states) can correspond to the same trade-off. Our extraction procedure, therefore, infers task-relevant latents from reward preferences rather than privileged goal information.

5 EXPERIMENTS

We evaluate MO-FB on two MORL benchmarks with known reference sets/fronts, enabling direct verification of Pareto front/CCS recovery.

[1] In our implementation, B is applied to the state/observation encoder output.

Environments. **Deep Sea Treasure ($d = 2$).** We evaluate on the bi-objective Deep Sea Treasure (DST) [1, 28] benchmark, a classic MORL gridworld where the agent controls a submarine that navigates a 2D ocean map to collect one of several treasures. Each episode terminates upon reaching a treasure, giving a 2D reward vector that trades off (i) treasure value and (ii) a per-step time/fuel penalty (constant -1), so that deeper (more distant) treasures tend to be more valuable but require longer trajectories. DST is particularly useful because its Pareto-optimal set/front is known for standard configurations, enabling direct comparison between the extracted Pareto front and a ground-truth reference, rather than relying solely on scalar evaluation metrics such as hypervolume or cardinality.

Fruit Tree Navigation ($d \leq 6$). We also evaluate on Fruit Tree Navigation, a discrete many-objective benchmark introduced by Yang et al. [28]. The environment is a full binary tree of depth d: from the root, the agent repeatedly chooses between the left and right subtree until it reaches a leaf (terminal) node. Each leaf is assigned a 6-dimensional reward vector $r \in \mathbb{R}^6$ corresponding to nutritional components of the harvested fruit: Protein, Carbs, Fats, Vitamins, Minerals, Water. The rewards are constructed such that every leaf is optimal for some preference vector, meaning all leaves lie on the convex coverage set (CCS) [28]. Consequently, FTN provides a controlled setting where the entire CCS is known by design, making it suitable for validating whether our method can recover broad preference-conditioned solution coverage in a higher-dimensional objective space.

Evaluation protocol. Inference-time Pareto front extraction proceeds as follows. We first freeze a reward-free pre-trained FB checkpoint. At evaluation time, we sample preference vectors over the objective simplex, construct a candidate latent set primarily from backward embeddings of visited states, optionally augment this pool with lightweight proposals depending on the extraction strategy, and, for each preference, select a latent by scalarized episodic evaluation. We then roll out the policy induced by the selected latent, record its vector return, and extract the non-dominated subset of returns as the recovered Pareto front. No retraining or fine-tuning is performed during extraction.

5.1 Pareto front extraction from a single pre-trained FB agent (DST)

In this experiment, we test whether a reward-free pre-trained FB agent contains sufficient coverage to recover the reference Pareto front across multiple Deep Sea Treasure (DST) settings *without retraining*.

Setup. We evaluate a *single* pre-trained FB checkpoint on three DST variants (`concave`, `convex`, `mirrored`). The checkpoint is trained once on the *default* DST layout and then evaluated zero-shot on all variants. Importantly, `concave` and `convex` share the *same underlying grid layout and dynamics* but use *different reward specifications* (yielding concave vs. convex Pareto geometry), while `mirrored` uses a *spatially flipped* layout that induces different transitions and visitation structure. For each variant, we perform *inference-time* latent selection by sweeping scalarization weights and selecting a latent $z(w)$ from a candidate set derived from backward-network activations along rollouts (a visited-state latent pool), optionally augmented with lightweight proposals (e.g., normalised Gaussian samples). For each weight, we roll out the selected latent-conditioned policy and retain the non-dominated set of observed vector returns.

