A Connection Between Importance Sampling and Likelihood Ratio Policy Gradients

Jie Tang
jietang@eecs.berkeley.edu

Pieter Abbeel
pabbeel@cs.berkeley.edu

EECS Department, University of California, Berkeley

Introduction

- Likelihood ratio policy gradient methods (PGMs) are state of the art techniques for reinforcement learning in continuous state spaces.
- Model-free learning with strong convergence guarantees
- PGMs have been successfully applied to a variety of difficult robotics problems, e.g.
  - Learning to hit balls with a bat [8]
  - Learning legged robot gaits [10]

Problem Formulation

- States $x_t \in \mathbf{R}^n$
- Actions $u_t \in \mathbf{R}^m$
- Reward function $r(x_t, u_t) \in \mathbf{R}$
- Discount factor γ.
- Sample trajectories τ by running policy π_θ.
- Given a parameterized policy representation $\pi_\theta(u_t|x_t)$, optimize discounted sum of reward

$$\min_\theta J(\theta) = E\left[\sum_{t=0}^{H} \gamma^t r_t\right]$$

Policy Gradient Methods

- Gradient descent technique: pick an initial starting θ_0 and update

$$\theta_{k+1} = \theta_k + \alpha_k \nabla_\theta J(\theta_k)$$

- Can choose stepsize adaptively (RPROP) [9]
- Given sampled trajectories τ^i, can compute Monte Carlo estimates of the gradient (REINFORCE)

$$\nabla_\theta J(\theta) = E[\nabla_\theta \log p_\theta(\tau) r(\tau)]$$

- Can add zero-mean baseline term to reduce variance and improve convergence rate.

$$\nabla_\theta J(\theta) = E[\nabla_\theta \log p_\theta(\tau)(r(\tau) - b)]$$

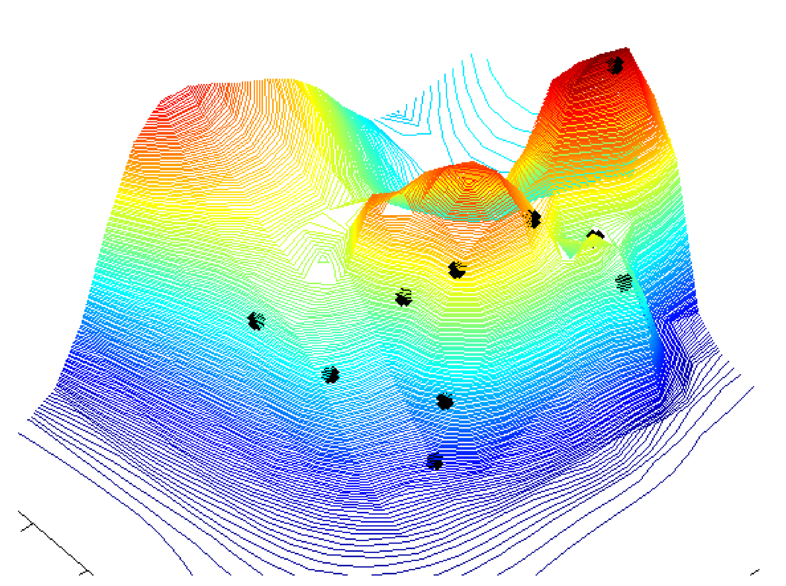
- Optimal minimum variance baseline has been shown to greatly improve convergence speed. [7]

Importance Sampling

- Importance sampling reweights old samples to create unbiased estimators for novel, arbitrary policies.
- Given sample trajectories τ, estimate value function

$$\widehat{J}(\theta) = E_q\left[\frac{p_\theta(\tau)}{q_\theta'(\tau)} r(\tau)\right]$$

- Below, a contour plot of $\widehat{J}$ for the first 2 cartpole policy parameters. Sample trajectories are marked in black.



