Maximum Entropy RL and Policy Gradients: Why and What to Explore?

Adrien Bolland
adrien.bolland@uliege.be
February 6, 2026

Sequential decision-making

Taking actions based on states in an environment where time evolves.

Images generated using ChatGPT.

Sequential decision-making

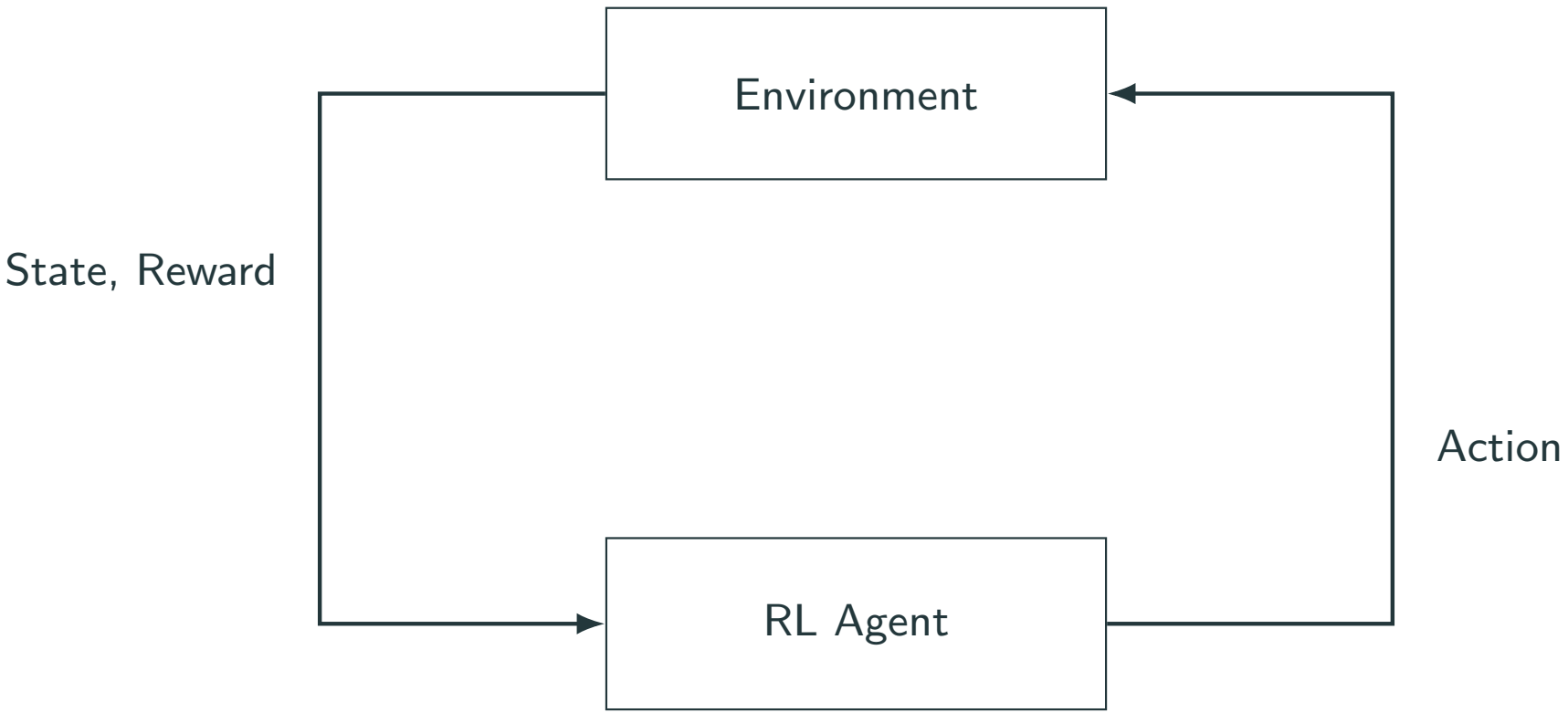
Taking actions based on states in an environment where time evolves.

The quality of states and actions is measured using rewards.

Images generated using ChatGPT.

Reinforcement learning

Agents learn from experience to maximize the expected sum of rewards.



Notations

Some RL notations:

- $s_t \in \mathcal{S}$ for the states,
- $a_t \in \mathcal{A}$ for the actions,
- $p_0(s_0)$ for the initial state distribution,
- $p(s_{t+1}|s_t, a_t)$ for the transition distribution,
- $R(s_t, a_t)$ for the reward function,
- γ for the discount factor,
- $\pi(a_t|s_t)$ for stochastic policies,
- $d^{\pi_\theta,\gamma}(\bar{s})$ for the state occupancy.

