Fast Policy Learning through Imitation and Reinforcement

Ching-An Cheng, Xinyan Yan, Nolan Wagener, and Byron Boots

Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, USA

Motivation

- The ability to adapt control policies to new environments is an important problem in robotics.
- **Sample Efficiency Issue** The agent needs to learn a good policy within limited interactions.
- Unlike learning in simulated worlds, real-world robot experiments are expensive and time-consuming.

Problem Statement

- **Goal** To quickly find a policy $\pi \in \Pi$ that minimizes an expected cost over trajectory distribution ρ_π

$$\min_{\pi\in\Pi} J(\pi), \qquad \text{where } J(\pi) = \mathbb{E}_{\rho_\pi}\left[\sum_{t=0}^{\infty} \gamma^t c(s_t, a_t)\right].$$

- This problem can be equivalently written as

$$\min_{\pi\in\Pi} \mathbb{E}_{s,t\sim d_\pi} \mathbb{E}_{a\sim\pi|s}[A_{\pi'}(s,a)],$$

where $A_{\pi'}$ is the advantage function with respect to some *fixed* reference policy π'.
- That is, find a policy π that performs better than the reference policy π' on its own state distribution d_π.

Approaches to Policy Learning

Reinforcement Learning (RL)

- Only *minimal* information about the problem is used.
- While learning does converge to a locally optimal solution, it may converge slowly.

Imitation Learning (IL)

- We often have access to *suboptimal* experts (like human experts and heuristic solutions).
- These expert policies can provide more informed policy search directions to speed up learning.
- But IL generally cannot learn a policy that is better than the expert policy.

Hybrid IL+RL

- Various methods have been proposed to combine RL and IL, with promising empirical performance.
- Some of them, however, rely on unrealistic assumptions (e.g. restarting the system at arbitrary states).
- Others are heuristically designed, lacking clear properties.

Our Approach to Hybrid IL+RL

- LOKI is a hybrid method that can both speed up learning and achieve locally optimal performance.
- Its design is motivated by the difference in the theoretical properties between RL and IL.
- We show that LOKI has good empirical and theoretical properties.
- Moreover, it is super simple to implement.

References

- R. Sutton, D. A. McAllester, S. Singh, and Y. Mansour, Policy gradient methods for reinforcement learning with function approximation, 2000
- Ross et al., G. Gordon, D. Bagnell, A reduction of imitation learning and structured prediction to no-regret online learning, 2011
- K.-W. Chang, A. Akshay , A. Agarwal, H. Daume III, and J. Langford, Learning to search better than your teacher, 2015
- W. Sun, A. Venkatraman, G. Gordon, B. Boots, and D. Bagnell, Deeply AggreVaTeD: differentiable imitation learning for sequential prediction, 2017
- W. Sun, D. Bagnell, and B. Boots, Truncated horizon policy search: deep combination of reinforcement and imitation, 2018

LOKI: Locally Optimal search after K-step Imitation

- LOKI splits policy optimization into two phases, with a switching time K that is randomly determined.

$$\text{Imitation Phase} \xrightarrow{\text{after } K \text{ steps of updates}} \text{Reinforcement Phase}$$

- At step n, the policy is updated by mirror descent with Bregman divergence D_{R_n} and step size η_n

$$\theta_{n+1} = \arg\min_{\theta\in\Theta} \langle g_n, \theta\rangle + \frac{1}{\eta_n} D_{R_n}(\theta\|\theta_n).$$

- g_n is a stochastic approximation of the (partial) derivative of $\mathbb{E}_{d_\pi}\mathbb{E}_\pi[A_{\pi'}]$ with respect to policy π.
- It uses a different reference policy π' in each phase.

