Policy Gradient Methods

Eric Graves

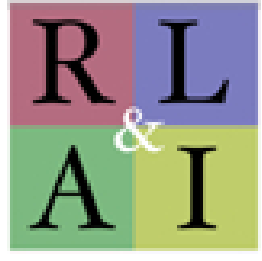


Table of Contents

Reinforcement Learning

- Learning to interact with an unknown environment through trial and error, with the goal of maximizing a reward signal.

Reinforcement Learning

- Learning to interact with an unknown environment through trial and error, with the goal of maximizing a reward signal.
- The agent's task is to find a policy that maximizes total reward.

Reinforcement Learning

- Learning to interact with an unknown environment through trial and error, with the goal of maximizing a reward signal.
- The agent's task is to find a policy that maximizes total reward.
- So far we've seen approaches requiring value functions: Q-learning, SARSA, etc.

Reinforcement Learning

- Learning to interact with an unknown environment through trial and error, with the goal of maximizing a reward signal.
- The agent's task is to find a policy that maximizes total reward.
- So far we've seen approaches requiring value functions: Q-learning, SARSA, etc.
- What about trying to learn the policy directly?

Why learn a policy directly? [1]

- Learning a policy could have fewer parameters.
- Choice of parameterization gives you freedom to act randomly in sophisticated ways.
- The best policy may be stochastic (ex. poker).
- But if not, policy gradient methods can still approach determinism.
- Natural extensions to continuous action spaces.
- The policy improvement theorem does not hold when using function approximation.
- Access to techniques from the optimization literature.

Why not learn a policy directly?

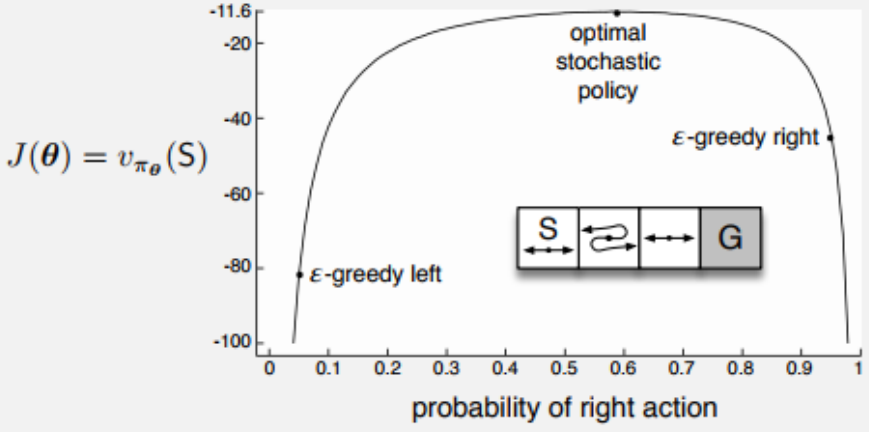
- If we're already learning a value function, then directly learning the policy requires extra computation and storage.
- Monte Carlo policy evaluation is extremely high variance.
- Primarily local convergence guarantees.[1]
- Smoothly changing the policy may slow learning unnecessarily.
- Extra step size makes them more difficult to apply.
- Softmax has an implicit temperature parameter (but not for action preferences).

[1] although Thomas [2] showed policy gradient methods converge to optimal policies whenever SARSA(λ) is guaranteed to converge to an optimal policy.

Short Corridor with Switched Actions [1]