Policy gradient algorithms – Optimization problem

Definition (Optimal Policies)

Agents should act according to a policy π^* that maximizes the expected return

$$J(\pi) = \mathbb{E}^{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right] .$$

Policy gradient algorithms – Recipe

In policy gradient algorithms:

1. The agent has a direct parameterized representation of the policy.

$$\pi_\theta(a_t|s_t) = \mathcal{N}(a_t|\mu_\theta(s_t), \Sigma_\theta(s_t))$$
$$\pi_\theta(a_t|s_t) \propto \exp\left(\phi_\theta(s_t, a_t)\right) \ .$$

Policy gradient algorithms – Recipe

In policy gradient algorithms:

1. The agent has a direct parameterized representation of the policy.

$$\pi_\theta(a_t|s_t) = \mathcal{N}(a_t|\mu_\theta(s_t), \Sigma_\theta(s_t))$$
$$\pi_\theta(a_t|s_t) \propto \exp\left(\phi_\theta(s_t, a_t)\right) \ .$$

2. The agent learns by stochastic gradient ascent.

$$\theta \leftarrow \theta + \alpha\hat{d} \qquad \hat{d} \approx \nabla_\theta J(\pi_\theta)$$
$$\hat{d} \approx \arg\min_d \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t (d \cdot \nabla_\theta \log \pi_\theta(a_t|s_t) - Q^{\pi_\theta}(s_t, a_t))^2\right] \ .$$

Policy gradient algorithms – Exploration

The policy should remain sufficiently stochastic during the learning period to ensure sufficient exploration of the environment.

Policy gradient algorithms – Exploration

The policy should remain sufficiently stochastic during the learning period to ensure sufficient exploration of the environment.

Exploration-exploitation dilemma.

- An RL agent collects information and exploits information.
- Some actions are deliberately suboptimal.
- Necessary for some algorithms and an efficiency criterion.

Numerical optimization perspective

The exploration-exploitation perspective falls short for studying policy gradients.

Numerical optimization perspective

The exploration-exploitation perspective falls short for studying policy gradients.

Forgetting about RL.

- The problem consists in maximizing an objective function.
- The optimization is performed by SGA.
- Algorithms converge at a certain rate.
- No exploration arguments involved.

Numerical optimization perspective

The exploration-exploitation perspective falls short for studying policy gradients.

Forgetting about RL.

- The problem consists in maximizing an objective function.
- The optimization is performed by SGA.
- Algorithms converge at a certain rate.
- No exploration arguments involved.

Understand exploration using numerical optimization arguments.

Maximum entropy RL

Learning objective

Policy gradient algorithms optimize the learning objective $L(\theta)$ by SGA:

$$L(\theta) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty}\gamma^t\left(R(s_t,a_t)+\lambda R^{int}(s_t,a_t)\right)\right] = J(\pi_\theta)+\lambda J^{int}(\pi_\theta)\,.$$

Maximum entropy RL

Learning objective

Policy gradient algorithms optimize the learning objective $L(\theta)$ by SGA:

$$L(\theta) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty}\gamma^t\left(R(s_t,a_t) + \lambda R^{int}(s_t,a_t)\right)\right] = J(\pi_\theta) + \lambda J^{int}(\pi_\theta)\,.$$

The intrinsic return or intrinsic reward corresponds to an entropy function.

$$R^{int,s}(s,a) = -\log d^{\pi_\theta,\gamma}(s) \qquad J^{int,s}(\pi_\theta) = \mathcal{H}_s\left[d^{\pi_\theta,\gamma}(s)\right]$$
$$R^{int,a}(s,a) = -\log \pi_\theta(a|s) \qquad J^{int,a}(\pi_\theta) = \mathbb{E}^{\pi_\theta}\left[\mathcal{H}_a\left[\pi_\theta(a|s)\right]\right]\,.$$

Abstract

- Why does maximum entropy RL help with convergence?
- A new objective: maximize entropy of future states and actions?

Study of the maximum entropy RL objective

Shape of the learning objective

Let us assume unbiased gradient estimates of the learning objective.

$$\theta \leftarrow \theta + \alpha \hat{d} \qquad \mathbb{E}\left[\hat{d}\right] = \nabla_\theta L(\theta)\,.$$

If the function is concave, SGA converges to the global maximum.

Shape of the learning objective

Let us assume unbiased gradient estimates of the learning objective.

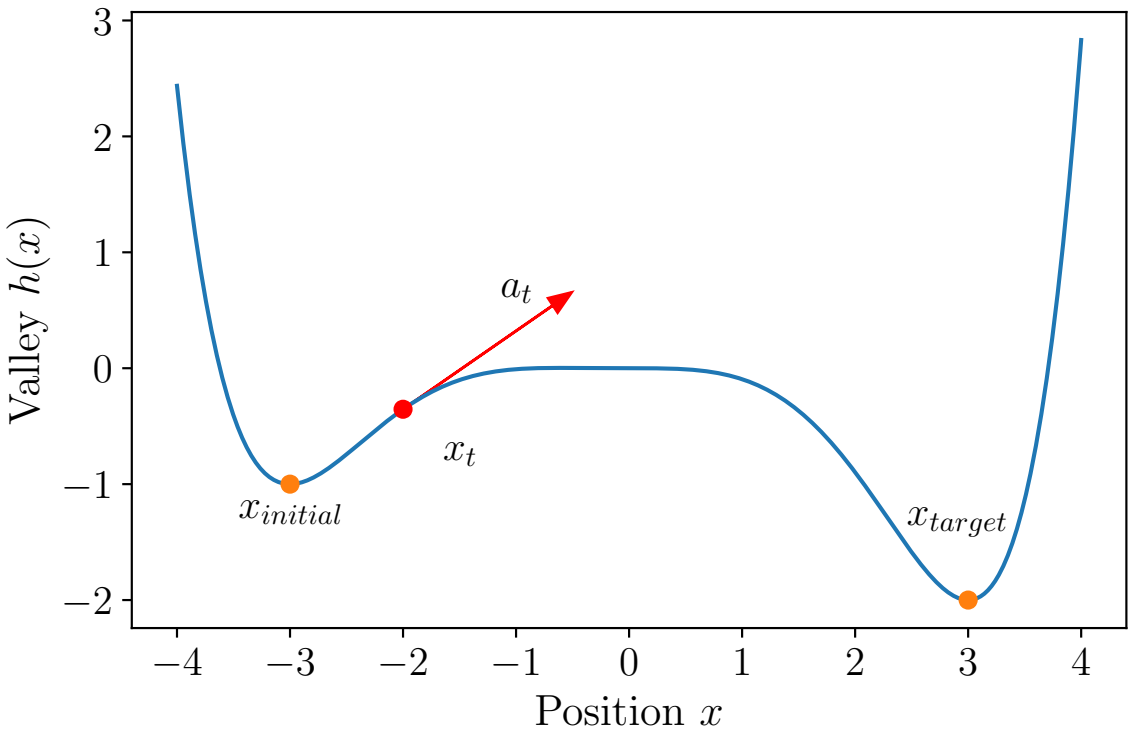
$$\theta \leftarrow \theta + \alpha \hat{d} \qquad \mathbb{E}\left[\hat{d}\right] = \nabla_\theta L(\theta)\,.$$