Importance Sampling and PGMs

- Novel observation: given a single sample trajectory, the gradient of $\widehat{J}$ is the REINFORCE gradient direction.

$$\begin{aligned}
\nabla_{\theta_i} J(\theta) &= \lim_{\epsilon \to 0} \frac{J(\theta + \epsilon e_i) - J(\theta - \epsilon e_i)}{2\epsilon} \\
&= E\left[\frac{r(\tau)}{p_\theta(\tau)} \lim_{\epsilon \to 0} \frac{p_{\theta+\epsilon e_i}(\tau) - p_{\theta - \epsilon e_i}(\tau)}{2\epsilon}\right] \\
&= E\left[\frac{r(\tau)}{p_\theta(\tau)} \nabla_{\theta_i} p_\theta(\tau)\right] \\
&= E\left[\nabla_{\theta_i} \log p_\theta(\tau) r(\tau)\right]
\end{aligned}$$

- Suggests PGMs do not make full use of data.
- Past work: greedy hill climbing on $\widehat{J}$ [6]
- Our approach: find local optima of $\widehat{J}$ through numerical optimization.
  - Use effective sample size (ESS) to limit search areas of θ space with many samples [5]
  - Estimate Fisher information matrix and use it to estimate "natural" numerical gradient [1, 4]
  - Do optimal line search (e.g. Armijo rule) [2]
  - Use general minimum variance baseline for estimating $\widehat{J}$.

A Generalization of Min Variance Baseline

- Let $\Phi = \nabla_\theta \log p_\theta(\tau)$. An estimator for a scalar or vector-valued quantity admits a unbiased baseline of the form $E_p\left[b^T \Phi\right]$ or $E\left[B\Phi\right]$, respectively, since $\int_\tau p_\theta(\tau) \nabla_\theta \log p_\theta(\tau) d\tau = 0$.
- Extends naturally to IS setting.
- An unbiased estimator for $\widehat{J}$ is

$$\widehat{J} = E_q\left[\frac{p_\theta(\tau)}{q_\theta(\tau)}\left(r(\tau) - b^T\Phi\right)\right]$$

- Choose b by minimizing variance.

$$b = E_q\left[\frac{p(\tau)^2}{q(\tau)^2}\Phi\Phi^T\right]^{-1} E_q\left[\frac{p(\tau)}{q(\tau)}\Phi r(\tau)\right]$$

  - b is the product of an (IS) inverse-Fisher information matrix term and an (IS) REINFORCE gradient, i.e. it is the IS natural gradient.
- b is a product of expectations: in principle, can reapply baseline trick indefinitely. However, increases model complexity. Our approach uses the baseline trick once more to get min variance estimator for the REINFORCE term.

$$E_q\left[\frac{p(\tau)}{q(\tau)}\left(r(\tau)I - B\right)\Phi\right]$$

  - Compute B using least squares.
- Introducing baselines increases model complexity. Requires more samples (or IS).

Algorithm Summary

Input: policy π_θ, θ_0
$\theta \leftarrow \theta_0, paths \leftarrow \{\}, history \leftarrow [\theta_0]$
repeat

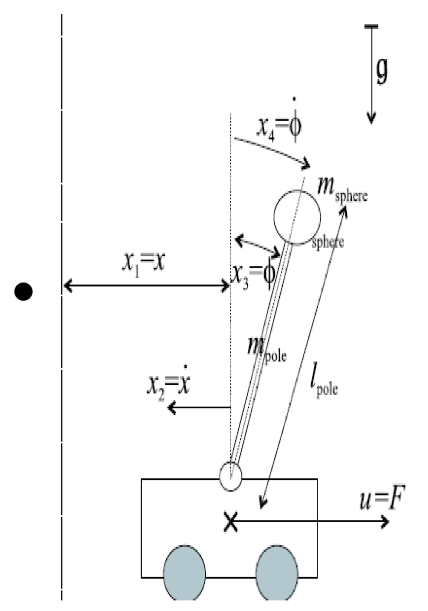
1. Draw samples from real system
for $1 : M$ **do**
$paths \leftarrow add\_path(paths, sample\_path(\pi_\theta))$
end for

