南京航空航天大學

Nanjing University of Aeronautics and Astronautics

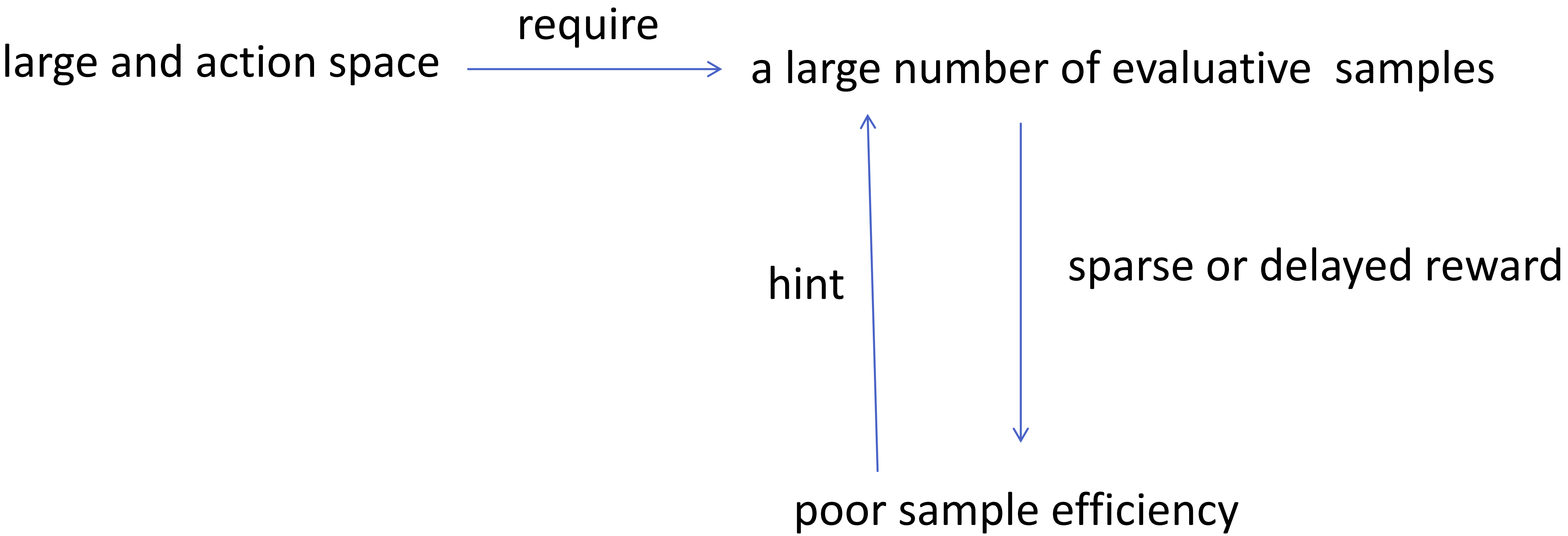
Shortest-Path Constrained Reinforcement Learning for Sparse Reward Tasks

Sungryull Sohn [*1 2] Sungtae Lee [*3] Jongwook Choi [1] Harm van Seijen [4] Mehdi Fatemi [4] Honglak Lee [2 1]

ICML 2021

Introduction

Central challenge



Preliminaries

Markov Decision Process (MDP)

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \rho, \gamma)$$

$$\pi^* = \arg\max_{\pi} \mathbb{E}^{\pi}_{s\sim\rho}\left[\sum_t \gamma^t r_t \mid s_0 = s\right] = \arg\max_{\pi} \mathbb{E}_{s\sim\rho}\left[V^{\pi}(s)\right].$$

model based $\begin{cases} P\pi = \pi \\ \sum \pi_i = 1 \end{cases}$ $V = R + \gamma PV \rightarrow V = (I - \gamma P)^{-1} R$

Constrained MDP

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\tau\sim\pi}\left[R(\tau)\right], \text{ s.t. } C(\pi) \leq \alpha.$$

transition cost function $c(s, a, r, s') \in \mathbb{R}$

$$C(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_t \gamma^t c(s_t, a_t, r_{t+1}, s_{t+1})\right]$$

Formulation

$$\tau = \{s_0, \dots, s_{\ell(\tau)}\}$$

Definition 1 (Path set).

$$\mathcal{T}^{\pi}_{s,s'} = \{\tau \mid s_0 = s, s_{\ell(\tau)} = s', p_{\pi}(\tau) > 0, s_t \neq s' \text{ for } \forall t < \ell(\tau)\}.$$

