Sample-Efficient Multi-Objective Learning via Generalized Policy Improvement Prioritization

Lucas N. Alegre[1,2] Ana L. C. Bazzan[1] Diederik M. Roijers[2] Ann Nowé[2] Bruno C. da Silva[3]

[1] Universidade Federal do Rio Grande do Sul [2] AI-Lab - Vrije Universiteit Brussel [3] University of Massachusetts

lnalegre@inf.ufrgs.br @lnalegre github.com/LucasAlegre/morl-baselines

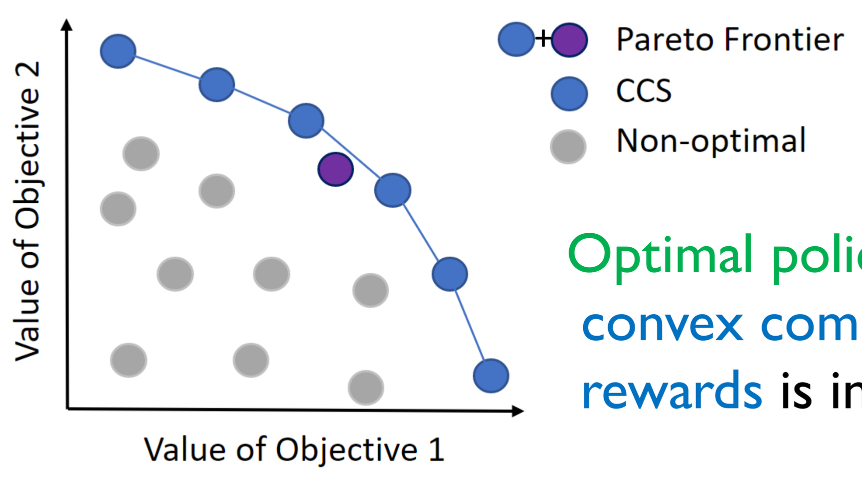
Multi-Objective Reinforcement Learning

Multi-objective reward

$\mathbf{r}: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto \mathbb{R}^m$

Goal: find optimal policies for all convex combinations of the rewards/objectives

Convex Coverage Set (CCS)



Optimal policy w.r.t. *any* convex combination of rewards is in the **CCS**!

Generalized Policy Improvement (GPI)

GPI is the computation of a policy π' that improves over a **set** *of policies* $\pi \in \Pi$

$$\boxed{\pi^{GPI}(s; w) = \arg\max_{a \in \mathcal{A}} \max_{\pi \in \Pi} q_w^{\pi}(s, a)}$$

GPI Theorem

$$q_w^{GPI}(s,a) \geq \max_{\pi \in \Pi} q_w^{\pi}(s,a)$$

for any $w \in \mathcal{W}$

Main Contributions

We introduce two Generalized Policy Improvement (GPI)-based prioritization schemes that improve sample-efficiency in MORL:

GPI Linear Support (GPI-LS)

- Identify the most promising preferences/objectives to train on
- Guaranteed convergence to optimal (or ϵ-optimal) solutions

GPI-Prioritized Dyna (GPI-PD)

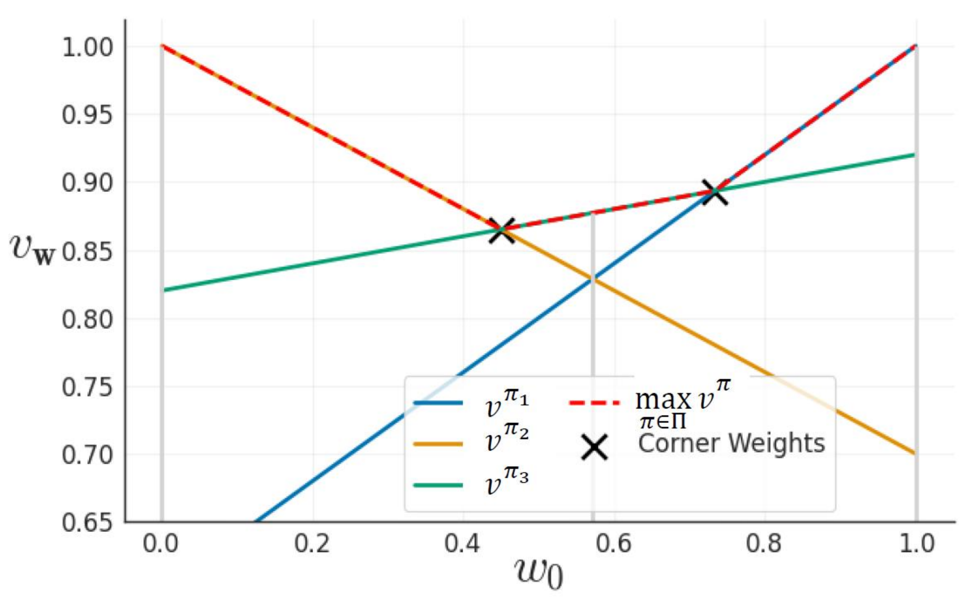
- Identify relevant previous experiences when learning a new policy
- First model-based MORL method for continuous states/actions