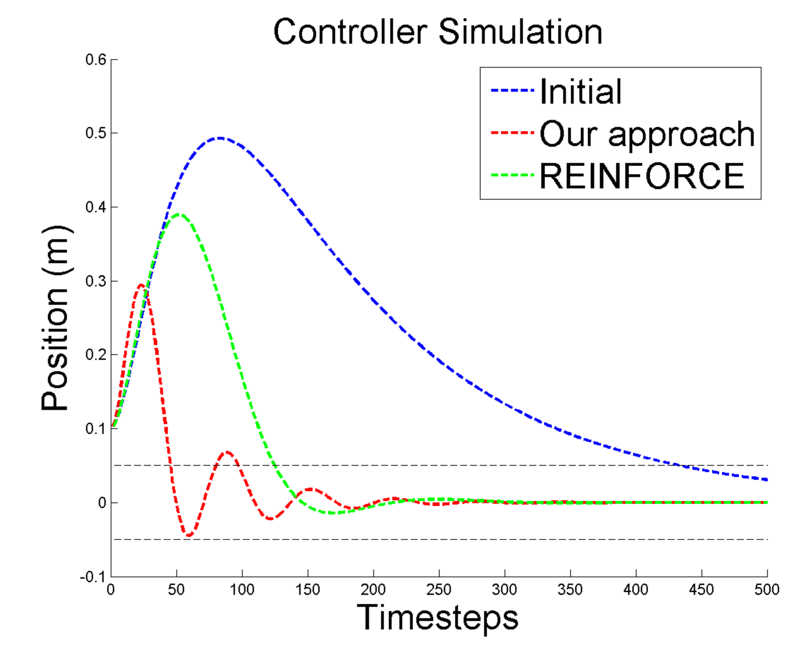
2. Use IS, optimal baseline(s) to learn value function
$\widehat{J}(\theta) \leftarrow E_q\left[\frac{p_\theta(\tau)}{q_{gh}(\tau)}(r - b^T \nabla_\theta \log p_\theta(\tau))\right]$

3. Run gradient descent searches from all past θ's
for $1 : N$ **do**
for all $\theta' \in history$ **do**
$g \leftarrow natural\_finite\_difference\_gradient\_step(\widehat{J})$
$\alpha \leftarrow linesearch(\widehat{J}, g)$
$\theta' \leftarrow \theta' + \alpha g$
end for
end for
$\theta = \arg\max_{\theta'} \widehat{J}(\theta')$
$history \leftarrow update\_history(history, \theta)$
until convergence

Cartpole Setup

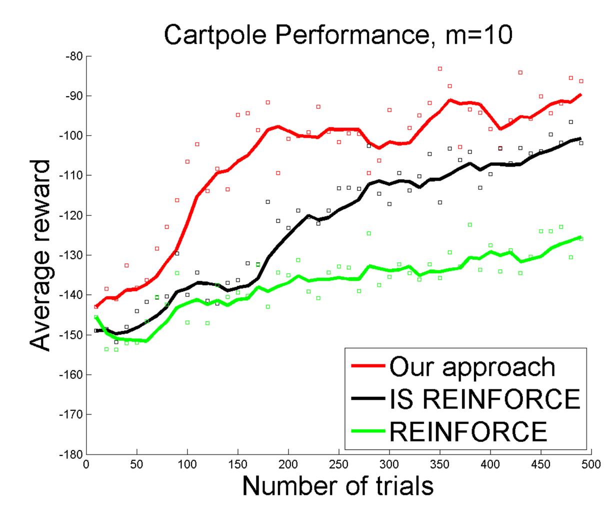
- $x \in \mathbf{R}^4$, $u \in \mathbf{R}$, $\theta = (K, \eta) \in \mathbf{R}^5$, $\eta \in \mathbf{R}$
- Policy $\pi_\theta(u|x) = N\left(Kx, 0.1 + \frac{1}{1+e^\eta}\right)$
- Reward is 0 inside target region, -2 if pole falls, -1 o.w.