Definition 2 (Non-rewarding path set).

$$\mathcal{T}^{\pi}_{s,s',\mathrm{nr}} = \{\tau \mid \tau \in \mathcal{T}^{\pi}_{s,s'}, r_t = 0 \text{ for } \forall t < \ell(\tau)\}.$$

Definition 3 (π-distance from s to s').

$$D^{\pi}_{\mathrm{nr}}(s, s') = \log_{\gamma}\left(\mathbb{E}_{\tau \sim \pi:\ \tau \in \mathcal{T}^{\pi}_{s,s',\mathrm{nr}}}\left[\gamma^{\ell(\tau)}\right]\right)$$

Formulation

Definition 4 (Shortest path distance from s to s').

$$D_{\mathrm{nr}}(s, s') = \min_\pi D^\pi_{\mathrm{nr}}(s, s').$$

Definition 5 (Shortest path policy from s to s').

$$\pi \in \Pi^{SP}_{s \to s'} = \{\pi \in \Pi \mid D^\pi_{\mathrm{nr}}(s, s') = D_{\mathrm{nr}}(s, s')\}.$$

Definition 6 (Shortest-path constraint). *A policy π satisfies the shortest-path (SP) constraint if $\pi \in \Pi^{SP}$, where $\Pi^{SP} = \{\pi \mid$ For all $s, s' \in \mathcal{T}^\pi_{\Phi,\mathrm{nr}}$, it holds $\pi \in \Pi^{SP}_{s \to s'}\}$.*

对于所有存在没有奖励的可达点对(s, s’)， 必须走最短的策略集。

Formulation

Theorem 1. *For any MDP, an optimal policy* π^* *satisfies the shortest-path constraint:* $\pi^* \in \Pi^{SP}$.

difficulty: requires a distance predictor $D_{\text{nr}}(s, s')$.

Relaxation: k-shortest-path Constraint

a binary decision problem： k-reachability

is the state s' reachable from s within k steps?

Formulation

Definition 7 (k-shortest-path constraint). *A policy π satisfies the k-shortest-path constraint if $\pi \in \Pi_k^{SP}$, where*

$$\Pi_k^{SP} = \{\pi \mid \textit{For all } s, s' \in \mathcal{T}_{\Phi,\mathrm{nr}}^{\pi}, D_{\mathrm{nr}}^{\pi}(s, s') \leq k, \textit{ it holds } \pi \in \Pi_{s \to s'}^{SP}\}.$$

所有的 k 步可达无奖励的点对， 必须走最短的策略集

Lemma 2. *For an MDP $\mathcal{M}$, $\Pi_m^{SP} \subset \Pi_k^{SP}$ if $k < m$.*

Theorem 3. *For an MDP $\mathcal{M}$ and any $k \in \mathbb{R}$, an optimal policy π^* is a k-shortest-path policy.*

Proof. Theorem 1 tells $\pi^* \in \Pi^{\mathrm{SP}}$. Eq. (3) tells $\Pi^{\mathrm{SP}} = \Pi_{\infty}^{\mathrm{SP}}$ and Lemma 2 tells $\Pi_{\infty}^{\mathrm{SP}} \subset \Pi_k^{\mathrm{SP}}$. Collectively, we have $\pi^* \in \Pi^{\mathrm{SP}} = \Pi_{\infty}^{\mathrm{SP}} \subset \Pi_k^{\mathrm{SP}}$. □

Shortest-Path Reinforcement Learning (SPRL)

objective of RL with the k-SP constraint Π_k^{SP}

$$\pi^* = \arg\max_\pi \mathbb{E}^\pi \left[R(\tau)\right], \quad \text{s.t.} \quad \pi \in \Pi_k^{\mathrm{SP}}$$

k-SP constraint cost-based form:

$$\Pi_k^{\mathrm{SP}} = \{\pi \mid C_k^{\mathrm{SP}}(\pi) = 0\}, \text{ where}$$

$$C_k^{\mathrm{SP}}(\pi) = \sum_{(s,s' \in \mathcal{T}^\pi_{\Phi,\mathrm{nr}}):D^\pi_{\mathrm{nr}}(s,s') \le k} \mathbb{I}\left[D_{\mathrm{nr}}(s,s') < D^\pi_{\mathrm{nr}}(s,s')\right].$$