If the function is concave, SGA converges to the global maximum.

Let us look at the objective function depending on the factor λ.

Illustration

We consider a car moving in a valley, and denote by x its position and by v its speed. The car starts at $x_{initial}$ and perceives rewards proportional to the depth in the valley; an optimal sequence of actions moves the car to x_{target}.



Learning objective in the hill environment

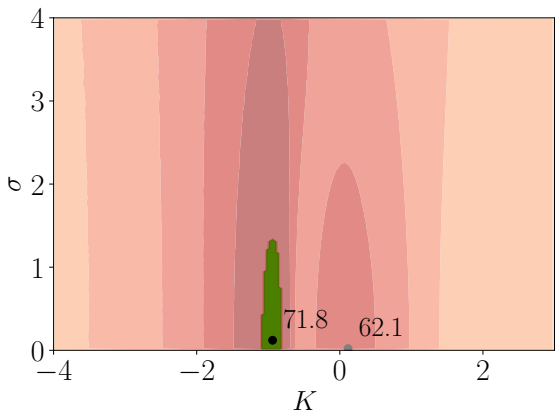
We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

$$L(K, \sigma) = \mathbb{E}^{\pi_\theta} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_s R^{int,s}(s_t, a_t) \right) \right] .$$

Learning objective in the hill environment

We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

$$L(K, \sigma) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_s R^{int,s}(s_t, a_t)\right)\right] .$$

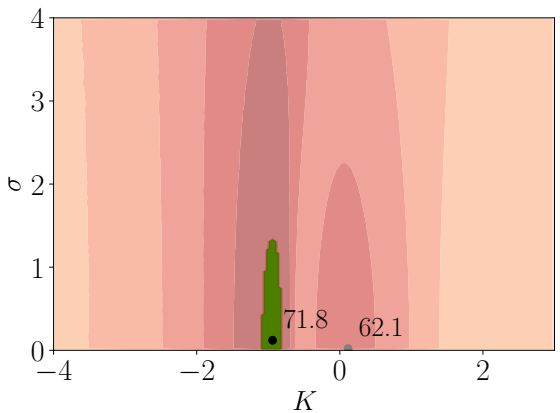


(a) $\lambda_s = 0.05$

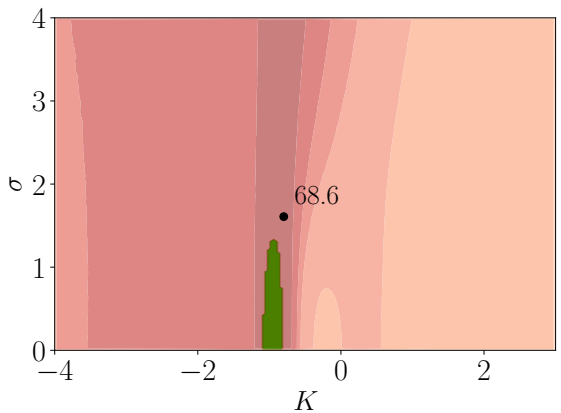
Learning objective in the hill environment

We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

$$L(K, \sigma) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_s R^{int,s}(s_t, a_t)\right)\right] .$$



(a) $\lambda_s = 0.05$

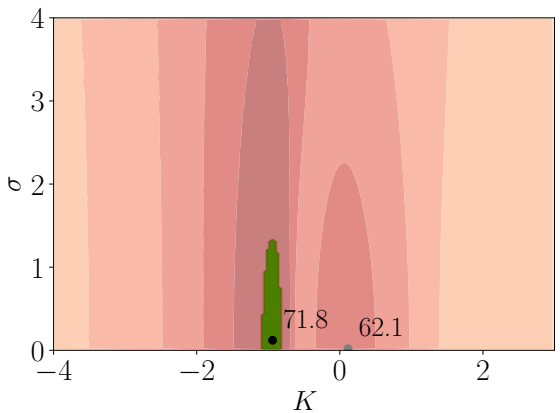


(c) $\lambda_s = 1$

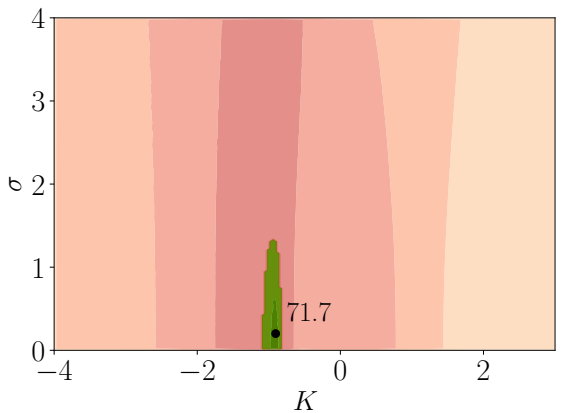
Learning objective in the hill environment