GPI Linear Support (GPI-LS)

- Iteratively learns a policy set Π whose value vectors $\mathcal{V}$ approx. the CCS
- **Key idea:**

Consider only **corner weights,** and prioritize them based on the **performance improvement given by GPI:**

$$\boxed{\arg\max_{\mathbf{w} \in \mathcal{W}_{\text{corner}}} \left(v_{\mathbf{w}}^{\text{GPI}} - \max_{\pi \in \Pi} v_{\mathbf{w}}^{\pi}\right)}$$



- Maximum improvement is guaranteed to be in one of the corner weights (Thm. 3.2)
- Selects the corner weight with higher GPI priority
- Learns an improved policy for the selected reward weights

GPI-LS is guaranteed to:

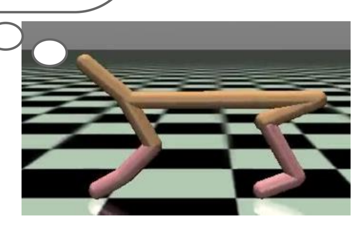
- Identify a CCS in a finite number of iterations
- Identify an ϵ-CCS in case the learning algorithm is ϵ-optimal

GPI Prioritized Dyna (GPI-PD)

Policies learned via a Dyna-style approach using a learned dynamics model

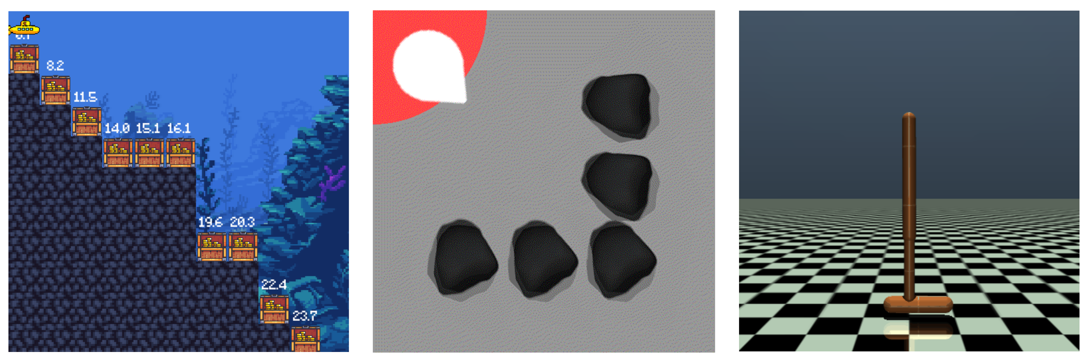
for *H Dyna steps* **do** ▷ GPI-Prioritized Dyna
  Sample $S \sim \mathcal{B}$ according to $P_{\mathbf{w}_t}$ (Eq. (10))
  $A \leftarrow \pi^{\text{GPI}}(S; \mathbf{w}_t)$; $(\hat{S}', \hat{\mathbf{R}}) \sim p_{\varphi}(\cdot | S, A)$
  Add $(S, A, \hat{\mathbf{R}}, \hat{S}')$ to $\mathcal{B}_{\text{model}}$

$$\boxed{P_{\mathbf{w}}(s,a) \propto q_{\mathbf{w}}^{\text{GPI}}(s,a) - q_{\mathbf{w}}^{\pi}(s,a)}$$