Example 13.1 Short corridor with switched actions

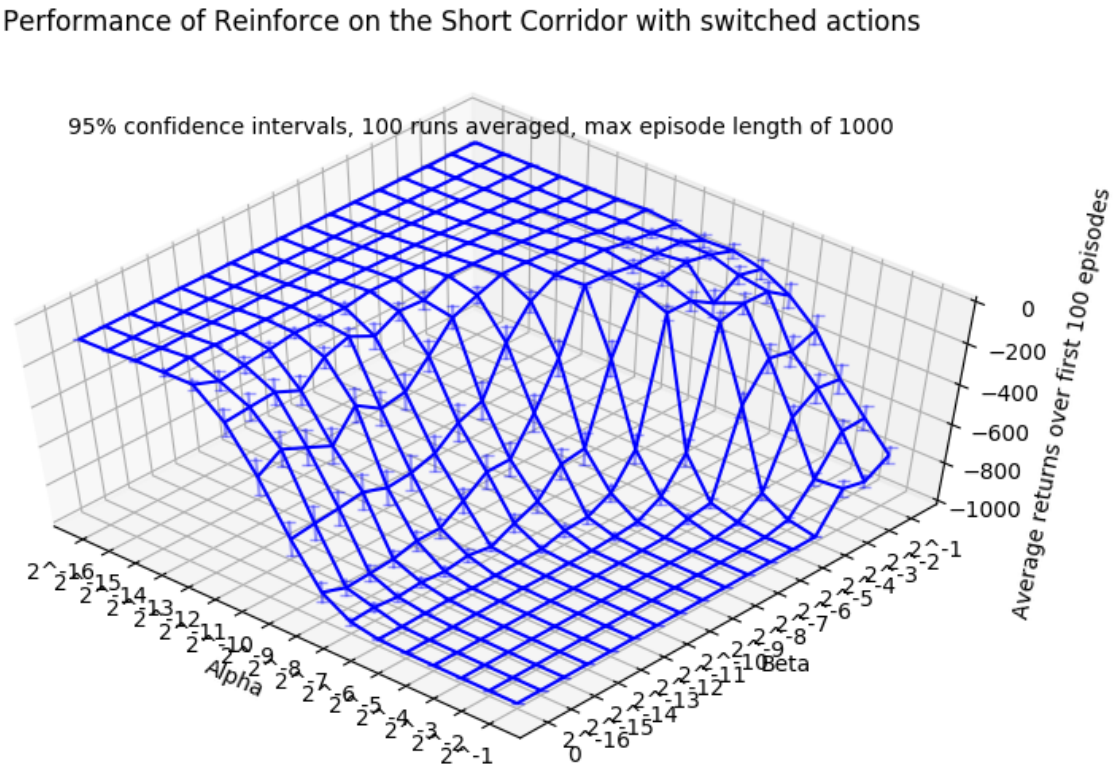
Consider the small corridor gridworld shown inset in the graph below. The reward is -1 per step, as usual. In each of the three nonterminal states there are only two actions, right and left. These actions have their usual consequences in the first and third states, but in the second state they are reversed, so that right moves to the left and left moves to the right. The problem is difficult because all the states appear identical under the function approximation. In particular, we define $\mathbf{x}(s, \text{right}) = [1, 0]^\top$ and $\mathbf{x}(s, \text{left}) = [0, 1]^\top$, for all s. An action-value method with ε-greedy action selection is forced to choose between just two policies: choosing right with high probability $1 - \varepsilon/2$ on all steps or choosing left with the same high probability on all time steps. If $\varepsilon = 0.1$, then these two policies achieve a value (at the start state) of less than -44 and -82, respectively, as shown in the graph. A method can do significantly better if it can learn a specific probability with which to select right. The best probability is about 0.59, which achieves a value of about -11.6.