We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

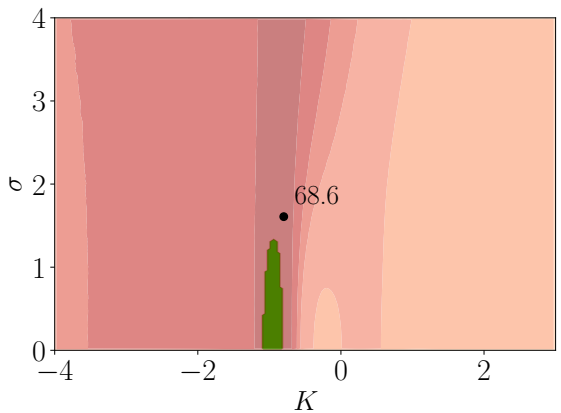
$$L(K, \sigma) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_s R^{int,s}(s_t, a_t)\right)\right] .$$



(a) $\lambda_s = 0.05$



(b) $\lambda_s = 0.1$



(c) $\lambda_s = 1$

Learning objective in the hill environment

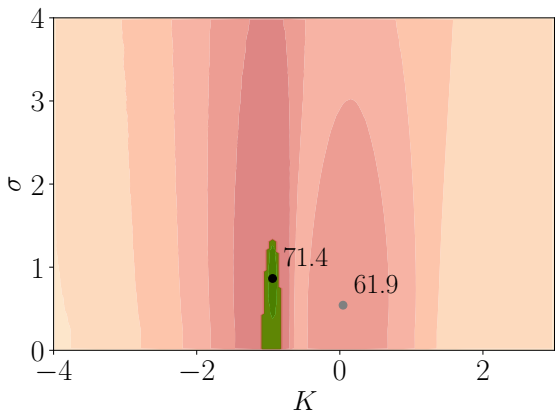
We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

$$L(K, \sigma) = \mathbb{E}^{\pi_\theta} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_a R^{int,a}(s_t, a_t) \right) \right] .$$

Learning objective in the hill environment

We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

$$L(K, \sigma) = \mathbb{E}^{\pi_\theta} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_a R^{int,a}(s_t, a_t) \right) \right] .$$

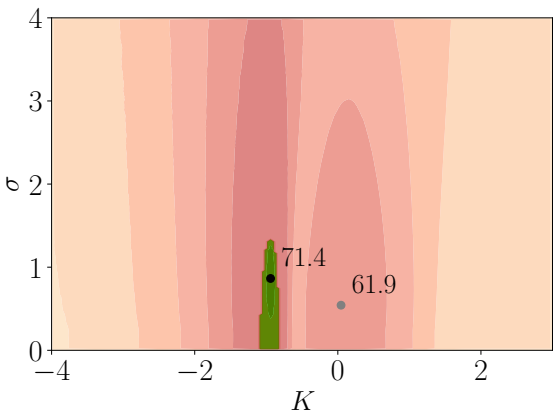


(a) $\lambda_a = 0.01$

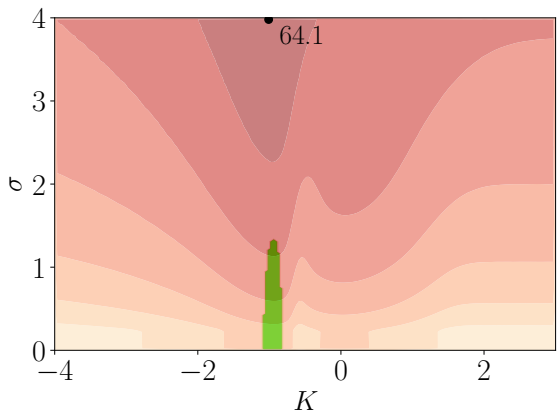
Learning objective in the hill environment

We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

$$L(K, \sigma) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_a R^{int,a}(s_t, a_t)\right)\right] .$$



(a) $\lambda_a = 0.01$

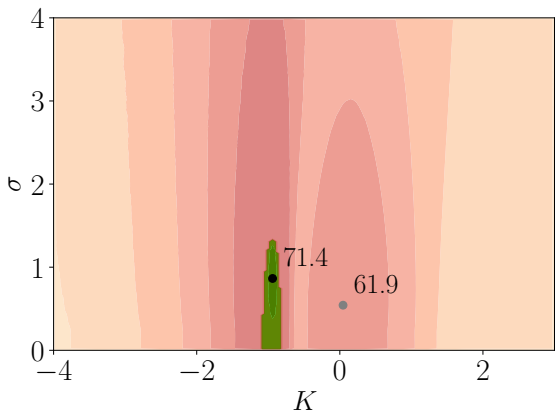


(c) $\lambda_a = 0.5$

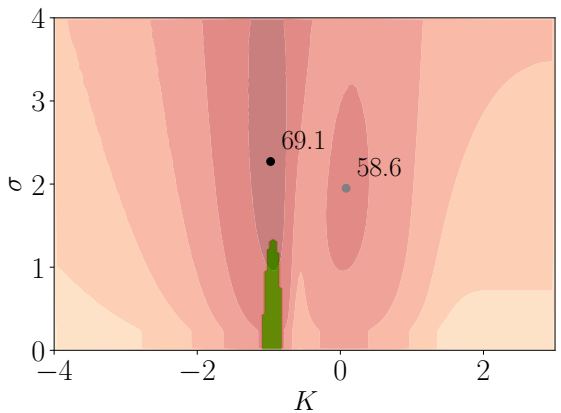
Learning objective in the hill environment

