Relative Entropy Policy Search

Jan Peters

March 17, 2007

Abstract

This technical report describes a cute idea of how to create new policy search approaches. It directly relates to the Natural Actor-Critic methods but allows the derivation of one shot solutions. Future work may include the application to interesting problems.

1 Problem Statement

In reinforcement learning, we have an agent which is in a state s and draws actions a from a policy π. Upon an action, it received a reward $r\left(s,a\right)=\mathcal{R}_{sa}$ and transfers to a next state s' where it will do a next action a'. In most cases, we have Markovian environments and policies, where $s'\sim p(s'|s,a)=\mathcal{P}_{sa}^{s'}$ and $a\sim\pi(a|s)$. The goal of all reinforcement learning methods is the maximization of the expected return

$$\bar{J}(\pi)=E\left\{\sum\nolimits_{t=0}^{T}r(s_t,a_t)\right\}. \tag{1}$$

We are generally interested in two cases, i.e., (i) the episodic open loop case where the system is always restarted from initial state distribution $p(s_0)$, and (ii) the stationary infinite horizon case where $T\to\infty$. Both have substantial differences in their mathematical treatment as well as their optimal solution.

1.1 Episodic Open-Loop Case

In the episodic open-loop case, a distribution $p(\tau)$ over trajectories τ is assumed and a return $R(\tau)$ of a trajectory τ, both are given by

$$p(\tau)=p(s_0)\prod\nolimits_{t=1}^{T}p(s_{t+1}|s_t,a_t)\pi(a_t|t), \tag{2}$$

$$R(\tau)=\sum\nolimits_{t=0}^{T}r(s_t,a_t). \tag{3}$$

The expected return can now be given as $\bar{J}(\pi)=\sum_\tau p(\tau)R(\tau)$. Note, that all approximations to the optimal policy depend on the initial state distribution $p(s_0)$. This case has been predominant in our previous work.

1.2 Stationary Case

Among the different case of infinite horizons (i.e., recurrent/stationary and transient case), the stationary case can be analyzed with a particular beauty. In this case, the system will converge to a stationary state distribution $\mu^\pi(s)$ given by

$$\mu^\pi(s') = \sum\nolimits_{s,a} p(s'|s,a)\pi(a|s)\mu^\pi(s), \tag{4}$$

$$1 = \sum\nolimits_{s} \mu^\pi(s). \tag{5}$$

upon a sufficient amount of steps. The expected return can now be expressed by $\bar{J}(\pi) = \sum_s \mu^\pi(s)\pi(a|s)r(s,a)$. This case has been predominant in the RL literature. We can express this optimal control problem as

$$\min_{\pi,\mu} \bar{J}(\pi) = \sum\nolimits_{s} \mu(s)\pi(a|s)r(s,a), \tag{6}$$

$$\text{s.t. } \forall s'.\mu(s') = \sum\nolimits_{s,a} p(s'|s,a)\pi(a|s)\mu(s), \tag{7}$$

$$1 = \sum\nolimits_{s,a} \mu(s)\pi(a|s), \tag{8}$$

$$\forall s,a.0 \leq \mu(s)\pi(a|s) \leq 1 \tag{9}$$

From this problem, we can directly derive the Bellman-Poisson equations. For simplicity, we omit the inequality constraints.

Theorem 1 *The optimality conditions for the stationary case are given by*

$$\mathcal{R}_{sa} - \bar{J}(\pi) + \sum\nolimits_{s'} \mathcal{P}_{sa}^{s'} V_{s'} - V_s = 0, \tag{10}$$

for all states s and actions a with $J(\pi) = \sum_{s,a} p_{sa}\mathcal{R}_{sa}$.

