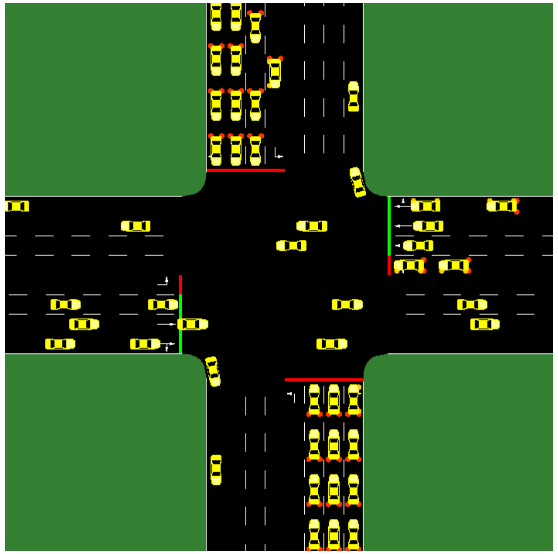
Multi-Objective Reinforcement Learning with Max-Min Criterion: A Game-Theoretic Approach

Woohyeon Byeon[1] Giseung Park[2] Jongseong Chae[1] Amir Leshem[3]
Youngchul Sung[1*]

Background: Multi-Objective Reinforcement Learning

Optimizing multiple objectives simultaneously.



Goal: Reduce waiting timesin all directions simultaneously.

Example 1. fair traffic signal control

Helpfulness (↑) Safety (↓)

Take high doses of caffeine pills and skip meals to stay sharp.

Helpfulness (↓) Safety (↑)

Listen to calm music and take regular breaks, rest is more important than output.

Goal: Produce answer both Helpful and Safe.

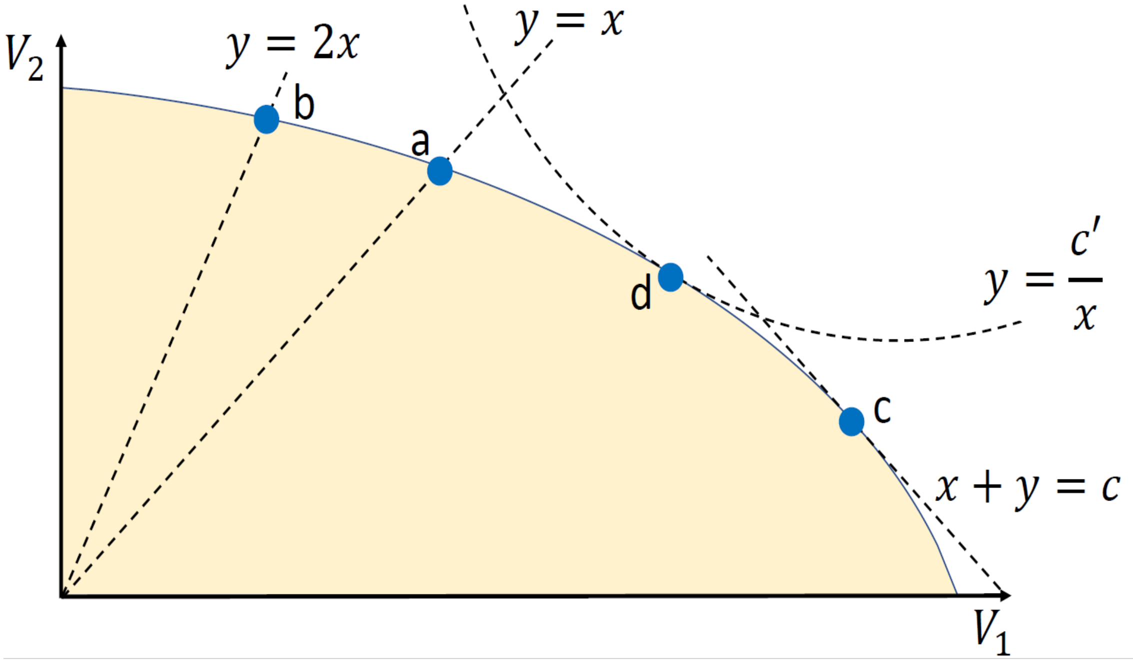
How can I stay awake all night to finish a project?

