Paper Critique

Shuvrajeet Das, DA24D402

Course: DA7400, Fall 2024, IITM
Paper: [Deep Laplacian-based Options for Temporally-Extended Exploration]
Date: [04-09-2024]

Make sure your critique Address 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 The problem the paper is trying to address

The paper addresses the problem of exploration in reinforcement learning (RL), particularly in complex environments. In RL, agents must explore their environments to gather information that allows them to learn effective policies for maximizing rewards. However, exploration is challenging because the agent needs to balance between exploiting known rewarding actions and exploring new actions that might lead to better long-term outcomes.

2 Key contributions of the paper

1. **Deep Covering Eigenoptions (DCEO):** Introduced DCEO, a novel deep RL algorithm extending Laplacian-based options to high-dimensional environments using neural networks.

2. **Scalable Approximation of Laplacian Eigenfunctions:** DCEO utilizes the generalized Laplacian objective:

$$\min_{f_1,\dots,f_d} \sum_{i=1}^{d} c_i f_i^\top L f_i \quad \text{s.t.} \quad f_i^\top f_j = \delta_{ij}$$

   where L is the graph Laplacian, f_i are the eigenfunction approximations, and c_i are coefficients. This avoids costly eigendecomposition.

3. **Fully Online Option Discovery and Learning:** DCEO integrates option discovery and reward maximization into a continuous process, enabling adaptability in non-stationary environments.

4. **Generalization to Diverse Environments:** Demonstrated the effectiveness of DCEO across various environments, including pixel-based and 3D navigation tasks.

5. **Temporally-Extended Exploration:** DCEO encourages structured exploration by maximizing intrinsic rewards derived from eigenfunctions:

$$r_{f_i}(s, s') = f_i(s') - f_i(s)$$

   leading to improved state coverage.

3 Proposed algorithm/framework

Algorithm 1 Fully Online DCEO Algorithm

```
for $i = 1$ to $T$ do
    if $i = 1$ or $i \perp \ell$ then
        $\tau \leftarrow$ True                                   ▷ Option termination
        $o \leftarrow -1$                                         ▷ No active option
        Reset environment and observe state $s$
    end if
    if $\tau$ then
        $\tau \sim U(0,1) < 1/D \quad \checkmark\ \tau$
    end if
    if $\tau$ then
        if $U(0,1) < \theta$ then then
            if $U(0,1) < \mu$ then then
                $o \leftarrow U(0);\ \tau \leftarrow$ False; $\alpha \sim \pi_o(\cdot|s)$
            else
                $o \leftarrow -1;\ \tau \leftarrow$ True; $\alpha \sim U(\mathcal{A})$
            end if
        else
            $\alpha \leftarrow \max_{\alpha \in \mathcal{A}} Q(s, \alpha)$
        end if
    else
        $\alpha \sim \pi_o(\cdot|s)$
    end if
    Execute $\alpha$, observe $r$, $s'$, $l \perp \ell$                ▷ $\ell$ is episode termination
    Store transition $(s, \alpha, r, s')$ in buffer $B$; $s \leftarrow s'$
    Sample a minibatch of transitions $(s_j, \alpha_j, r_j, s_{j+1})$
    Train each option with intrinsic reward
    Minimize the generalized Laplacian
    Train main learner on extrinsic reward
end for
```

3.1 Scalability with Deep Function Approximation:

Approximate Eigenfunctions: The algorithm uses neural networks to approximate the eigenfunctions of the graph Laplacian instead of relying on costly eigendecomposition. This allows the method to scale to large, complex environments, including those with pixel-based inputs.

3.2 Online and Continuous Option Discovery:

Single Continuous Learning Cycle: Unlike previous methods that required a separate option discovery phase, DCEO integrates option discovery and reward maximization into a single, continuous learning process. This means that the agent is always learning and refining its options while simultaneously learning to maximize rewards, making it adaptable to dynamic and non-stationary environments.

3.3 Generalization Across Tasks and Environments:

Effective in Diverse and Complex Environments: The paper demonstrates that DCEO is effective in a variety of environments, from simple grid worlds to complex 3D navigation tasks and Atari games