Proof. We write Lagrangian of the problem as

$$L = \sum\nolimits_{s} p_{sa}\mathcal{R}_{sa} - \sum\nolimits_{s'} V_{s'} \left(\sum\nolimits_{a'} p_{s'a'} - \sum\nolimits_{s,a} \mathcal{P}_{sa}^{s'} p_{sa}\right) - \lambda\left(1 - \sum\nolimits_{s,a} p_{sa}\right),$$

using $p_{sa} = \mu(s)\pi(a|s)$ and $\sum_a p_{sa} = \mu(s)$. Differentiation yields the Bellman-Poisson equation

$$\partial_{p_{sa}} L = \mathcal{R}_{sa} - \lambda + \sum\nolimits_{s'} \mathcal{P}_{sa}^{s'} V_{s'} - V_s = 0.$$

The convex combination of zeros yields zero, thus the p_{sa}-weighted sum of all optimality constraints yields

$$0 = \sum\nolimits_{s,a} p_{sa}\partial_{p_{sa}} L = \sum\nolimits_{s,a} p_{sa}\mathcal{R}_{sa} - \lambda + \sum\nolimits_{s'} \underbrace{\left(\sum\nolimits_{s,a} p_{sa}\mathcal{P}_{sa}^{s'}\right)}_{\mu(s')} V_{s'} - \sum\nolimits_{s,a} p_{sa} V_s,$$

and as the later two terms disappear, we obtain $\lambda = \sum_{s,a} p_{sa}\mathcal{R}_{sa} = \bar{J}(\pi)$. ■

The conditions of optimality do not directly yield an optimal control policy. In special cases, we can solve this problem, e.g., for discrete states, we can solve for the optimal policy by rewriting the optimality principle as a linear equation system and requiring a deterministic policy.

2 Relative Entropy Policy Search (REPS)

Reinforcement Learning methods can be considered "optimal control techniques by sample approximations". Here, we usually have a sampling policy which want to optimize. The bag of methods ranges from provably working heuristics (e.g., Q-Learning, SARSA, Resdidual Gradient, ...) to principled approaches (e.g., Policy Gradients Methods, PSDP, Natural Actor-Critic, Reward-Weighted Regression, Policy Search by Inference, DP or DDP with learned models, ...) which tend to be complicated in practice. In this paper, we take a *fundamentally different approach* to most previous methods where we attempt to *choose the optimal control policy regularized by our experience.* Interestingly, we can directly obtain solutions in that case.

2.1 REPS in the Episodic Open-Loop Case

Assume that we have sampled a lot of state-action pairs (e.g., from a infinite, recurrent path or an episodic setup), then we have a data distribution $q^{\pi}(\tau)$ determined by our sampling policy π and we have obtained rewards $R(\tau)$. The general goal now is to choose a new policy π' such that the resulting data distribution $p^{\pi'}(\tau)$ is as close as possible to the original one as possible while maximizing the rewards.The natural gradient brings us the following idea, let us bound the information loss as in

$$\begin{aligned}\max_{\pi'} J\left(\pi'\right) = & \underbrace{\sum\nolimits_{\tau} p^{\pi'}(\tau) C(\tau)}_{\text{Expected Return } J(\pi)} , \\ \text{s.t.}\quad \varepsilon \geq & \underbrace{\sum\nolimits_{\tau} p^{\pi'}(\tau) \log \frac{p^{\pi'}(\tau)}{p^{\pi}(\tau)}}_{\text{Regularization } D(p^{\pi'}, p^{\pi}(\tau))} , \\ 1 = & \sum\nolimits_{\tau} p^{\pi'}(\tau).\end{aligned}$$

Another way of looking at this problem is that we have a new goal

$$\begin{aligned}\max_{\pi'} J\left(\pi'\right) = & \underbrace{\sum\nolimits_{\tau} p^{\pi'}(\tau) R(\tau)}_{\text{Expected Return } J(\pi)} + \frac{1}{\eta} \underbrace{\sum\nolimits_{\tau} p^{\pi'}(\tau) \log \frac{p^{\pi'}(\tau)}{p^{\pi}(\tau)}}_{\text{Regularization } D(p^{\pi'}, p^{\pi}(\tau))} , \\ \text{s.t.}\quad 1 = & \sum\nolimits_{\tau} p^{\pi'}(\tau).\end{aligned}$$

where $\eta > 0$ is an open parameter. However, the latter case is unprincipled as it has no real interpretation.

In the simplest case, i.e., the one of an open loop policy where $p^{\pi'}(\tau)$ can be set for episode τ, we can derive the optimal solution straightfowardly.