We optimize the policy $\pi_{K,\sigma}(a|s) = \mathcal{N}(a|K \times (x - x_{target}), \sigma)$ with the objective

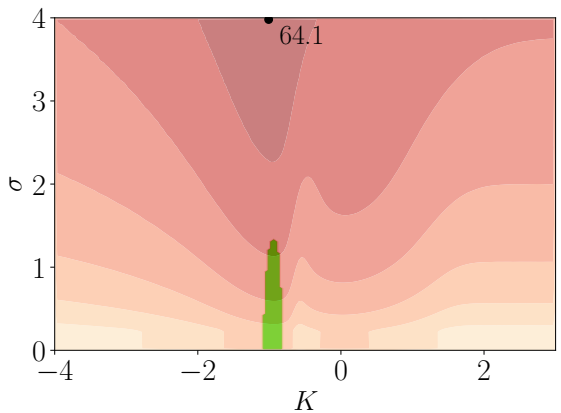
$$L(K, \sigma) = \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t, a_t) + \lambda_a R^{int,a}(s_t, a_t)\right)\right] .$$



(a) $\lambda_a = 0.01$



(b) $\lambda_a = 0.1$



(c) $\lambda_a = 0.5$

Objective function

Studies have related exploration to smoothing objectives or robust optimization.

[1] Ahmed, Z., Le Roux, N., Norouzi, M., & Schuurmans, D. (2019, May). Understanding the impact of entropy on policy optimization. In International Conference on Machine Learning (pp. 151-160). PMLR.

[2] Husain, H., Ciosek, K., & Tomioka, R. (2021, March). Regularized policies are reward robust. In International Conference on Artificial Intelligence and Statistics (pp. 64-72). PMLR.

[3] Brekelmans, R., Genewein, T., Grau-Moya, J., Delétang, G., Kunesch, M., Legg, S., & Ortega, P. (2022). Your policy regularizer is secretly an adversary. arXiv preprint arXiv:2203.12592.

[4] Bolland, A., Lambrechts, G., & Ernst, D. (2025). Behind the myth of exploration in policy gradients. 18th European Workshop on Reinforcement Learning.

[5] Ashlag, Y., Koren, U., Mutti, M., Derman, E., Bacon, P. L., & Mannor, S. (2025). State entropy regularization for robust reinforcement learning. arXiv preprint arXiv:2506.07085.

Exploration and convergence results

Convergence of SGA

Let us consider:

- A log-linear policy.
- Bounded approximation and estimation errors.
- Initial state-action pairs are generated from $\nu(s, a)$.

Does NPG converge to the optimum?

---

[1] Agarwal, A., Kakade, S. M., Lee, J. D., & Mahajan, G. (2021). On the theory of policy gradient methods: Optimality, approximation, and distribution shift. Journal of Machine Learning Research, 22(98), 1-76.

Convergence of SGA

Let us consider:

- A log-linear policy.
- Bounded approximation and estimation errors.
- Initial state-action pairs are generated from $\nu(s, a)$.

Does NPG converge to the optimum?

$$\mathbb{E}\left[\min_{t \leq T} J(\pi^*) - J(\pi_{\theta_t})\right] \leq O\left(\frac{1}{\sqrt{T}} + \varepsilon\right) .$$

Typically ε is bounded under conditions on the state-action distribution.

---

[1] Agarwal, A., Kakade, S. M., Lee, J. D., & Mahajan, G. (2021). On the theory of policy gradient methods: Optimality, approximation, and distribution shift. Journal of Machine Learning Research, 22(98), 1-76.

Relationship with maximum entropy exploration

The state-action distribution from $\nu(s, a)$ should cover $d^{\pi^*}(s)\mathcal{U}(a)$.

Relationship with maximum entropy exploration

The state-action distribution from $\nu(s, a)$ should cover $d^{\pi^*}(s)\mathcal{U}(a)$.

Most algorithms do not control the initial state distribution.

Maximum entropy RL encourages broader state and action coverage.

Exploring future states and actions

Maximum entropy RL

Learning objective

A general (or generalized) maximum entropy RL objective is:

$$
\begin{aligned}
L(\theta) &= J(\pi_\theta) + \lambda \mathcal{H}_{s,a}\left[d^{\pi_\theta,\gamma}(s,a)\right] \\
&= \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t,a_t) - \lambda \log\left(d^{\pi_\theta,\gamma}(s_t)\pi_\theta(a_t|s_t)\right)\right)\right] .
\end{aligned}
$$

[1] Hazan, E., Kakade, S., Singh, K., & Van Soest, A. (2019, May). Provably efficient maximum entropy exploration. In International Conference on Machine Learning (pp. 2681-2691). PMLR.