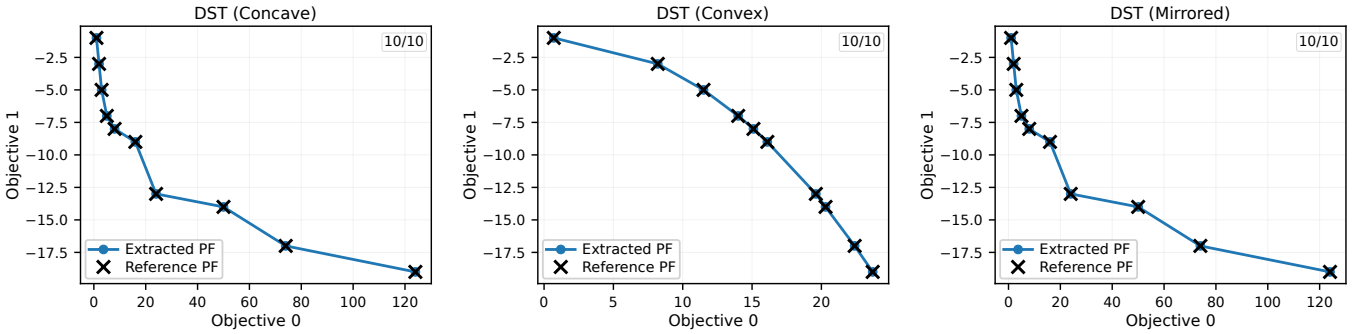
For DST, we report both representative single-seed Pareto front overlays and aggregated 5-seed robustness results. We evaluate preference sweeps with `num_weights` $\in \{41, 121\}$ across all three variants, and additionally study a denser concave-only sweep with $\{41, 81, 121, 161\}$. Unless otherwise stated, the main-text visualisations use `num_weights=121`.

Result. Figure 1 summarises extraction from the same pre-trained agent across all three variants, with *no retraining*. In the shown representative run (seed 1, `num_weights=121`), we recover the full front on all three cases (`concave`: 10/10, `convex`: 10/10, `mirrored`: 10/10). Since `concave` and `convex` differ only in the *reward function* (not the dynamics), successful recovery on both provides evidence that our post-hoc extraction is *not limited to the convex coverage set* induced by linear scalarization during evaluation; instead, the fixed pre-trained policy family contains sufficient diversity to recover the full Pareto front in these settings. Moreover, successful extraction on `mirrored` indicates that the learned representation and latent-conditioned policy family can transfer to a *previously unseen* layout with a different transition structure. Across variants, different *generic* candidate-generation schemes perform best: `concave` is solved with visited-pool decoding (`pool_only` with episode-wise z resampling during pool collection), while `convex` and `mirrored` are solved with `pool_plus_gaussian`. Importantly, these differences reflect inference-time candidate support rather than any change to the pre-trained model.

Robustness across seeds. To assess robustness, we additionally ran 5-seed sweeps on all three DST variants for `num_weights` $\in \{41, 121\}$ and report the best-performing strategy among the two non-oracle schemes above. At `num_weights=41`, recovery is less reliable (`concave`: 0.84 ± 0.12, `convex`: 1.00 ± 0.00, `mirrored`: 0.82 ± 0.10). Increasing to `num_weights=121` improves consistency (`concave`: 0.94 ± 0.05, `convex`: 1.00 ± 0.00, `mirrored`: 0.98 ± 0.04). In terms of perfect recovery, `num_weights=121` yields 2/5 perfect seeds on `concave`, 5/5 on `convex`, and 4/5 on `mirrored`.

On the concave map, an additional sweep over `num_weights` $\in \{41, 81, 121, 161\}$ shows improvement up to 121, after which gains plateau (0.80 ± 0.13, 0.88 ± 0.10, 0.92 ± 0.08, 0.92 ± 0.08). This suggests that denser preference sweeps substantially improve inference reliability, while the remaining misses are primarily attributable to inference-time preference coverage and candidate selection rather than a clear inability of the pre-trained FB representation to support the required behaviours.

Computational budget. The reward-free pretraining budget is incurred once per environment: 1M environment steps for DST. All Pareto front extraction results are then obtained from this frozen checkpoint via an inference-time outer loop, rather than by retraining policies for each new objective specification. For the DST robustness experiments, this outer loop evaluates `num_weights` $\in \{41, 121\}$ using 20 episodes per weight, with candidate-pool evaluation based on a candidate rollout size of 96 and 3 candidate



(a) `Concave`. (b) `Convex`. (c) `Mirrored`.