Claim 1 *The Relative Entropy Policy Search algorithm for open-loop policies is given by*

$$p^{\pi'}(\tau) = \frac{p^{\pi}(\tau)\exp(\eta R(\tau))}{\sum_\tau p^{\pi}(\tau)\exp(\eta R(\tau))}, \tag{11}$$

where $\pi(\tau)$ denotes the previous open-loop policy and $\pi'(\tau)$ the current open loop policy. The parameter η can be obtained by optimizing the dual function $g(\eta) = L(p^{\pi'}, \eta)$.

Proof. We have a Lagrangian of

$$L = \sum\nolimits_\tau p^{\pi'}(\tau)\left(R(\tau) - \lambda + \frac{1}{\eta}\log\frac{p^{\pi'}(\tau)}{p^{\pi}(\tau)}\right) + \frac{\varepsilon}{\eta} + \lambda.$$

Differentiating with respect to $p^{\pi'}(\tau)$, setting to zero

$$\partial_{p^{\pi'}(\tau)}L = R(\tau) - \lambda + \frac{1}{\eta}\left(\log\frac{p^{\pi'}(\tau)}{p^{\pi}(\tau)} + 1\right) = 0 \tag{12}$$

and solving yields

$$p^{\pi'}(\tau) = p^{\pi}(\tau)\exp\left(\eta R(\tau)\right)\exp\left(\eta\lambda - 1\right).$$

Summing over all τ and inserting the equality constraint gives us

$$\sum\nolimits_\tau p^{\pi'}(\tau) = \exp\left(\eta\lambda - 1\right)\sum\nolimits_\tau p^{\pi}(\tau)\exp\left(\eta R(\tau)\right) = 1.$$

As a result, we obtain Equation (13). ■

This algorithm can also be motivated by the EM-like policy update point of view. Note both the similarity to the η-soft greedy update, however, also note that the previous policy plays a much larger role here, making the algorithm substantially different from SARSA-like approaches.

Nevertheless, open-loop policies are a rare special case as most problems require feedback. In this case, the solution is not as straightforward as for the open-loop case. We will now show how to perform Relative Entropy Policy Search (REPS) for both the stationary and the episodic case. Subsequently, we will take it and show its relation to Natural Policy Gradients.

2.2 REPS for the Stationary Closed-Loop Case

The stationary case is the most practical for dealing with infinite horizon problems. Again, we can give the reinforcement learning problem as a relative entropy-regularized problem. We have the goal function of

$$\max_{\pi',\mu'} J(\pi) = \sum\nolimits_{s} \mu'(s)\pi'(a|s)r(s,a) \tag{13}$$

$$\text{s.t.} \quad \varepsilon \geq \sum\nolimits_{\tau} \mu'(s)\pi'(a|s)\log\frac{\mu'(s)\pi'(a|s)}{\mu(s)\pi(a|s)},$$

$$\forall s'.\mu(s') = \sum\nolimits_{s,a} p(s'|s,a)\pi(a|s)\mu(s), \tag{14}$$

$$1 = \sum\nolimits_{s,a} \mu(s)\pi(a|s). \tag{15}$$

Just as in the open-loop case, we can give a clear algorithm. However, it will always requiring solving an additional optimization problem.

Claim 2 *The Relative Entropy Policy Search algorithm for stationary case yields a new policy*

$$\pi'(a|s) = \frac{\pi(a|s)\exp\left(-\eta\left(\mathcal{R}_{sa} + \sum_{s'}\mathcal{P}_{sa}^{s'}V_{s'}\right)\right)}{\sum_{a}\pi(a|s)\exp\left(-\eta\left(\mathcal{R}_{sa} + \sum_{s'}\mathcal{P}_{sa}^{s'}V_{s'}\right)\right)}, \tag{16}$$

where $V_{s'}$ is determined by optimizing

$$g\left(V\right) = -\eta^{-1}\log\left(\sum_{s,a}\mu(s)\pi(a|s)\exp\left(\eta\left(\mathcal{R}_{sa} + \sum\nolimits_{s'}\mathcal{P}_{sa}^{s'}V_{s'} - V_{s}\right)\right)\right). \tag{17}$$