Shortest-Path Reinforcement Learning (SPRL)

apply the constraint to the on-policy trajectory $\tau = (s_0, s_1, \ldots)$
with (s_t, s_{t+l}) where $[t, t+l]$ represents each segment of τ with length l:

$$C_k^{\mathrm{SP}}(\pi) \simeq \mathbb{E}_{\tau\sim\pi}\left[C_k^{\mathrm{SP}}(\tau)\right]$$

$$C_k^{\mathrm{SP}}(\tau) = \sum_{(t,l):t\geq 0, l\leq k} \gamma^t \cdot \left(\prod_{j=t}^{t+l-1} \mathbb{I}\left[r_j = 0\right]\right) \cdot \mathbb{I}\left[D_{\mathrm{nr}}(s_t, s_{t+l}) < D_{\mathrm{nr}}^{\pi}(s_t, s_{t+l})\right]$$

$$\leq \sum_{(t,l):t\geq 0, l\leq k} \gamma^t \cdot \left(\prod_{j=t}^{t+l-1} \mathbb{I}\left[r_j = 0\right]\right) \cdot \mathbb{I}\left[D_{\mathrm{nr}}(s_t, s_{t+l}) < k\right]$$

$$\triangleq \widehat{C}_k^{\mathrm{SP}}(\pi)$$

Shortest-Path Reinforcement Learning (SPRL)

it is sufficient to consider only the cases $l = k$

Then, we simplify $\widehat{C}_k^{\text{SP}}(\tau)$ as

$$\widehat{C}_k^{\text{SP}}(\tau) = \sum_t \gamma^t \mathbb{I}\left[D_{\text{nr}}(s_t, s_{t+k}) < k\right] \prod_{j=t}^{t+k-1} \mathbb{I}\left[r_j = 0\right]$$

$$= \sum_t \gamma^t \mathbb{I}[t \geq k] \mathbb{I}\left[D_{\text{nr}}(s_{t-k}, s_t) < k\right] \prod_{j=t-k}^{t-1} \mathbb{I}\left[r_j = 0\right].$$

Finally, the per-time step cost c_t is given as:

$$c_t = \mathbb{I}[t \geq k] \cdot \mathbb{I}\left[D_{\text{nr}}(s_{t-k}, s_t) < k\right] \cdot \prod_{j=t-k}^{t-1} \mathbb{I}\left[r_j = 0\right],$$

feeding only the current time-step observation performs better than stacking the previous k-steps

Lagrange multiplier method to convert the objective

$$\min_{\lambda>0}\max_{\theta} L(\lambda,\theta) = \min_{\lambda>0}\max_{\theta} \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_t \gamma^t \left(r_t - \lambda c_t\right)\right]$$

Practical Implementation of the Cost Function.

$$c_t \simeq \mathrm{Rnet}_{k-1}(s_{t-k-\Delta t}, s_t) \cdot \prod_{j=t-k-\Delta t}^{t-1} \mathbb{I}\left[r_j = 0\right] \cdot \mathbb{I}(t \geq k + \Delta t).$$

$$\mathcal{L}_{\mathrm{Rnet}} = -\log\left(\mathrm{Rnet}_{k-1}(s_{\mathrm{anc}}, s_+)\right) - \log\left(1 - \mathrm{Rnet}_{k-1}(s_{\mathrm{anc}}, s_-)\right)$$