Figure 1: Pareto front extraction in Deep Sea Treasure from a single reward-free pre-trained FB agent across three environment variants, using `num_weights=121` at evaluation time. In each panel, the blue curve shows the extracted non-dominated set and the black x markers show the reference Pareto front. The in-panel annotation reports the number of reference Pareto points recovered. The shown plots are representative single-seed overlays; the corresponding 5-seed recovery statistics are reported in the text.

eval episodes. Thus, objective changes are handled by repeated evaluation-time latent selection and rollout, not by re-running RL training.

5.2 Reuse under changing objective subsets (FTN)

We test whether the same pre-trained FB representation can be reused when changing the objective specification at evaluation time, including both scalarization weights and the selected objective dimensions, without retraining.

Setup. Using one pre-trained FB agent on FTN, we extract Pareto fronts for multiple 2D projections by selecting objective pairs and sweeping preferences at evaluation time. The main-text FTN figures are representative single-seed results (seed 1) obtained with `num_weights=61`; unlike DST, we do not currently report a multi-seed robustness summary for these FTN plots.

Results. Using one pre-trained FB policy on Fruit Tree, we extracted Pareto fronts for multiple objective subsets without any additional training. For objective pairs (0,1) and (4,5), our post-hoc preference sweep over latent-conditioned policies recovered the full projected reference front in both cases (7/7 and 5/5 points, respectively). This supports the claim that the learned FB representation can be reused across downstream objective specifications: changing the objective weights and even the selected objective dimensions at evaluation time is sufficient to recover the corresponding Pareto trade-offs in these examples. We emphasise that these FTN results are currently representative single-seed demonstrations rather than multi-seed robustness summaries. An additional seed-1 result for objectives (2,3) yields partial recovery (8/10), indicating that objective-subset reuse is promising but not uniformly perfect under the current extraction setup.

Computational budget. The reward-free pretraining budget for FTN is incurred once (200k environment steps for the checkpoint used here). Pareto front extraction then reuses this frozen checkpoint via an inference-time outer loop with `num_weights=61`, 50 episodes per weight, and horizon 6. As in DST, changing the evaluation-time objective definition does not trigger any retraining or fine-tuning.

6 CONCLUSION & FUTURE WORK

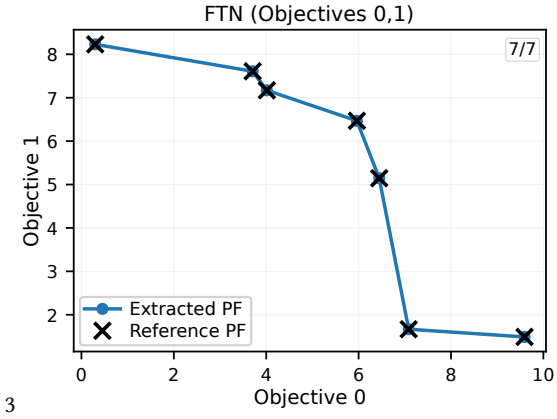
We introduced an inference-time procedure for extracting multi-objective solution sets from a reward-free pre-trained forward–backward (FB) agent, reframing Pareto front discovery as selection and evaluation over a single pre-trained policy/representation oracle. Using benchmarks with known reference sets, we showed that (i) non-trivial Pareto fronts can be recovered without any retraining or fine-tuning, and (ii) the same pre-trained representation can be reused to recover trade-offs under changing objective specifications.

6.1 Limitations

While our results demonstrate the potential of late-binding MORL, several limitations remain. First, our approach relies on a *strong coverage assumption*: the reward-free pretraining phase must produce sufficient behavioural diversity to cover the Pareto trade-offs eventually requested. While our experiments show that simple latent randomisation is effective in the studied domains, more complex environments may require explicitly coverage-augmented unsupervised objectives [25]. Second, our current evaluation is limited to discrete benchmarks with known ground-truth fronts. Scalability to high-dimensional continuous control (e.g., robotics) remains to be demonstrated and likely requires more sophisticated latent proposal and selection schemes to handle the vast policy space induced by the representation. Finally, the reliability of zero-shot extraction depends on the quality of the learned measure factorisation, which can be sensitive to representation error in complex or highly stochastic domains.