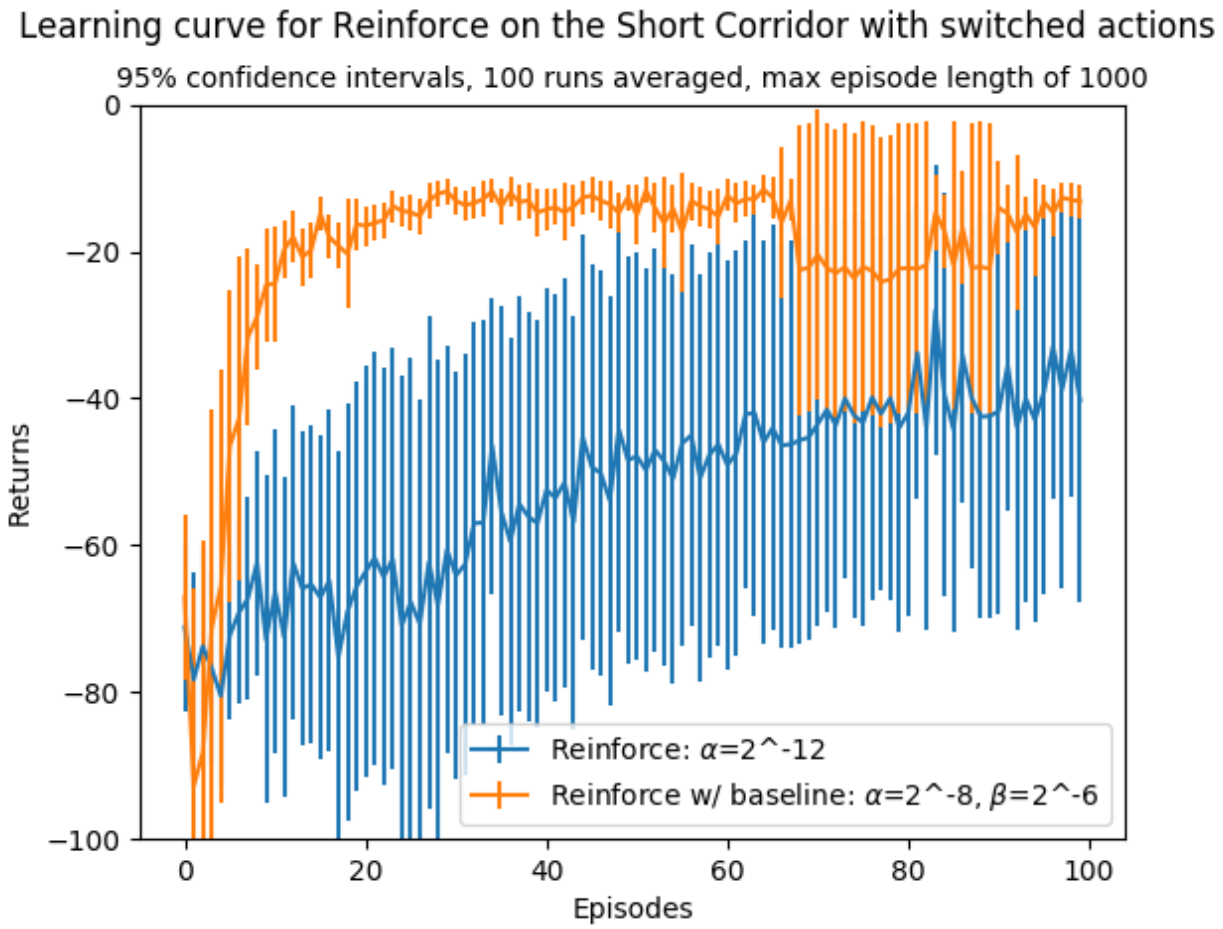
Methodology

- Parameters α and β swept between 2^{-16} and 2^{-1}, as well as $\beta = 0$ (no baseline).
- 100 episodes per run.
- Results averaged over 100 runs, and 95% confidence intervals calculated.
- Episodes were limited to 1000 timesteps, after which they were terminated.