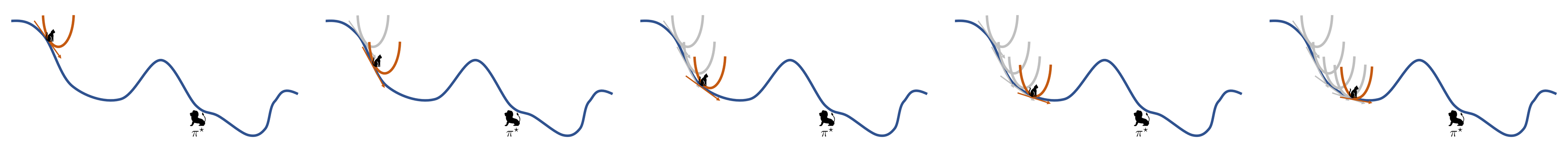
Reinforcement Phase (first-order RL)

- **Policy gradient**: the current policy π_n is the reference policy and

$$g_n = \nabla_\theta \mathbb{E}_{d_\pi}\mathbb{E}_\pi[A_{\pi_n}]|_{\pi=\pi_n} = (\nabla_\theta \mathbb{E}_{d_\pi})[0] + \mathbb{E}_{d_\pi}(\nabla_\theta\mathbb{E}_\pi)[A_{\pi_n}]|_{\pi=\pi_n} = \mathbb{E}_{d_\pi}(\nabla_\theta\mathbb{E}_\pi)[A_{\pi_n}]|_{\pi=\pi_n}$$

- With properly chosen R_n, mirror descent with policy gradient covers most model-free RL algorithms.
- **Majorization Optimization** With small enough step size, it constructs a *global* upper-bound approximation of the objective function and guarantees *monotonic* improvement of policies.

$$\mathbb{E}[J(\pi_1)] > \mathbb{E}[J(\pi_2)] > \mathbb{E}[J(\pi_3)] > \mathbb{E}[J(\pi_4)] > \mathbb{E}[J(\pi_5)] > \cdots$$



Imitation Phase (first-order IL)

- **Imitation gradient**: the expert policy $\pi^\star$ is the reference policy and

$$g_n = \nabla_\theta \mathbb{E}_{d_{\pi_n}}\mathbb{E}_\pi[\tilde{c}]|_{\pi=\pi_n} = \mathbb{E}_{d_{\pi_n}}(\nabla_\theta\mathbb{E}_\pi)[\tilde{c}]|_{\pi=\pi_n}$$

where $\tilde{c}(s,a)$ (e.g. $\mathbb{E}_{a^\star\sim\pi^\star}\|a-a^\star\|^2$) is chosen such that $\mathbb{E}_\pi[\tilde{c}] \geq \Omega(\mathbb{E}_\pi[A_{\pi^\star}])$, implying

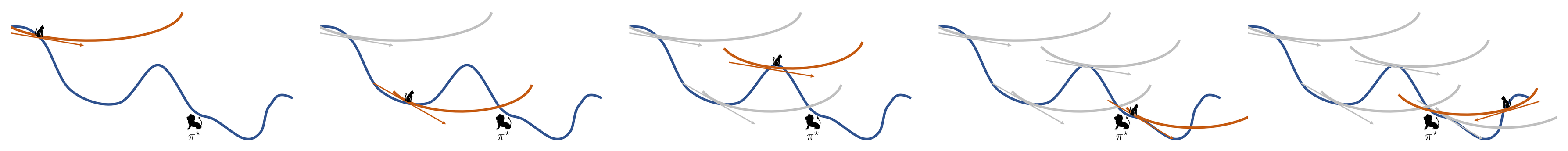
$$\mathbb{E}_{d_\pi}\mathbb{E}_\pi[\tilde{c}] \geq \Omega(J(\pi) - J(\pi^\star))$$