[2] Mutti, M., & Restelli, M. (2020, April). An intrinsically-motivated approach for learning highly exploring and fast mixing policies. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 04, pp. 5232-5239).

[3] Liu, H., & Abbeel, P. (2021). Behavior from the void: Unsupervised active pre-training. Advances in Neural Information Processing Systems, 34, 18459-18473.

[4] Islam, R., Seraj, R., Bacon, P. L., & Precup, D. (2019). Entropy regularization with discounted future state distribution in policy gradient methods. arXiv preprint arXiv:1912.05104.

[5] Guo, Z. D., Azar, M. G., Saade, A., Thakoor, S., Piot, B., Pires, B. A., ... & Munos, R. (2021). Geometric entropic exploration. arXiv preprint arXiv:2101.02055.

[6] Lee, L., Eysenbach, B., Parisotto, E., Xing, E., Levine, S., & Salakhutdinov, R. (2020). Efficient exploration via state marginal matching.

Maximum entropy RL

Learning objective

A general (or generalized) maximum entropy RL objective is:

$$
\begin{aligned}
L(\theta) &= J(\pi_\theta) + \lambda \mathcal{H}_{s,a}\left[d^{\pi_\theta,\gamma}(s,a)\right] \\
&= \mathbb{E}^{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \left(R(s_t,a_t) - \lambda \log\left(d^{\pi_\theta,\gamma}(s_t)\pi_\theta(a_t|s_t)\right)\right)\right] .
\end{aligned}
$$

The policy is learned by maximizing the expected sum of pseudo rewards.

---

[1] Hazan, E., Kakade, S., Singh, K., & Van Soest, A. (2019, May). Provably efficient maximum entropy exploration. In International Conference on Machine Learning (pp. 2681-2691). PMLR.

[2] Mutti, M., & Restelli, M. (2020, April). An intrinsically-motivated approach for learning highly exploring and fast mixing policies. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 04, pp. 5232-5239).

[3] Liu, H., & Abbeel, P. (2021). Behavior from the void: Unsupervised active pre-training. Advances in Neural Information Processing Systems, 34, 18459-18473.

[4] Islam, R., Seraj, R., Bacon, P. L., & Precup, D. (2019). Entropy regularization with discounted future state distribution in policy gradient methods. arXiv preprint arXiv:1912.05104.

[5] Guo, Z. D., Azar, M. G., Saade, A., Thakoor, S., Piot, B., Pires, B. A., ... & Munos, R. (2021). Geometric entropic exploration. arXiv preprint arXiv:2101.02055.

[6] Lee, L., Eysenbach, B., Parisotto, E., Xing, E., Levine, S., & Salakhutdinov, R. (2020). Efficient exploration via state marginal matching.

Visitation distribution

Open questions and limitations.

- Requires learning a model of the state visitation distribution.
- It is learned on-policy via Monte Carlo.
- No formal justification for this particular objective.
- Mostly explores at the first steps of the decision process.
- Do we want to explore the past?

---

[1] Bolland, A., Lambrechts, G., & Ernst, D. (2025). Off-policy maximum entropy RL with future state and action visitation measures. 18th European Workshop on Reinforcement Learning.

Visitation distribution

Open questions and limitations.

- Requires learning a model of the state visitation distribution.
- It is learned on-policy via Monte Carlo.
- No formal justification for this particular objective.
- Mostly explores at the first steps of the decision process.
- Do we want to explore the past?

Could we enforce exploring future states and actions at each time step?

---

[1] Bolland, A., Lambrechts, G., & Ernst, D. (2025). Off-policy maximum entropy RL with future state and action visitation measures. 18th European Workshop on Reinforcement Learning.

Future states and actions

In a state s, followed by an action a, we want all future states and actions to be uniformly distributed.

Future states and actions

In a state s, followed by an action a, we want all future states and actions to be uniformly distributed.

How to measure the future states and actions?

$$d^{\pi,\gamma}(\bar{s}|s,a) = (1-\gamma)\sum_{\Delta=1}^{\infty}\gamma^{\Delta-1}p_{\Delta}^{\pi}(\bar{s}|s,a)$$

$$d^{\pi,\gamma}(\bar{s},\bar{a}|s,a) = \pi(\bar{a}|\bar{s})d^{\pi,\gamma}(\bar{s}|s,a)\ .$$

Future features

In a state s, followed by an action a, we want all future features to be uniformly distributed.

Future features

In a state s, followed by an action a, we want all future features to be uniformly distributed.

Stochastic projection of the states and actions.

$$q^{\pi}(z|s,a) = \int h(z|\bar{s},\bar{a})\pi(\bar{a}|\bar{s})d^{\pi,\gamma}(\bar{s}|s,a)\, d\bar{s}\, d\bar{a}\,.$$

Exploring with maximum entropy RL

In a state s, followed by an action a, we want all future features to be distributed according to q^*.

Exploring with maximum entropy RL

In a state s, followed by an action a, we want all future features to be distributed according to q^*.