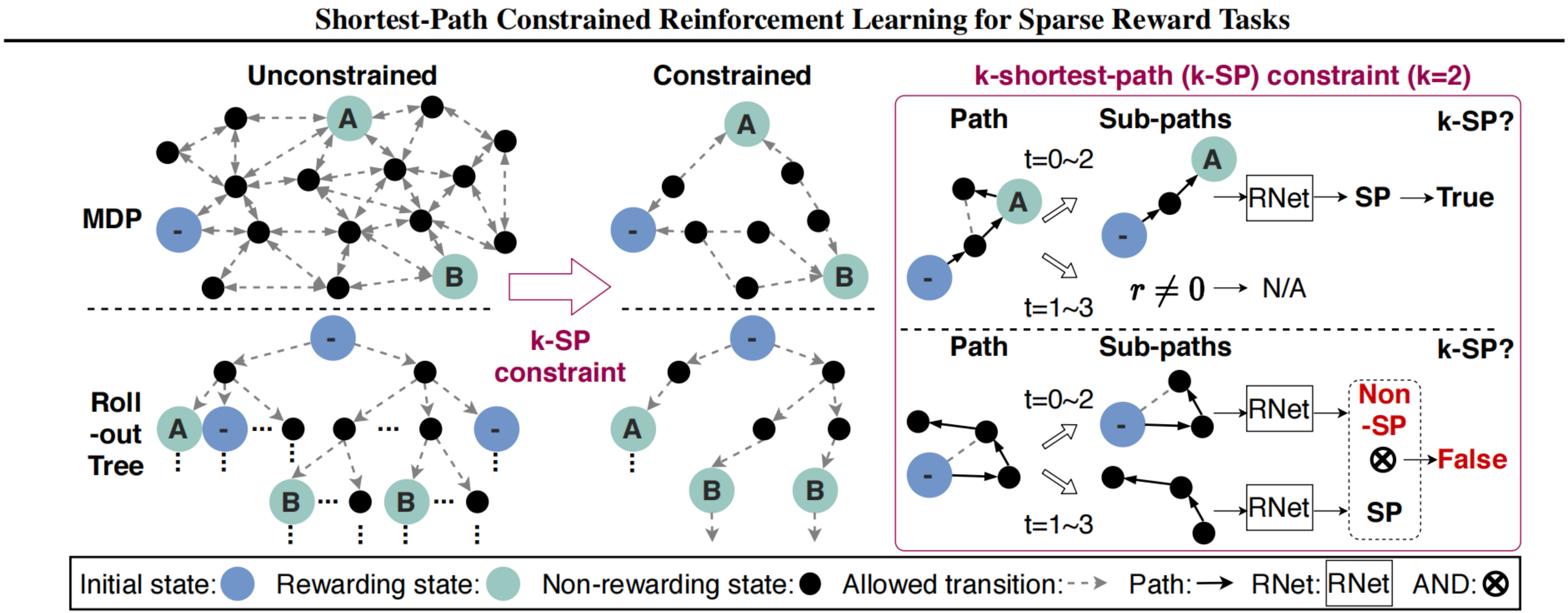
Shortest-Path Reinforcement Learning (SPRL)

Algorithm 1 Sampling the triplet data from an episode for RNet training

Require: Hyperparameters: $k \in \mathbb{N}$, Positive bias $\Delta^+ \in \mathbb{N}$, Negative bias $\Delta^- \in \mathbb{N}$

Initialize $t_{\text{anc}} \leftarrow 0$.
Initialize $S_{\text{anc}} = \emptyset$, $S_+ = \emptyset$, $S_- = \emptyset$.
while $t_{\text{anc}} < T$ **do**
    $S_{\text{anc}} = S_{\text{anc}} \cup \{s_{t_{\text{anc}}}\}$.
    $t_+ = \text{Uniform}(t_{\text{anc}} + 1, t_{\text{anc}} + k)$.
    $t_- = \text{Uniform}(t_{\text{anc}} + k + \Delta^-, T)$.
    $S_+ = S_+ \cup \{s_{t_+}\}$.
    $S_- = S_- \cup \{s_{t_-}\}$.
    $t_{\text{anc}} = \text{Uniform}(t_+ + 1, t_+ + \Delta^+)$.
end while
Return S_{anc}, S_+, S_-

Shortest-Path Reinforcement Learning (SPRL)