- Mirror descent with imitation gradient is a general first-order algorithm to online IL.
- It solves a surrogate RL problem: $\min_{\pi\in\Pi}\mathbb{E}_{d_\pi}\mathbb{E}_\pi[\tilde{c}]$. This surrogate RL problem has a nice property, called the *normalization property*: if $\pi^\star \in \Pi$, then there is a $\pi \in \Pi$ such that $\mathbb{E}_{d_\pi}\mathbb{E}_\pi[\tilde{c}] \leq 0$ for all π'.
- As a result, this surrogate RL problem can be solved *without* using the policy gradient:

$$\nabla_\theta \mathbb{E}_{d_\pi}\mathbb{E}_\pi[\tilde{c}] = (\nabla_\theta\mathbb{E}_{d_\pi})[\tilde{c}] + \mathbb{E}_{d_\pi}(\nabla_\theta\mathbb{E}_\pi)[\tilde{c}] \neq \mathbb{E}_{d_\pi}(\nabla_\theta\mathbb{E}_\pi)[\tilde{c}]$$

- Imitation gradient can have **smaller bias and variance** than policy gradient, as a Q-function estimate and the likelihood-ratio trick are not required.
- **Online Optimization** Mirror descent with imitation gradient generally leads to *on-average* improvement, and it constructs online loss surfaces which provide more global search directions toward the (suboptimal) expert policy up to ϵ_Π distance.

$$\frac{1}{N}\sum_{n=1}^{N} J(\pi_n) \leq J(\pi^\star) + \epsilon_\Pi + o(1)$$



Comparison of First-Order Oracles

Method	First-Order Oracle
POLICY GRADIENT (Sutton et al, 2000)	$\mathbb{E}_{d_{\pi_n}}(\nabla_\theta\mathbb{E}_\pi)[A_{\pi_n}]$
DAGGERED (Ross et al., 2011)	$\mathbb{E}_{d_{\pi_n}}(\nabla_\theta\mathbb{E}_\pi)[\mathbb{E}_{\pi^*}[D]]$
AGGREVATED (Sun et al., 2017)	$\mathbb{E}_{d_{\pi_n}}(\nabla_\theta\mathbb{E}_\pi)[A_{\pi^*}]$
SLOLS (Chang et al., 2015)‡	$\mathbb{E}_{d_{\pi_n}}(\nabla_\theta\mathbb{E}_\pi)[(1-\lambda)A_{\pi_n} + \lambda A_{\pi^*}]$
THOR (Sun et al., 2018)	$\mathbb{E}_{d_{\pi_n}}(\nabla_\theta\mathbb{E}_\pi)[A_{\pi_n,t}^{H,\pi^*}]$

D is a distance function in the action space (e.g. $\|a-a^*\|^2$)

‡ This is a simplification of what was originally used in (Chang et al., 2015) but it has the same convergence guarantee.

$A_{\pi_n,t}^{H,\pi^*}$ is a truncated advantage function

Results

Theoretical Properties

- (Informal) *Let N be the total number of iterations of policy update across both phases, and $K \ll N$ be randomly selected with probability $P(K=n) \propto n^p$ for some $0 \leq p \ll N$. Then* LOKI *performs almost as if it started from the expert policy, despite actually starting from a random policy.*
- Because LOKI learns an on-policy value function estimate in the Imitation Phase, the variance of the policy gradient in the Reinforcement Phase can be reduced.
- Optional batch IL can also be used to initialize the policy before the Imitation Phase.

Empirical Results

- We validated LOKI (implemented with TRPO) using several robotic control experiments in DART simulation environment and compared it with several baselines: Ideal (starting RL from the expert), TRPO (RL baseline), DAGGERED (IL baseline), THOR and SLOLS (RL+IL baselines).
- LOKI in general performs closely to Ideal and learns faster than other baselines.
- As LOKI uses on-policy estimates, it does not suffer from the covariate shift problem (i.e. change of input distributions) like other hybrid approaches.