Prioritizes experiences for which GPI results in larger performance improvements

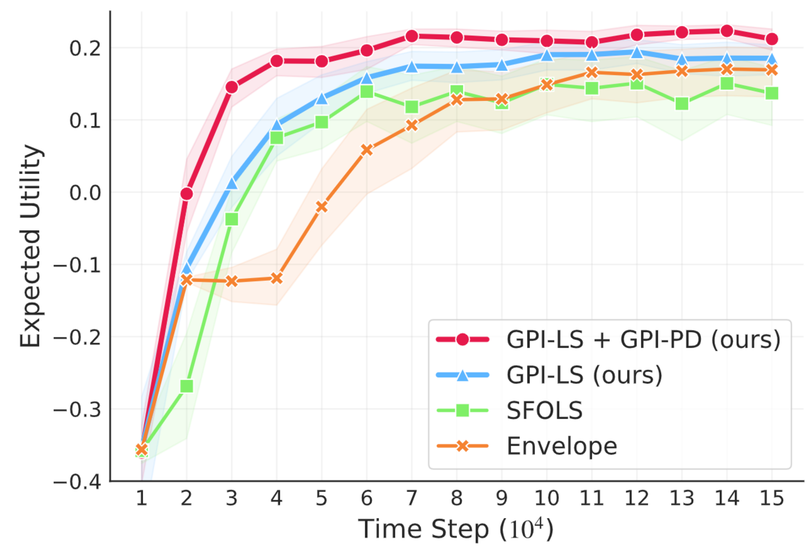
Experiments



Deep Sea Treasure, Minecart, and MO-Hopper

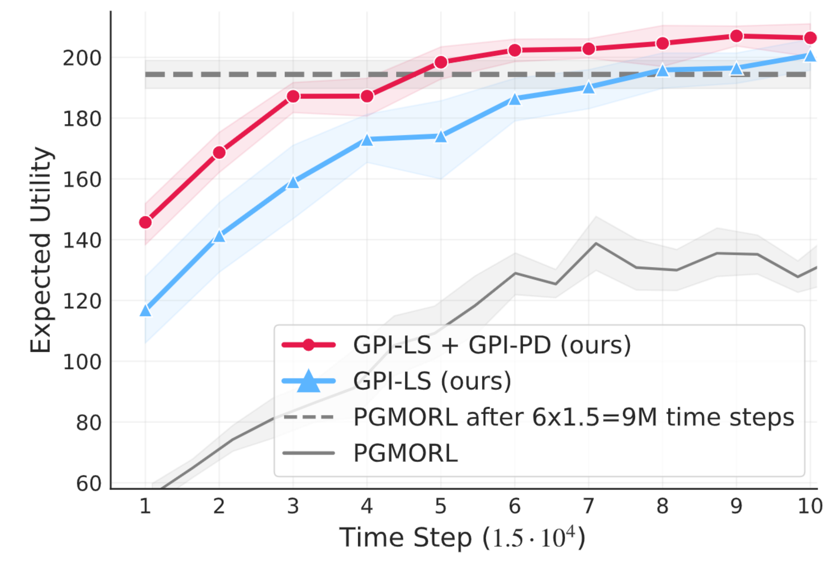
discrete and continuous state and action spaces

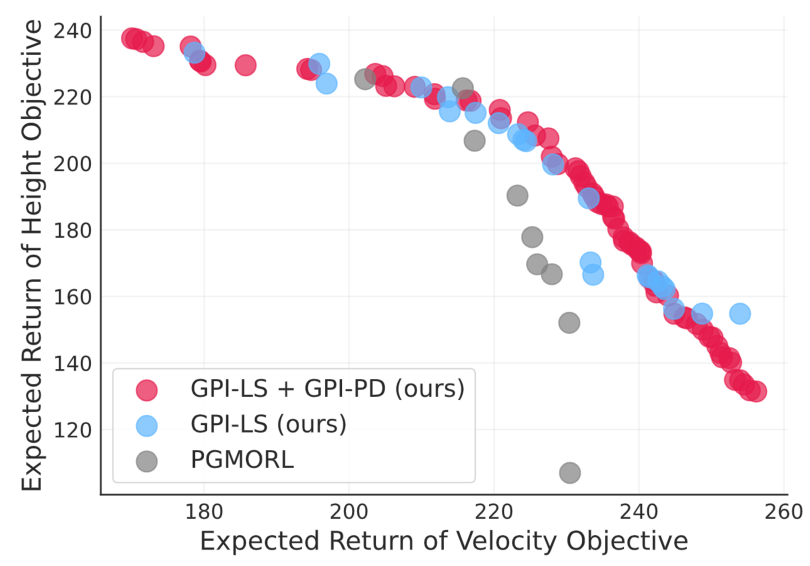
Evaluation metric: *Expected Utility* (EU) $\text{EU}(\Pi) = \mathbb{E}_{\mathbf{w} \sim \mathcal{W}}[\max_{\pi \in \Pi} v_{\mathbf{w}}^{\pi}]$



Minecart

- GPI-LS and GPI-LS+GPI-PD consistently identify (near) optimal solutions
- Expected utility strictly dominates that of competitors





MO-Hopper

- Our methods achieved higher expected utility and converged to better solutions
- Require *ten times less* environment interactions compared to SOTA method
- The Pareto Front identified by our methods cover better the space of possible trade-offs between objectives

Discussion & Conclusion

- We introduced two principled prioritization methods
  - Monotonically improve the quality of the set of policies
  - Convergence guarantees to (near) optimal solutions
- GPI-PD is the first model-based MORL algorithm for continuous states
- Outperforms state-of-the-art MORL algorithms in challenging tasks
  - Significantly improves sample-efficiency