Paper Critique

Anirud N, CE21B014

Course: DA7400, Fall 2024, IITM
Paper: Generating Adjacency-Constrained Subgoals in Hierarchical Reinforcement Learning
Date: 6 September 2024

Make sure your critique addresses the following points:

1. The problem the paper is trying to address
2. Key contributions of the paper
3. Proposed algorithm/framework
4. How the proposed algorithm addressed the described problem

Note: Be concise with your explanations. Unnecessary verbosity will be penalized. Please don't exceed 2 pages.

---

1 Problem Statement

Hierarchical RL suffers from training inefficiency as the action space on the high level is large. Searching in a large goal space comes with many issues in both the high-level policy's option goal generation and the learning of low-level policy. High-level exploration in such a large action space is inefficient. Techniques like action reduction require additional information, as a restricted action set may not be sufficient. This paper presents an optimality-preserving high-level action space reduction method for goal-conditioned HRL

2 Key Contributions

- The paper proves that the high-level action space can be restricted from a full goal space to a k-step adjacent region centered at the current space.
- As long as we progress to the optimal policy, we can replace the distance subgoals with k-step sub-goals
- Reducing action space increases efficiency in learning high-level and low-level policies
- Introduction of k step adjacency constraint for high-level action space. Theoretical proves that this constraint preserves the optimal hierarchical policy.

3 Proposed Algorithm/Framework

Shortest transition distance - a minimum number of steps needed to reach the target state from the start state is used to determine k-step adjacency. A direct Euclidian distance is not a good representation of the structure of the MDP. The shortest transition distance does not consider symmetry, nor does it consider the environment's irreversibility. K step adjacency constraint :

$$d_{\text{st}}(s_1, s_2) \approx \min_{\pi \in \{\pi_1, \pi_2, \cdots, \pi_n\}} \sum_{t=0}^{\infty} tP(\mathcal{T}_{s_1 s_2} = t|\pi), \tag{1}$$

Based on the adjacency value for any two states discovered in the sample trajectories, the adjacency matrix is constructed to check if the two states are adjacent ie the value of $d(s_1, s_2) <$

k. But in practice - this is non differential and we cannot generalise it to new visited states. So, we use an adjacency network to store this adjacency matrix. And the euclidian distance is regressed to find the shortest transition distance.:

$$\tilde{d}_{\text{st}}(s_1, s_2|\phi) = \frac{k}{\epsilon_k}\|\psi_\phi(g_1) - \psi_\phi(g_2)\|_2 \approx d_{\text{st}}(s_1, s_2), \tag{2}$$

We only need to ensure a binary relation for implementing the adjacency constraint, i.e., $\|\psi_\phi(g_1) - \psi_\phi(g_2)\|_2 > \epsilon_k$ for $d_{\text{st}}(s_1, s_2) > k$, and $\|\psi_\phi(g_1) - \psi_\phi(g_2)\|_2 < \epsilon_k$ for $d_{\text{st}}(s_1, s_2) < k$, as shown in Figure **??**. Inspired by modern metric learning approaches [**?**], we adopt a contrastive-like loss function for this distillation process:

$$\begin{aligned}\mathcal{L}_{\text{dis}}(\phi) = \mathbb{E}_{s_i, s_j \in \mathcal{S}} \big[& l \cdot \max\left(\|\psi_\phi(g_i) - \psi_\phi(g_j)\|_2 - \epsilon_k,\, 0\right) \\ & + (1 - l) \cdot \max\left(\epsilon_k + \delta - \|\psi_\phi(g_i) - \psi_\phi(g_j)\|_2,\, 0\right)\big],\end{aligned} \tag{3}$$

3.1 Combining HRL and Adjacency constraint

Now that we have the learned adjacency network, we can incorporate this constraint into the goal-conditioned HRL. High level policy objective :

$$\mathcal{L}_{\text{high}}(\theta_h) = -\mathbb{E}_{\pi^h_{\theta_h}} \sum_{t=0}^{T-1} \left(\gamma^t r^h_{kt} - \eta \cdot \mathcal{L}_{\text{adj}}\right), \tag{4}$$

where

$$\mathcal{L}_{\text{adj}}(\theta_h) = H\left(\tilde{d}_{\text{st}}\left(s_{kt}, \varphi^{-1}(g_{kt})|\phi\right), k\right) \propto \max\left(\|\psi_\phi(\varphi(s_{kt})) - \psi_\phi(g_{kt})\|_2 - \epsilon_k,\, 0\right), \tag{5}$$

4 Conclusions and Results

It yields a performance similar to that of the Oracle variant HRAC-O. It surpasses NoAdj by a large margin. The adjacency constraint methods prove to be very effective. It also works better than the NegReward variant, showing the importance of a "differentiable adjacency loss" that helps in stronger supervision. The K-step adjacency constraint helps reduce the problem of training inefficiency. Key issues include: making more effective and interpretable hierarchies to explore in a more meaningful action space.