Experiment

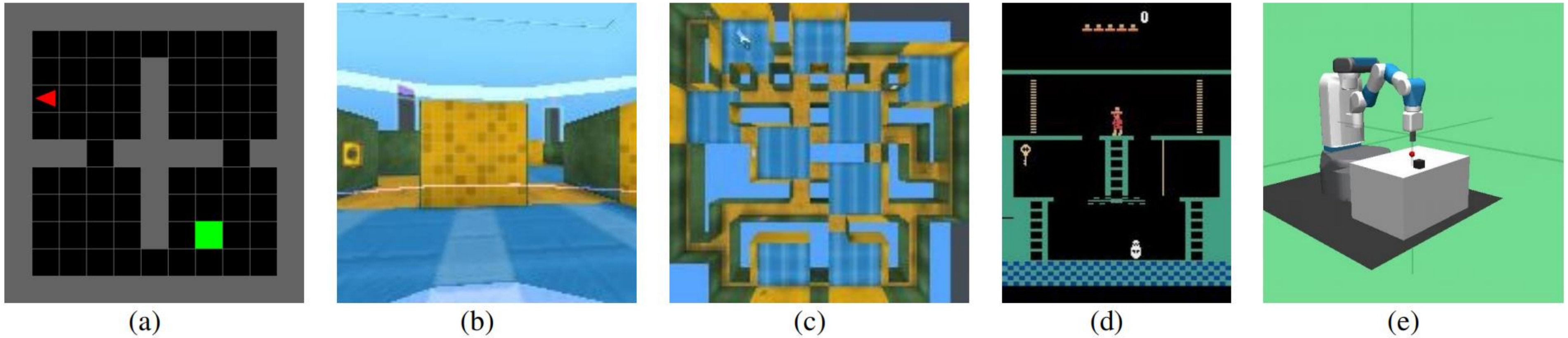


Figure 2. An example observation of (a) *FourRooms-11×11*, (b) *GoalLarge* in *DeepMind Lab*, (c) the maze layout (not available to the agent) of *GoalLarge*, (d) *Montezuma's Revenge* in *Atari*, and (e) *FetchPush-v1* in *Fetch*.

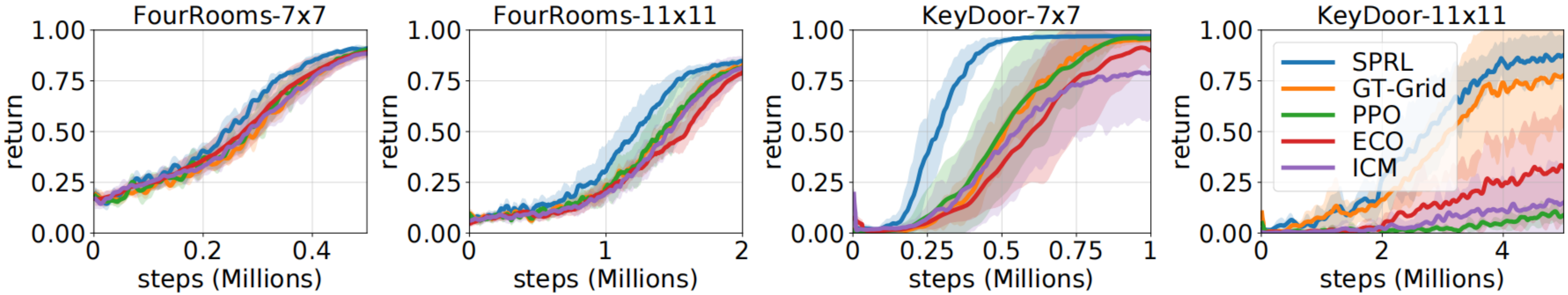


Figure 3. Progress of average episode reward on *MiniGrid* tasks. We report the mean (solid curve) and standard error (shadowed area) o the performance over six random seeds.