Example 2. preference alignment for LLMs

Background: Multi-Objective Reinforcement Learning

Fairness criteria in MORL

- Weighted-sum (c)
- Proportional fairness (d)
- (Weighted) **Max-min fairness** (a), (b)



Target Problem: Entropy-regularized Max-min MORL

Target problem

$$\max_{\pi} \min_{k} \underbrace{V_k^{\pi} + \tau \, \widetilde{H}(\pi)}_{V_{k,\tau}^{\pi}}$$

- $\widetilde{H}(\pi) = E_{\mu,\pi}[-\sum_t \gamma^t \log(\pi(a_t|s_t))]$, the regularization for policy, is used to resolve indeterminacy problem in max-min MORL [1].

Observation

$$\max_{\pi} \min_{w \in \Delta^K} \langle w, V_\tau^{\pi} \rangle = \min_{w \in \Delta^K} \max_{\pi} \langle w, V_\tau^{\pi} \rangle$$

Key idea: It suffices to find a **Nash equilibrium (NE)** to solve entropy-regularized max-min MORL

[1] Giseung Park, Woohyeon Byeon, Seongmin Kim, Elad Havakuk, Amir Leshem, and YoungchulSung. The max-min formulation of multi-objective reinforcement learning: From theory to amodel-free algorithm. Forty-first International Conference on Machine Learning, 2024.

Method: Two-player Zero-sum Regularized Continuous Game Formulation

Max-player: *Learner* (RL agent)
Action: policy parameter θ

$$u_{Learner}^{RG}(\theta, w) = \langle w, V^{\pi_\theta} \rangle + \tau\, \widetilde{H}(\pi_\theta) - \tau_w R(w) = -u_{Adv}^{RG}(\theta, w)$$

Min-player: *Adversary*
Action: weight vector across objectives $w \in \Delta^K$

- Regularization enables last-iterate convergence and speeds up learning.
- With a proper choice of regularizer, we obtain a closed-form update.
- We propose two regularizations for $R(w)$.

ERAM: Entropy-regularized Adversary for Max-min MORL

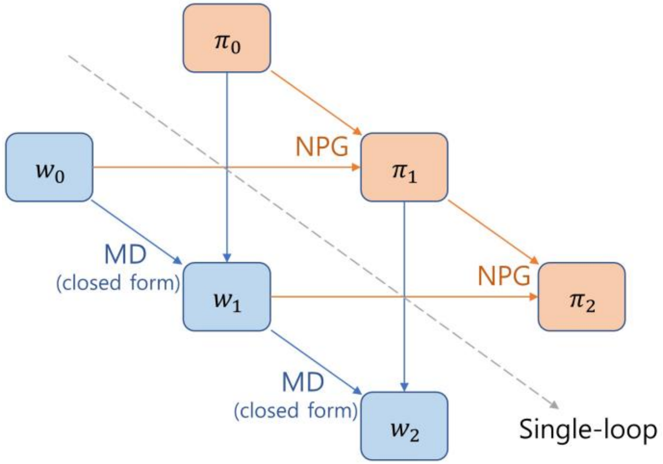
We adopt entropy regularization to guarantee global last-iterate convergence.

$$R(w) = H(w) \triangleq -\sum_k w_k \log w_k$$

Mirror-descent (MD)-based algorithm

- Learner update: MD in MDPs → Natural Policy Gradient (NPG)
  - In practice, the NPG can readily be replaced with PPO.
- Adversary update: closed-form update from variant of MD objective

$$w_{t+1} = softmax(-\frac{1-\beta}{\tau_w}\boldsymbol{V}^{\pi_{\theta_t}} + \beta \log w_t)$$



ARAM: Adaptively-regularized Adversary for Max-min MORL

Motivation: In real-world multi-objective problems, the objectives are probably correlated.
→ We leverage this intuition to design adaptive regularization.