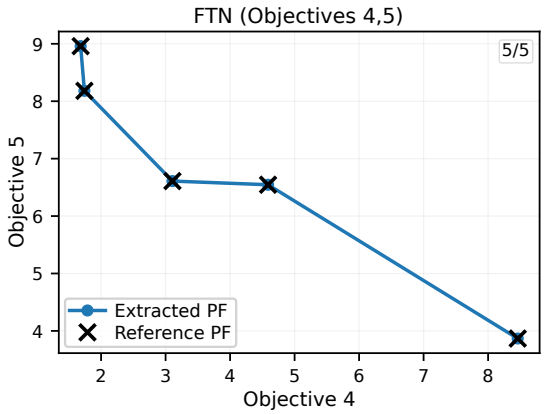
6.2 Future Work

Scaling to larger benchmarks and stronger baselines. Our experiments focused on small MOMDPs with known reference fronts, enabling direct verification of front recovery rather than relying



3

(a) `Objectives (0,1)`.



(b) `Objectives (4,5)`.

Figure 2: Pareto front extraction in Fruit Tree Navigation under post hoc changes in the selected objective subset, using the same reward-free pre-trained FB agent in both cases. In each panel, the blue curve shows the extracted non-dominated set and the black x markers show the corresponding reference Pareto front for the selected objective pair. The in-panel annotation reports the number of reference Pareto points recovered. These main-text FTN plots are representative single-seed results (seed 1).

solely on scalar metrics. A natural next step is to evaluate in higher-dimensional and continuous-control benchmarks such as the MO-Gymnasium / MO-MuJoCo suite [6] and compare against state-of-the-art MORL methods on these domains [17, 22]. Such experiments would clarify how far reward-free pretraining plus inference-time extraction can scale, and when additional preference-conditioned training signals become necessary.

Data coverage and exploration for reward-free pretraining. Both FB and our inference-time selection rely on the diversity and quality of the reward-free data used for pretraining. In our current setup, data collection follows a simple latent-randomisation strategy (sampling a latent z from a prior and rolling out the corresponding policy π_z), which can be suboptimal in large environments. An important direction is to integrate stronger exploration and unsupervised skill-discovery objectives during pretraining, for example, intrinsic-motivation methods such as curiosity and random network distillation (RND) [21], or metric-aware unsupervised RL objectives such as METRA [20]. A complementary direction is to incorporate exploration-aware extensions of FB that explicitly target data coverage limitations in the online unsupervised setting [25]. Overall, improving pretraining coverage should directly improve both representation quality and the downstream Pareto coverage achievable by inference-time extraction.

Offline MORL and limited-data regimes. FB naturally supports learning from offline datasets, suggesting a promising connection to offline MORL. A key question is how Pareto front extraction degrades under small or low-diversity datasets, and how failure modes studied in offline zero-shot RL (e.g., distribution shift and overestimation in learned value or measure models) impact latent selection. Conservative variants developed for offline zero-shot RL [14] are a natural starting point to mitigate out-of-distribution errors that can harm inference-time selection. Systematic experiments across dataset sizes and qualities would help characterise when reward-free offline pretraining can serve as a practical "policy oracle" for multi-objective decision support.

Principled preference selection and oracle-based outer loops. Our current pipeline uses a simple preference sweep and relies on random sampling of weight vectors to obtain coverage. A more principled approach would combine reward-free pretraining with outer-loop methods that adaptively select which scalarizations to query, such as GPI-based preference prioritisation [4] or divide-and-conquer Pareto front discovery methods such as IPRO [23]. From our viewpoint, these algorithms could treat the pre-trained policy family as a fast inner solver, replacing repeated training with inference-time evaluation and selection. This raises new questions about how to design query strategies and stopping criteria when the inner solver is approximate, and how to couple principled preference selection with latent proposal in a way that preserves coverage guarantees.

