We present a reinforcement learning method for long-horizon planning over high-dimensional state spaces by learning a state representation amenable to optimization and a goal-conditioned policy to abstract time.

**Summary**

We present a reinforcement learning method for long-horizon planning over high-dimensional state spaces by learning a state representation amenable to optimization and a goal-conditioned policy to abstract time.

**Introduction**

- **Problem:** Reasoning about long-horizon tasks is challenging; but reasoning about short-horizon tasks is easy.
- **Solution:** Decompose long-horizon task into sequence of short-horizon tasks, by planning subgoals that guide agent from start state to goal state.

**Latent Embeddings for Abstracted Planning (LEAP)**

- **Temporal difference model (TDM):** Framework from goal-conditioned RL that produces policy and value function, $V(s, g, t)$: how close agent will get from current state $s$ to goal state $g$ after $t$ time steps, when attempting to reach that state in $t$ steps.

- **Latent variable model:** Learn $\beta$-VAE [1] that maps latent variable $z$ to images $g$.

**LEAP Overview**

1. Train goal-conditioned policy, value function, and latent variable model.

$$\pi(s \mid g, t), V_{TDM}(s, g, t), \mathbf{z}$$

2. Plan: optimize subgoals in latent space.

$$\min_{z_{1:K}} \| V(s, z_{1:K}, t_{1:K+1}, g) \| - \lambda \sum_{k=1}^{K} \log p(z_k)$$

3. Execute: go to first subgoal.

$$\pi(s \mid g, t)$$

4. Replan after reaching first subgoal.

**Experiments**

- We can apply LEAP to state-based experiments where obstacles are unknown.

**References**