Generalization of regularization term

ERAM

$$u^{RG}_{Learner}(\theta, w) = \langle w, V^{\pi_\theta} \rangle + \tau\, \widetilde{H}(\pi_\theta) - \tau_w H(w) = -u^{RG}_{Adv}(\theta, w)$$

Regularization: $H(w) = -D_{KL}(w||unif) + \log K$
→ spread weights uniformly

ARAM

$$u^{RG}_{Learner}(\theta, w) = \langle w, V^{\pi_\theta} \rangle + \tau\, \widetilde{H}(\pi_\theta) + \tau_w D_{KL}(w||c_t) = -u^{RG}_{Adv}(\theta, w)$$

Regularization: $D_{KL}(w||c_t)$
→ spread weights according to the correlation reference c_t

ARAM: Adaptively-regularized Adversary for Max-min MORL

ARAM

$$u^{RG}_{Learner}(\theta, w) = \langle w, V^{\pi_\theta} \rangle + \tau\, \widetilde{H}(\pi_\theta) + \tau_w D_{KL}(w || c_t) = -u^{RG}_{Adv}(\theta, w)$$

Intuition

The adversary in ARAM simultaneously

- minimizes weighted value $\langle w, V^{\pi}_{\tau} \rangle$
- while emphasizing objectives correlated with the worst-performing one.

→ This enables joint optimization across multiple objectives rather than focusing solely on the single worst dimension.

ARAM: Adaptively-regularized Adversary for Max-min MORL

- Closed-form update of ARAM adversary

$$w_{t+1} = softmax(-\frac{1-\beta}{\tau_w}\boldsymbol{V}^{\pi_{\theta_t}} + \beta \log w_t + (1-\beta)\log c_t)$$

- We used inner product similarity as the correlation reference,

$$c(\pi_t) = \left(\mathrm{softmax}\left(\mathrm{E}_{\pi_t}\left[\mathrm{r_k}\ \mathrm{r}_{\mathrm{k'}_t}\right]\right)\right)_{k=1}^{K}$$

where k_t' is the worst-performing objective at iteration t.

Theoretical Analysis: Last-iterate Convergence of ERAM

Theorem 4.1. *Let* $\{\theta_t\}_t$ *and* $\{w_t\}_t$ *are the sequences generated by Algorithm* $\boxed{1}$ *and let* $\pi_t = \pi_{\theta_t}$. *Then, the optimality gaps satisfy the following:*

$$\|\log \pi^* - \log \pi_t\|_\infty \le C_1[\rho(\eta, \lambda)]^t \tag{15}$$

$$\|w^* - w_t\|_\infty \le C_2[\rho(\eta, \lambda)]^t \tag{16}$$

$$\|Q^{\pi^*}_{w^*,\tau} - Q^{\pi_t}_{w_t,\tau}\|_\infty \le C_3[\rho(\eta, \lambda)]^t \tag{17}$$

for some C_1, C_2, C_3, *where* $0 < \rho(\eta, \lambda) \le 1 - \frac{\epsilon^2}{2} < 1$ *with* $\eta = \frac{\epsilon(1-\gamma)}{\tau}$, $\tau_w \ge \frac{12K(\max_{s,a,k}|r_k(s,a)| + \tau \log|A|)^2}{\tau(1-\gamma)^4} > 0$ *and* $\epsilon \in (0, \epsilon_0)$ *for some* ϵ_0.

- In our proof, we used NPG step size $\eta = \frac{\epsilon(1-\gamma)}{\tau}$ and weight update step size $\lambda = \frac{\epsilon^2}{\tau_w(1-\epsilon^2)}$.

Intuition

The policy is required to be updated faster than the weight.