ACKNOWLEDGMENTS

Hicham Azmani is supported by the Research Foundation – Flanders (FWO), grant number 1SA9826N. Roxana Rădulescu is supported by the DECIDE project (NWA.1766.24.031) funded by the Dutch Research Council (NWO) under the Dutch Research Agenda (NWA). This research was supported by funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" program.

REFERENCES

[1] Axel Abels, Diederik M. Roijers, Tom Lenaerts, Ann Nowé, and Denis Steckelmacher. 2019. Dynamic Weights in Multi-Objective Deep Reinforcement Learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA (Proceedings of Machine Learning Research)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 11–20. http://proceedings.mlr.press/v97/abels19a.html

[2] Siddhant Agarwal, Harshit Sikchi, Peter Stone, and Amy Zhang. 2024. Proto Successor Measure: Representing the Space of All Possible Solutions of Reinforcement Learning. *CoRR* abs/2411.19418 (2024). https://doi.org/10.48550/ARXIV.2411.19418

[3] Lucas N. Alegre. 2025. *Sample-efficient multi-task and multi-objective reinforcement learning by combining multiple behaviors.* Ph.D. thesis. Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil. https://lume.ufrgs.br/handle/10183/290816

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

[5] Lucas Nunes Alegre, Ana L. C. Bazzan, and Bruno C. da Silva. 2022. Optimistic Linear Support and Successor Features as a Basis for Optimal Policy Transfer. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA (Proceedings of Machine Learning Research)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (Eds.). PMLR, 394–413. https://proceedings.mlr.press/v162/alegre22a.html

[6] Lucas N. Alegre, Florian Felten, El-Ghazali Talbi, Grégoire Danoy, Ann Nowé, Ana L. C. Bazzan, and Bruno C. da Silva. 2022. MO-Gym: A Library of Multi-Objective Reinforcement Learning Environments. In *Proceedings of the 34th Benelux Conference on Artificial Intelligence BNAIC/Benelearn 2022*.

[7] Valerie Belton and Theodor Stewart. 2002. *Multiple criteria decision analysis: an integrated approach.* Springer Science & Business Media.

[8] Léonard Blier, Corentin Tallec, and Yann Ollivier. 2021. Learning Successor States and Goal-Dependent Values: A Mathematical Viewpoint. *CoRR* abs/2101.07123 (2021). arXiv:2101.07123 https://arxiv.org/abs/2101.07123

[9] Diana Borsa, André Barreto, John Quan, Daniel J. Mankowitz, Hado van Hasselt, Rémi Munos, David Silver, and Tom Schaul. 2019. Universal Successor Features Approximators. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. https://openreview.net/forum?id=S1VWjiRcKX

[10] A. Castelletti, F. Pianosi, and M. Restelli. 2013. A multiobjective reinforcement learning approach to water resources systems operation: Pareto frontier approximation in a single run. *Water Resources Research* 49, 6 (2013), 3476–3486. https://doi.org/10.1002/wrcr.20295

[11] Ying-Tu Chen, Wei Hung, Bing-Shu Wu, Zhang-Wei Hong, and Ping-Chun Hsieh. 2026. A Reward-Free Viewpoint on Multi-Objective Reinforcement Learning. In *The Fourteenth International Conference on Learning Representations*. https://openreview.net/forum?id=IwiwmY3Mzz

[12] Zhuoming Deng, Zhilin Lu, Zhifei Guo, Wenfeng Yao, Wenmeng Zhao, Baorong Zhou, and Chao Hong. 2020. Coordinated optimization of generation and compensation to enhance short-term voltage security of power systems using accelerated multi-objective reinforcement learning. *IEEE Access* 8 (2020), 34770–34782.

[13] Conor F. Hayes, Roxana Radulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M. Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel de Oliveira Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers. 2022. A practical guide to multi-objective reinforcement learning and planning. *Auton. Agents Multi Agent Syst.* 36, 1 (2022), 26. https://doi.org/10.1007/S10458-022-09552-Y

[14] Scott R. Jeen, Tom Bewley, and Jonathan M. Cullen. 2024. Zero-Shot Reinforcement Learning from Low Quality Data. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (Eds.). http://papers.nips.cc/paper_files/paper/2024/hash/1e38b2a0b77541b14a3315c99697b835-Abstract-Conference.html

[15] Ralph L Keeney. 1993. Value-focused thinking: A path to creative decision making. *Interfaces* 23, 3 (1993), 62–67.