This intrinsic behavior is enforced with an intrinsic reward.

$$\begin{aligned} R^{int}(s,a) &= -KL_z\left[q^\pi(z|s,a) \| q^*(z)\right] \\ &= \mathop{\mathbb{E}}_{z\sim q^\pi(\cdot|s,a)} \left[\log q^*(z) - \log q^\pi(z|s,a)\right] . \end{aligned}$$

Efficient intrinsic reward learning

The distribution of future states is the fixed-point of a contractive flow operator.

$$\mathcal{T}^{\pi} d^{\pi,\gamma}(\bar{s}|s,a) = d^{\pi,\gamma}(\bar{s}|s,a) \,.$$

Efficient intrinsic reward learning

The distribution of future states is the fixed-point of a contractive flow operator.

$$\mathcal{T}^{\pi} d^{\pi,\gamma}(\bar{s}|s,a) = d^{\pi,\gamma}(\bar{s}|s,a) \,.$$

The operator is a mixture between two distributions.

$$\mathcal{T}^{\pi} d^{\pi,\gamma}(\bar{s}|s,a) = (1-\gamma)p(\bar{s}|s,a) + \gamma \mathop{\mathbb{E}}_{\substack{s'\sim p(\cdot|s,a) \\ a'\sim\pi(\cdot|s')}} \left[d^{\pi,\gamma}(\bar{s}|s',a') \right] \,.$$

Efficient intrinsic reward learning

The distribution of future states is the fixed-point of a contractive flow operator.

$$\mathcal{T}^{\pi} d^{\pi,\gamma}(\bar{s}|s,a) = d^{\pi,\gamma}(\bar{s}|s,a) \; .$$

The operator is a mixture between two distributions.

$$\mathcal{T}^{\pi} d^{\pi,\gamma}(\bar{s}|s,a) = (1-\gamma)p(\bar{s}|s,a) + \gamma \mathop{\mathbb{E}}_{\substack{s' \sim p(\cdot|s,a) \\ a' \sim \pi(\cdot|s')}} \left[d^{\pi,\gamma}(\bar{s}|s',a') \right] \; .$$

Given a transition (s, a, s') and $d^{\pi,\gamma}$, we can sample $\bar{s}$ from $\mathcal{T}^{\pi} d^{\pi,\gamma}(\bar{s}|s,a)$.

Efficient intrinsic reward learning

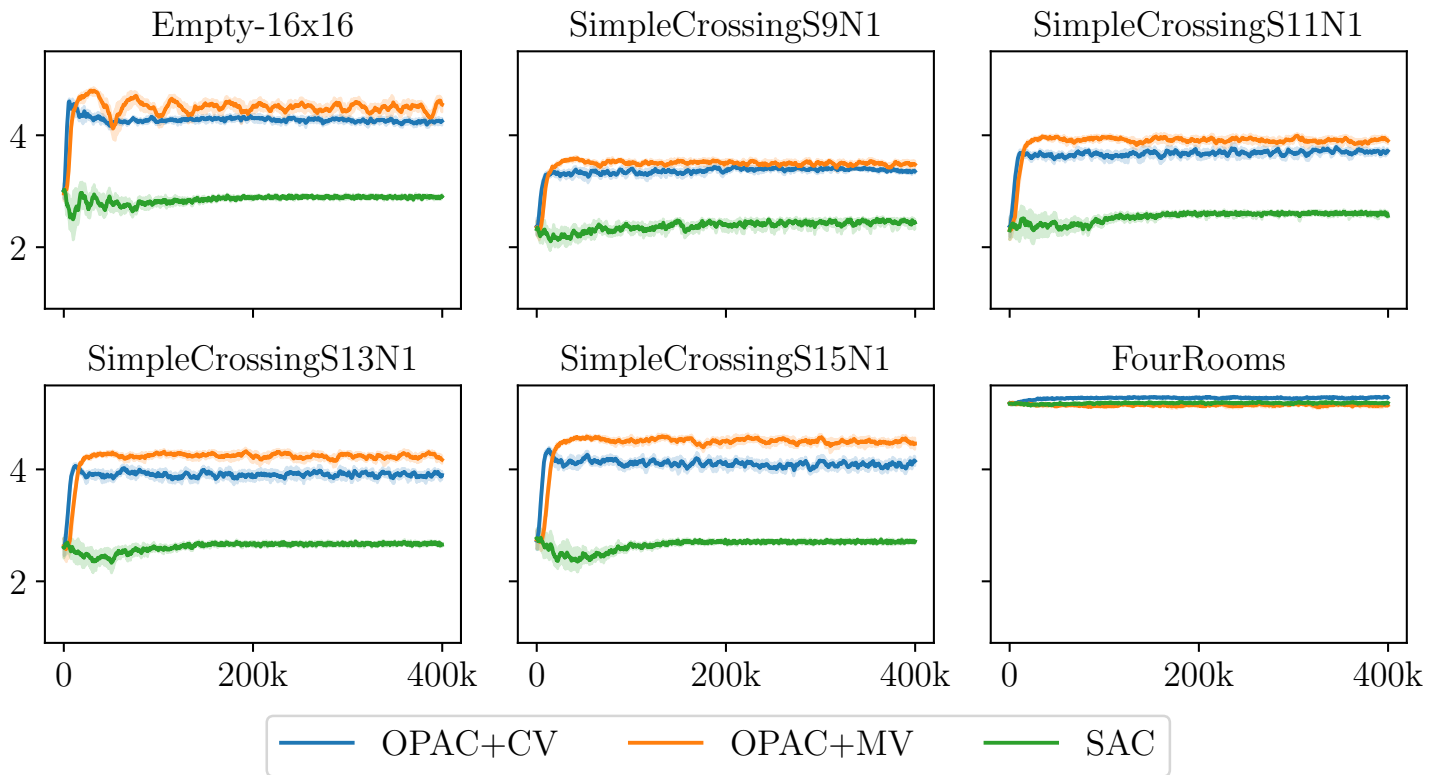
The fixed-point can be estimated with an off-policy TD-like method.

