Reinforcement Learning with Sparse Rewards using Guidance from Offline Demonstration

Desik Rengarajan
Gargi Vaidya, Akshay Sarvesh, Dileep Kalathil, Srinivas Shakkottai

TEXAS A&M UNIVERSITY®

Problem

- Designing reward functions is a challenging problem in reinforcement learning.
- Providing sparse rewards that only indicate whether the task is completed partially or fully is easier.
- Existing RL algorithms fail to learn in a reasonable time in sparse environments due to needless exploration.
- For many problems there exists data that has been gathered over time using an empirically (sub-optimal) behavior policy.
- Additionally, this behavior data might only contain measurements of a subset of the true state.
- Can we use such data to aid learning in sparse reward environments?

Contributions

- Develop an algorithm, **Learning Online with Guidance Offline (LOGO)**, that can exploit offline demonstration data for reinforcement learning in a sparse reward setting.
- **Theoretical guarantee:** Derive a lower bound on the performance improvement of our algorithm.
- Derive a generalized version of the Performance Difference Lemma for policy dependent reward functions to develop a surrogate objective.
- Extend LOGO for the case where the demonstration data only contains a censored version of the true state.
- Demonstrate on MuJoCo and real world environments.

Algorithm: LOGO

TEXAS A&M UNIVERSITY
Department of Electrical & Computer Engineering

Each iteration of LOGO has two steps,

- **Step 1: Policy Improvement:** One step policy improvement using the Trust Region Policy Optimization (TRPO).

$$\pi_{k+1/2} = \arg\max_{\pi} \quad \mathbb{E}_{s\sim d^{\pi_k}, a\sim\pi}\left[A_R^{\pi_k}(s,a)\right] \quad \text{s.t.} \quad D_{\mathrm{KL}}^{\pi_k}(\pi, \pi_k) \leq \delta$$

- **Step 2: Policy Guidance:** Find a policy closest to the behavior policy, subject to it being in the trust region of the policy generated in the first step.

$$\pi_{k+1} = \arg\min_{\pi} \quad D_{\mathrm{KL}}^{\pi}(\pi, \pi_{\mathrm{b}}) \quad \text{s.t.} \quad D_{\mathrm{KL}}^{\max}(\pi, \pi_{k+1/2}) \leq \delta_k$$

- Step 2 ensures that the policy chosen is always guided by the behavior policy, but the level of alignment with the behavior policy can be reduced by shrinking the trust region.

TEXAS A&M UNIVERSITY
Department of Electrical
& Computer Engineering

Assumption

In the initial episodes of learning, $\mathbb{E}_{a\sim\pi_{\mathrm{b}}}\left[A_R^{\pi}(s,a)\right] \geq \beta > 0, \forall s.$

Theorem

Suppose π_k *and* $\pi_{k+1/2}$ *are related by the policy improvement step and* $\pi_{k+1/2}$ *and* π_{k+1} *are related by the policy guidance step, then*
(i) *If* $\pi_{k+1/2}$ *satisfies Assumption 1, then*

$$J_R(\pi_{k+1}) - J_R(\pi_k) \geq \frac{-\sqrt{2\delta}\gamma\epsilon_{R,k}}{(1-\gamma)^2} + \frac{\beta}{(1-\gamma)} - \frac{\epsilon_{R,k+1/2}}{(1-\gamma)}\sqrt{2D_{\mathrm{KL}}^{\pi}(\pi_{k+1},\pi_{\mathrm{b}})}.$$

(ii) *If* $\pi_{k+1/2}$ *does not satisfy Assumption 1, then*

$$J_R(\pi_{k+1}) - J_R(\pi_k) \geq -(\sqrt{2\delta}\gamma\epsilon_{R,k} + 3R_{\max}\delta_k)/(1-\gamma)^2.$$

Where $\epsilon_{R,k}$ *and* $\epsilon_{R,k+1/2}$ *are as defined before and* $R_{\max} = \max_{s,a}|R(s,a)|.$

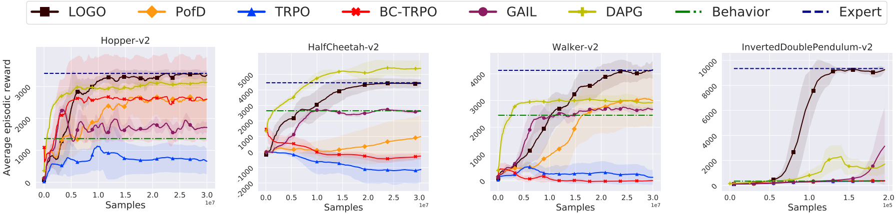
Practical Implementation of LOGO

- Derive a surrogate function for $D_{\mathrm{KL}}^{\pi}(\pi, \pi_{\mathrm{b}})$ that can be estimated using a policy dependent reward function C_π as $C_\pi(s, a) = \log(\pi(s, a)/\pi_{\mathrm{b}}(s, a))$.
- Approximate the policy guidance step as

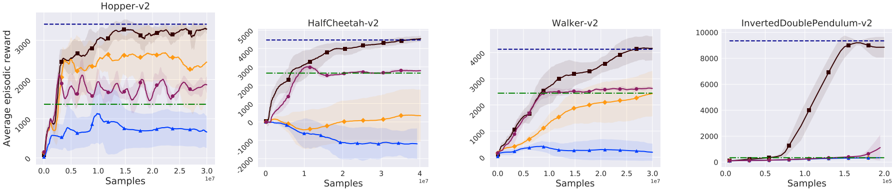
$$\pi_{k+1} = \arg\min_{\pi} \quad \mathbb{E}_{s \sim d^{\pi_{k+1/2}}, a \sim \pi(s,\cdot)}[A_{C_{\pi_{k+1/2}}}^{\pi_{k+1/2}}(s, a)] \quad \text{s.t.} \quad D_{\mathrm{KL}}^{\max}(\pi, \pi_{k+1/2}) \leq \delta_k.$$

- Train a discriminator using the demonstration data and the data generated by the policy $\pi_{k+1/2}$ to approximate $C_{\pi_{k+1/2}}$ when π_{b} is not available.
- Extend LOGO to incomplete observation setting by estimating $C_{\pi_{k+1/2}}$ using a projected version of the state.

Experiments: MuJoCo



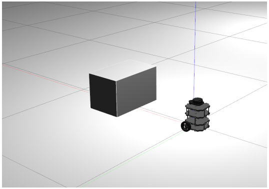
(a) Evaluation on MuJoCo with full offline observation.



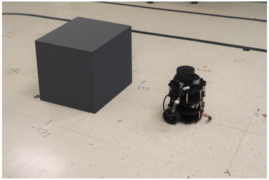
(a) Evaluation on MuJoCo with incomplete offline observation.

Experiments: TurtleBot

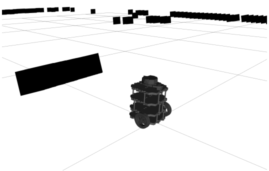
- Evaluate the performance of LOGO in the real-world using TurtleBot on two tasks
  - Waypoint tracking
  - Obstacle avoidance
- Create a sub-optimal π_{b} by training TRPO on our own low fidelity simulator, use it for guidance to train LOGO in Gazebo with sparse rewards, and evaluate in the real-world



(a) Gazebo setup



(b) Real-world setup



(c) Real-world 2D Lidar scan

Thank You!