[16] Victoria Krakovna, Jonathan Uesato, Vladimir Mikulik, Matthew Rahtz, Tom Everitt, Ramana Kumar, Zac Kenton, Jan Leike, and Shane Legg. 2020. *Specification gaming: the flip side of AI ingenuity.* Technical Report. DeepMind. https://deepmind.google/discover/blog/specification-gaming-the-flip-side-of-ai-ingenuity/

[17] Haoye Lu, Daniel Herman, and Yaoliang Yu. 2023. Multi-Objective Reinforcement Learning: Convexity, Stationarity and Pareto Optimality. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. https://openreview.net/forum?id=TjEzIsyEsQ6

[18] David Manheim and Scott Garrabrant. 2018. Categorizing Variants of Goodhart's Law. *arXiv preprint arXiv:1803.04585* (2018). https://arxiv.org/abs/1803.04585

[19] Alexander Pan, Kush Bhatia, and Jacob Steinhardt. 2022. The Effects of Reward Misspecification: Mapping and Mitigating Misaligned Models. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/2201.03544

[20] Seohong Park, Oleh Rybkin, and Sergey Levine. 2024. METRA: Scalable Unsupervised RL with Metric-Aware Abstraction. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net. https://openreview.net/forum?id=c5pwL0Soay

[21] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. 2017. Curiosity-driven Exploration by Self-supervised Prediction. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017 (Proceedings of Machine Learning Research)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, 2778–2787. http://proceedings.mlr.press/v70/pathak17a.html

[22] Mathieu Reymond, Eugenio Bargiacchi, and Ann Nowé. 2022. Pareto Conditioned Networks. In *21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022*, Piotr Faliszewski, Viviana Mascardi, Catherine Pelachaud, and Matthew E. Taylor (Eds.). International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 1110–1118. https://doi.org/10.5555/3535850.3535974

[23] Willem Röpke, Mathieu Reymond, Patrick Mannion, Diederik M. Roijers, Ann Nowé, and Roxana Radulescu. 2025. Divide and Conquer: Provably Unveiling the Pareto Front with Multi-Objective Reinforcement Learning. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025*, Sanmay Das, Ann Nowé, and Yevgeniy Vorobeychik (Eds.). International Foundation for Autonomous Agents and Multiagent Systems / ACM, 1774–1783. https://doi.org/10.5555/3709347.3743813

[24] Joar Skalse, Lewis Hammond, Charlie Griffin, and Alessandro Abate. 2022. Lexicographic Multi-Objective Reinforcement Learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, Luc De Raedt (Ed.). ijcai.org, 3430–3436. https://doi.org/10.24963/IJCAI.2022/476

[25] Jingbo Sun, Songjun Tu, Qichao Zhang, Haoran Li, Xin Liu, Yaran Chen, Ke Chen, and Dongbin Zhao. 2025. Unsupervised Zero-Shot Reinforcement Learning via Dual-Value Forward-Backward Representation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=0QnKnt411O

[26] Ahmed Touati and Yann Ollivier. 2021. Learning One Representation to Optimize All Rewards. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (Eds.). 13–23. https://proceedings.neurips.cc/paper/2021/hash/003dd617c12d444ff9c80f717c3fa982-Abstract.html

[27] Ahmed Touati, Jérémy Rapin, and Yann Ollivier. 2023. Does Zero-Shot Reinforcement Learning Exist?. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. https://openreview.net/forum?id=MYEap_OcQI

[28] Runzhe Yang, Xingyuan Sun, and Karthik Narasimhan. 2019. A Generalized Algorithm for Multi-Objective Reinforcement Learning and Policy Adaptation. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). 14610–14621. https://proceedings.neurips.cc/paper/2019/hash/4a46fbfca3f1465a27b210f4bdfe6ab3-Abstract.html