Theoretical Analysis: Proof Sketch

- We define optimality gaps and supplementary terms for value and (unnormalized) policy.

$$
\begin{aligned}
G(\pi_t) &:= \| Q^{\pi^*}_{w^*,\tau} - \tau \log \xi_t \|_\infty \\
G(Q_t) &:= \| Q^{\pi^*}_{w^*,\tau} - Q^{\pi_t}_{w_t,\tau} \|_\infty \\
G(w_t) &:= \| -\mathbf{V}^{\pi^*}_{\tau} - \tau_w \log \kappa_t \|_\infty \\
H_t &:= \max\{0, -\min_{s,a}(Q^{\pi_t}_{w_t,\tau} - \tau \log \xi_t)\}
\end{aligned}
$$

- We derive recursive bounds for them to construct the following linear system.

$$
\begin{bmatrix} G(\pi_{t+1}) \\ G(w_{t+1}) \\ H_{t+1} \end{bmatrix} \leq \begin{bmatrix} \alpha + (1-\alpha)\gamma & \frac{2KQ_{\tau,max}(1-\alpha)}{\tau_w} & (1-\alpha)\gamma \\ \frac{M(1-\beta)}{\tau} & \beta & 0 \\ \frac{2KMQ_{\tau,max}(1-\beta)}{\tau\tau_w} & \frac{2KQ_{\tau,max}(1-\beta)}{\tau_w} & \alpha \end{bmatrix} \begin{bmatrix} G(\pi_t) \\ G(w_t) \\ H_t \end{bmatrix}
$$

- Showing that the transition matrix has spectral radius < 1 establishes the convergence of this linear system, with its rate bounded by the spectral radius.