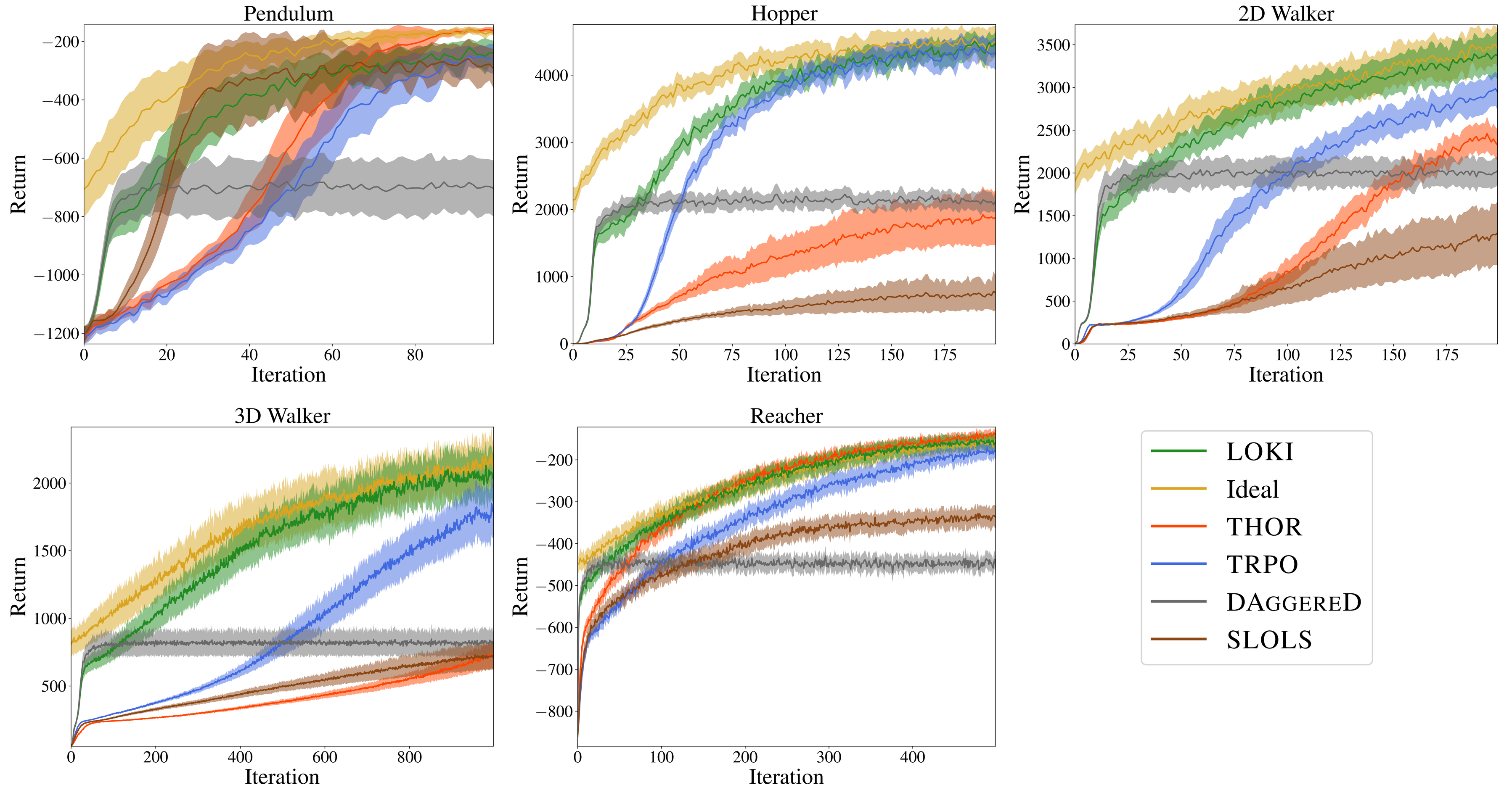


Figure: Learning curves. Shaded regions correspond to $\pm\frac{1}{2}$-standard deviation.