Experiment

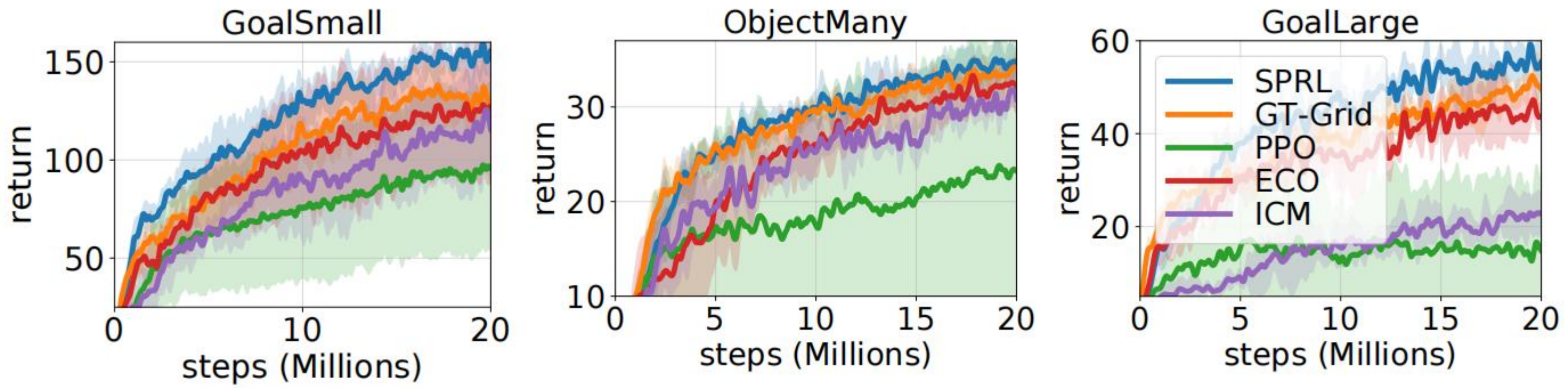
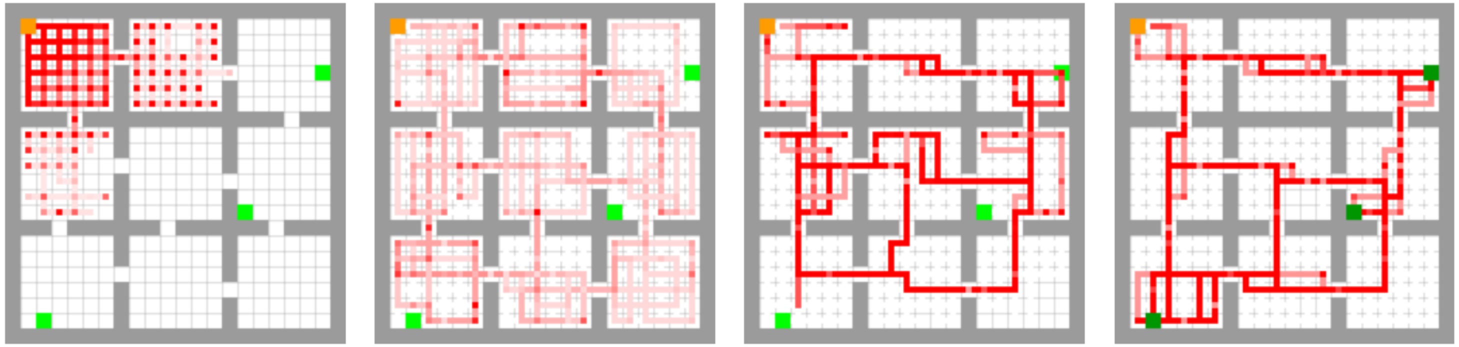


Figure 4. Progress of average episode reward on *DeepMind Lab* tasks. We report the mean (solid curve) and standard error (shadowed area) of the performance over four random seeds.



(a) **Random** (b) **GT-UCB** (c) **SPRL** (d) **SPRL**+Reward

Figure 8. Transition count maps for baselines and **SPRL**: (a), (b), and (c) are in a *reward-free* while (d) is in a *reward-aware* setting. In reward-free settings (a-c), we show rewarding states in light green only for the visualization, but the agent does not receive rewards from the environment. The location of the agent's initial state (orange) and rewarding states (dark green) are fixed. The episode length is limited to 500 steps.

Experiment

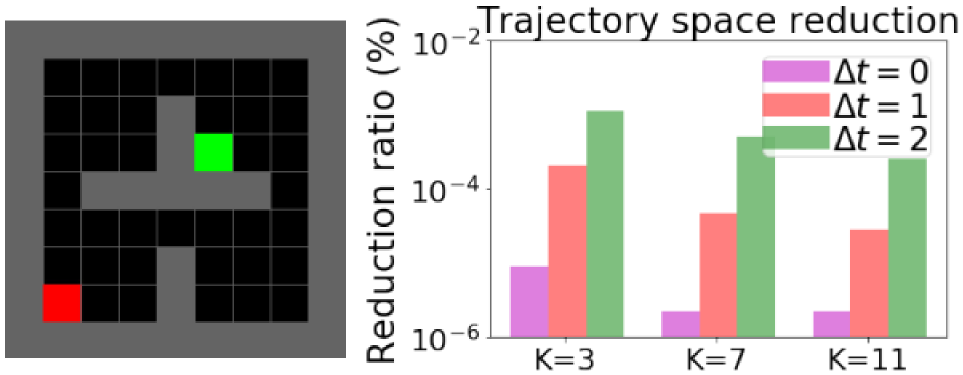


Figure 7. (Left) 7×7 Tabular four-rooms domain with initial agent location (red) and the goal location (green). (Right) The trajectory space reduction ratio (%) before and after constraining the trajectory space for various k and Δt with k-SP constraint. Even a small k can greatly reduce the trajectory space with a reasonable tolerance Δt.