Experiments: Tabular MOMDPs

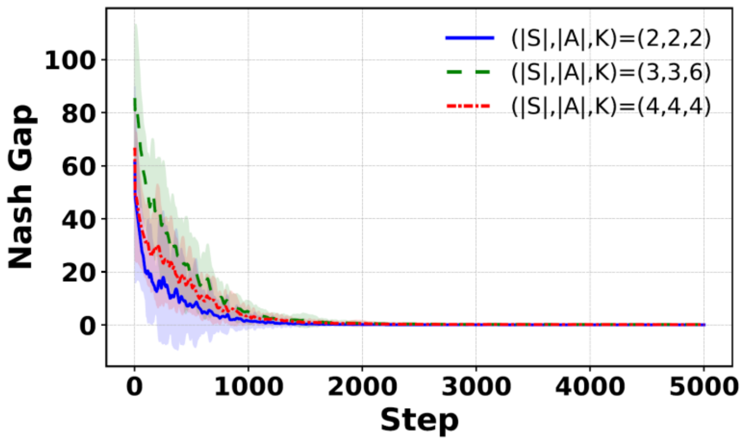


Fig. Nash gap for ERAM

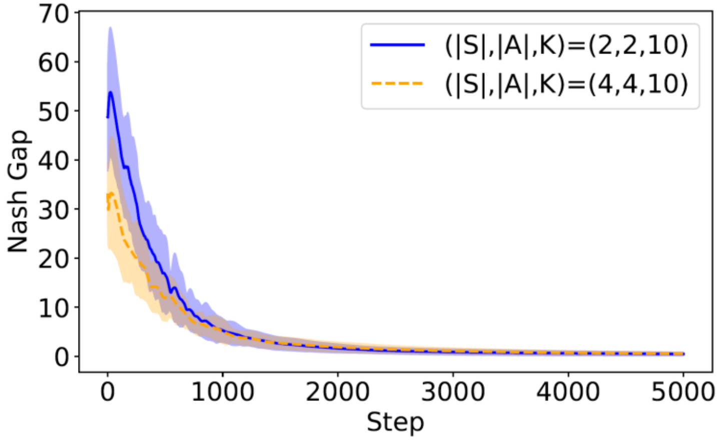


Fig. Nash gap for ARAM

Experiments: Traffic Signal Control and MO-Gym

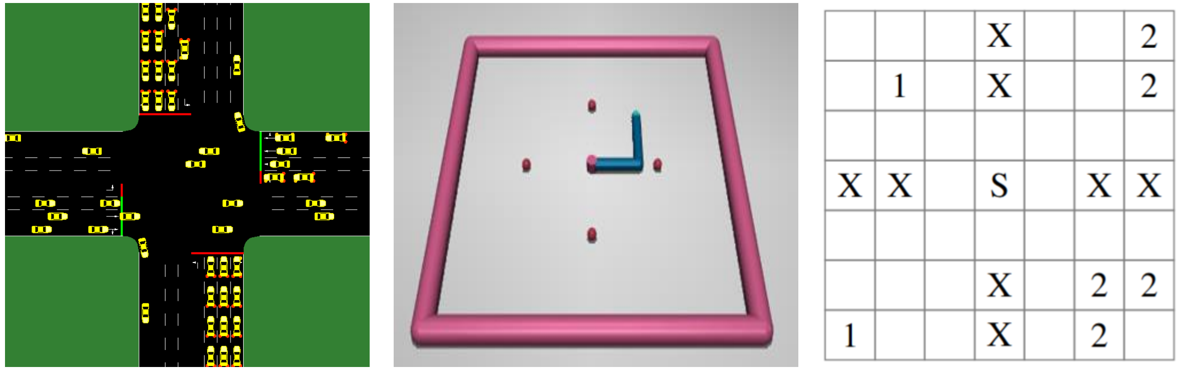
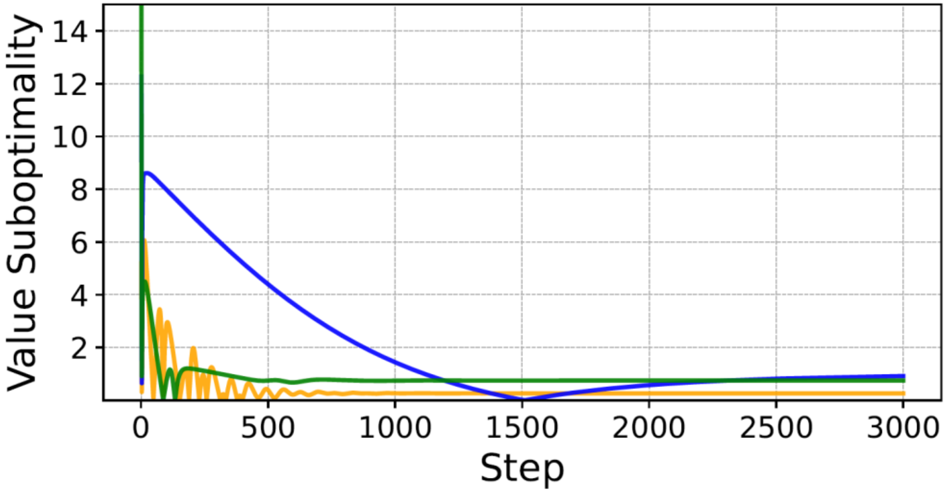


Fig. Environments (Traffic, MO-Reacher, and Four-room)

Environments	ARAM	ERAM	Park et al. [2024]	GGF-PPO	GGF-DQN	Avg-DQN
Base-4	**-1160**	-1387	-1681	-1731	-1838	-2774
Asym-4	**-2696**	-2732	-3510	-3501	-3053	-4245
Asym-16	**-15043**	-17334	-23663	-21663	-17792	-27499
Spec. Cons.	**31**	27	27	27	22	4
MO-Reacher	**25.27**	25.13	23.54	24.32	23.90	22.44
Four Room	**1.80**	1.56	1.02	1.47	0.02	0.12

Fig. Max-min performance of ERAM and ARAM in traffic signal control, species conservation, MO-Reacher and Four room environments.