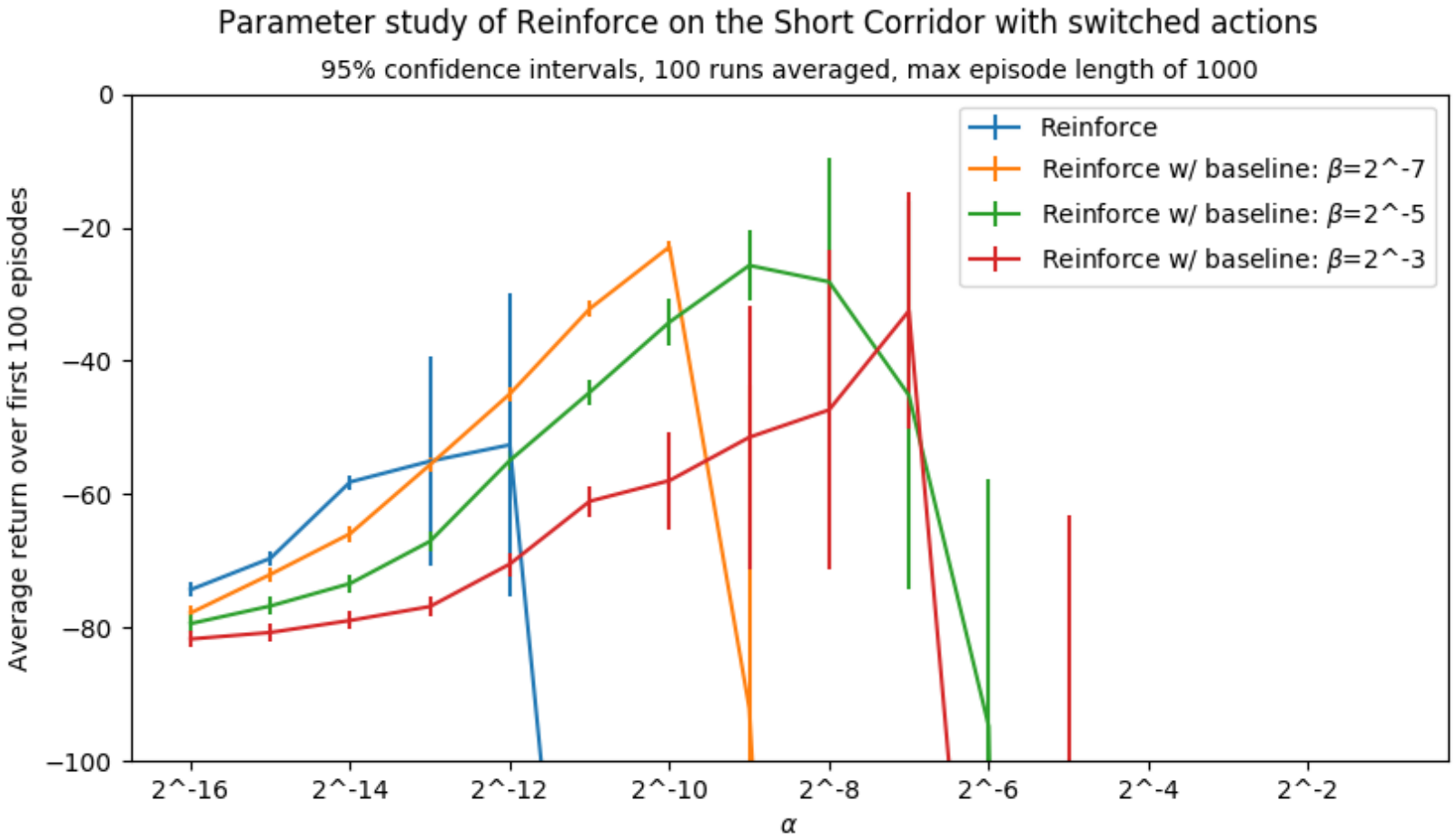
Parameter Sweep



Learning Curves



Parameter Study



Why use baselines?

Pros:

- Reduce variance.
- Allow a larger range of step sizes to be used effectively.
- Many choices of baseline (state-dependent, action-dependent, variance minimizing, etc.)

Cons:

- If the baseline is wrong, variance can be increased.
- Learning the baseline from samples could be difficult.

Project Ideas

- How can variance-reduced gradient techniques (SVRG, SARAH, SPIDER, etc.) be applied to policy gradient methods?

Project Ideas

- How can variance-reduced gradient techniques (SVRG, SARAH, SPIDER, etc.) be applied to policy gradient methods?
- How can accelerated gradient techniques (momentum, Nesterov, FISTA, etc.) be applied to policy gradient methods?

Project Ideas

- How can variance-reduced gradient techniques (SVRG, SARAH, SPIDER, etc.) be applied to policy gradient methods?
- How can accelerated gradient techniques (momentum, Nesterov, FISTA, etc.) be applied to policy gradient methods?
- Policy gradient theorems for alternative objective functions.
- How good of an estimate of the gradient is necessary?

References i

[1] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT press, 2nd edition, 2018.

[2] Philip Thomas. Bias in natural actor-critic algorithms. In *International Conference on Machine Learning*, pages 441–448, 2014.