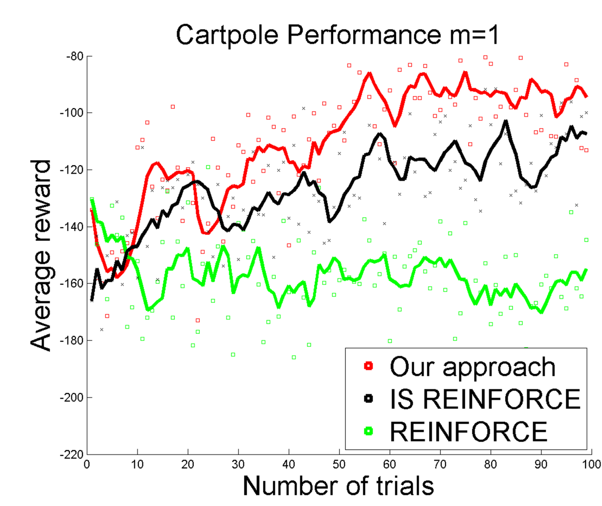




- Above right: cart position x under initial policy, policy learned with our approach, and policy learned with REINFORCE. Black lines show the target region where cost is 0.

Experimental Results





- REINFORCE uses optimal baseline and RPROP [9]
- IS REINFORCE is natural extension of REINFORCE, using IS to estimate gradient directly.
- (left) IS and optimal baseline do not account for the performance improvement over REINFORCE.
- (right) In practice, to minimize the number of trials on real hardware, we perform a policy update after every trial.
- Our approach performs well updating every trial. After 100 time steps we nearly equal performance after 500 time steps.
- REINFORCE gradient estimate is too noisy with 1 sample.

Conclusions

- PGMs are a special case of gradient descent over the $\widehat{J}$.
- Better approaches: use global search, not gradient descent
- Baselines used in PGMs are a special case of a general variance reduction technique.
  - Minimum variance unbiased estimators (MVUE) can be computed for estimating $\widehat{J}$.
  - Optimal baselines are themselves expectations, which can be given their own MVUE baselines.
  - Exploring applications of this technique to other domains
- Requires significantly fewer trials to learn good controllers for standard RL benchmark.

References

[1] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10, 1998.

[2] D. P. Bertsekas. *Nonlinear Programming.* Athena Scientific, 2004.

[3] P. Glynn. Likelihood Ratio Gradient Estimation: An Overview". In *Proceedings of the 1987 Winter Simulation Conference, Atlanta, GA*, 1987.

[4] S. Kakade. A natural policy gradient. In *Advances in Neural Information Processing Systems*, volume 14, 2001.

[5] J. S. Liu. *Monte Carlo Strategies in Scientific Computing.* Springer Publishing Company, Incorporated, 2008.

[6] L. Peshkin and C. R. Shelton. Learning from scarce experience. In *Proceedings of the Nineteenth International Conference on Machine Learning*, 2002.

[7] J. Peters and S. Schaal. Policy gradient methods for robotics. In *Proceedings of the IEEE International Conference on Intelligent Robotics Systems*, 2006.

[8] J. Peters, S. Vijayakumar, and S. Schaal. Natural actor-critic. In *Proceedings of the European Conference on Machine Learning (ECML)*, 2005.

[9] M. Riedmiller, J. Peters, and S. Schaal. Evaluation of policy gradient methods and variants on the cart-pole benchmark. In *IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*, 2007.

[10] R. Tedrake, T. W. Zhang, and H. Seung. Learning to walk in 20 minutes. In *Proceedings of the Fourteenth Yale Workshop on Adaptive and Learning Systems*, 2005.