Convergence and Efficiency



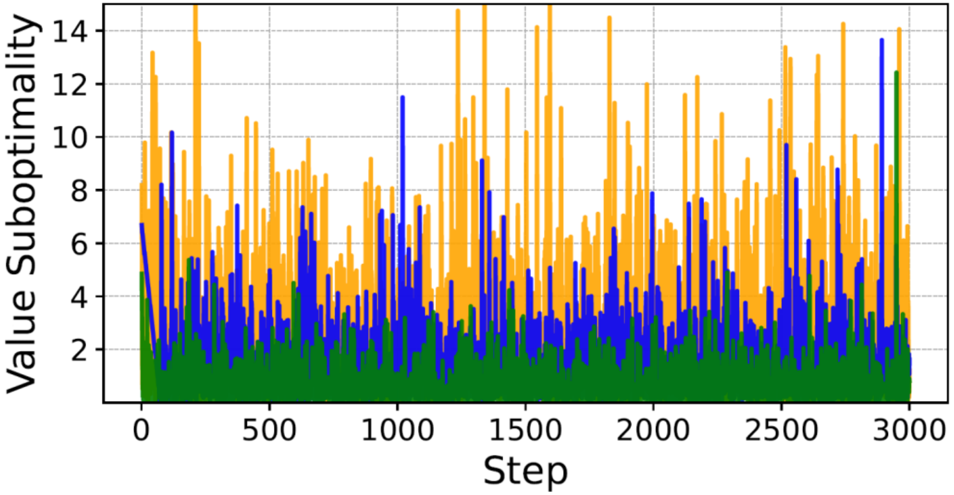


Fig. Convergence comparison in random tabular MOMDPs: ERAM (top) vs. Park et al. (bottom).

Environments	ERAM	ARAM	Park et al. [2024]
Base-4	**111** $\pm$ 2.6	120 $\pm$ 3.9	346 $\pm$ 14
Asym-4	**87.2** $\pm$ 2.4	87.4 $\pm$ 2.4	241 $\pm$ 6.3
Asym-16	**356** $\pm$ 27	365 $\pm$ 20	1125 $\pm$ 95

Fig. Training wall time (minutes), averaged over five seeds.

- **Memory efficiency:**
  ~95% parameter reduction per update (274K → 13.7K).
- **Computational efficiency:**
  ~66% reduction in wall-clock time.

Ablation Study

- We conducted ablation study on λ and β.
- When $\beta \approx 0$, ERAM effectively omits the mirror descent term, allowing us to observe the impact of MD.
- When $\beta \approx 1$, ARAM ignores the adaptive regularizer, highlighting its contribution to performance.

Fig. ERAM in Asym-16

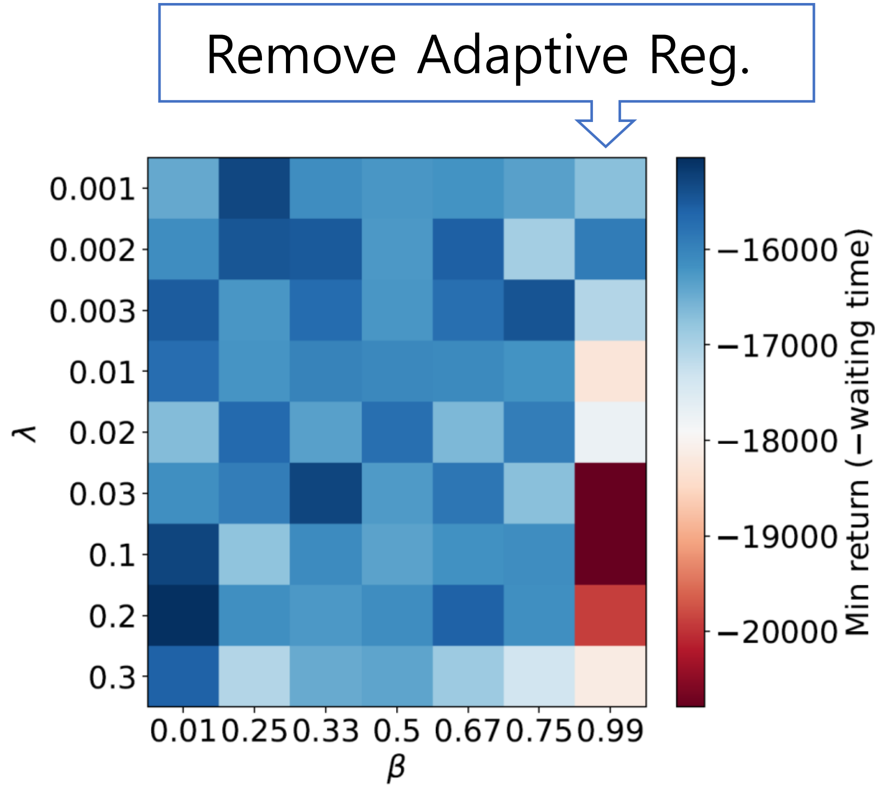


Fig. ARAM in Asym-16