1. Collect a set of transitions.
2. Initialize the distribution d_ψ.
3. Iteratively solve the minimum cross-entropy problem

$$\arg\min_{\psi} \mathop{\mathbb{E}}_{\substack{s,a\sim g(\cdot,\cdot) \\ \bar{s}\sim(\mathcal{T}^\pi)^N d_\psi(\cdot|s,a)}} [-\log d_\psi(\bar{s}|s,a)] \ .$$

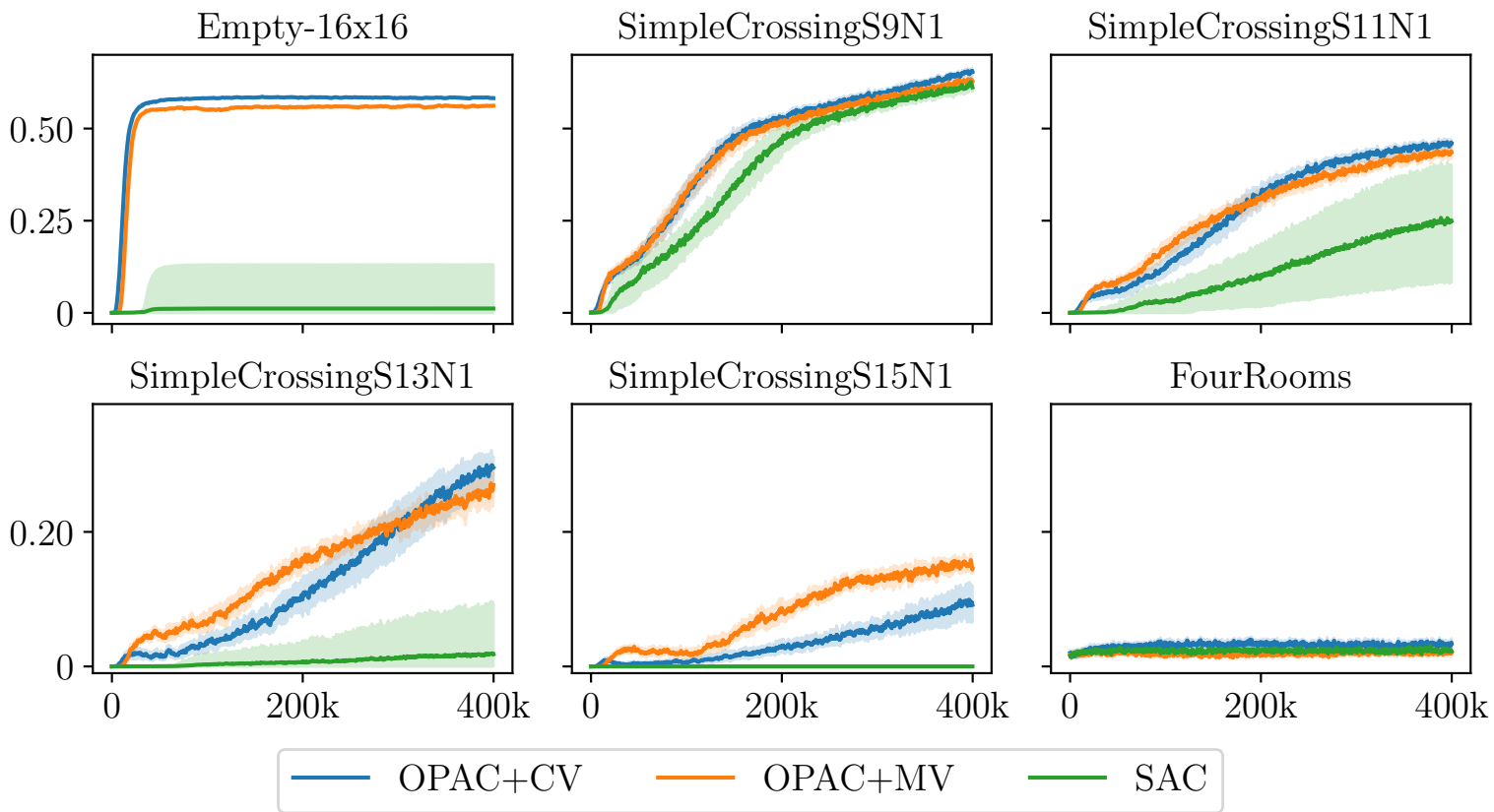
Minigrid exploration

This exploration method allows learning high-entropy policies (no MDP rewards).



Minigrid control

Expected return of policies with intrinsic exploration.



Conclusion

Conclusion

Exploration for learning optimal policies.

- Helps improve the convergence of policy gradient methods.
- Intrinsic motivation is widely inspired by intuition.

Conclusion

Exploration for learning optimal policies.

- Helps improve the convergence of policy gradient methods.
- Intrinsic motivation is widely inspired by intuition.

Other exploration purposes.

- Unsupervised discovery of behaviors.
- Unsupervised environment discovery.
- Unsupervised feature extraction.
- Generalization across tasks.