Proof. We write Lagrangian of the problem as

$$\begin{aligned}
L &= \sum\nolimits_{s} p_{sa}\left(\mathcal{R}_{sa} + \frac{1}{\eta}\log\frac{p_{sa}}{q_{sa}}\right) - \sum\nolimits_{s'} V_{s'}\left(\sum\nolimits_{a'} p_{s'a'} - \sum\nolimits_{s,a}\mathcal{P}_{sa}^{s'}p_{sa}\right)\\
&- \lambda\left(\sum\nolimits_{s,a} p_{sa} - 1\right) - \frac{\varepsilon}{\eta},\\
&= \sum\nolimits_{s} p_{sa}\left(\mathcal{R}_{sa} + \frac{1}{\eta}\log\frac{p_{sa}}{q_{sa}} - \lambda + \sum\nolimits_{s'}\mathcal{P}_{sa}^{s'}V_{s'}\right) - \sum\nolimits_{s',a'} p_{s'a'}V_{s'} + \lambda - \frac{\varepsilon}{\eta},\\
&= \sum\nolimits_{s} p_{sa}\left(\mathcal{R}_{sa} + \frac{1}{\eta}\log\frac{p_{sa}}{q_{sa}} - \lambda + \sum\nolimits_{s'}\mathcal{P}_{sa}^{s'}V_{s'} - V_{s}\right) + \lambda - \frac{\varepsilon}{\eta},
\end{aligned}$$

using $p_{sa} = \mu'(s)\pi'(a|s)$, $q_{sa} = \mu(s)\pi(a|s)$, and $\sum_{a} p_{sa} = \mu'(s)$. Differentiation yields the Bellman principle of optimality with regularization cost

$$\partial_{p_{sa}}L = \mathcal{R}_{sa} + \frac{1}{\eta}\log\frac{p_{sa}}{q_{sa}} + \frac{1}{\eta} - \lambda + \sum\nolimits_{s'}\mathcal{P}_{sa}^{s'}V_{s'} - V_{s} = 0.$$

We determine the optimal state-action distribution

$$p_{sa} = q_{sa}\exp\left(\eta\left(\mathcal{R}_{sa} + \sum\nolimits_{s'}\mathcal{P}_{sa}^{s'}V_{s'} - V_{s}\right)\right)\exp\left(\eta\lambda - 1\right).$$

Now, we can obtain the dual function $g\left(\lambda, V\right) = -\eta^{-1}\sum_{s,a} p_{sa} + \lambda$. Similar as before, we obtain

$$\exp\left(\eta\lambda - 1\right) = \left(\sum\nolimits_{s,a} q_{sa}\exp\left(\eta\left(\mathcal{R}_{sa} + \sum\nolimits_{s'}\mathcal{P}_{sa}^{s'}V_{s'} - V_{s}\right)\right)\right)^{-1},$$

by using $\sum_{s,a} p_{sa} = 1$. Thus, we have $g\left(V\right) = -\eta^{-1} + \lambda = \eta^{-1} \log \exp\left(\eta\lambda - 1\right)$, which yields Equation (19). Using the fact that $\pi'(a|s) = p_{sa} / \sum_a p_{sa}$, we obtain

$$\pi'(a|s) = \frac{\mu\left(s\right)\pi(a|s)\exp\left(\eta\left(\mathcal{R}_{sa} + \sum_{s'} \mathcal{P}_{sa}^{s'} V_{s'} - V_s\right)\right)}{\sum_a \mu\left(s\right)\pi(a|s)\exp\left(\eta\left(\mathcal{R}_{sa} + \sum_{s'} \mathcal{P}_{sa}^{s'} V_{s'} - V_s\right)\right)},$$

and after canceling out several terms, we have Equation (18). ■

This algorithm has a surprising similarity to SARSA, i.e., it uses nearly exactly an η-soft policy update (only that the SARSA one was unweighted by the previous policy) and it has a critic which employs a TD-style error function. However, suprisingly, neither of these steps is exactly equal.

2.3 Relation to Natural Policy Gradients

One of the more interesting developments in policy gradient methods were natural policy gradients. These methods yielded a significant speed-up over traditional gradient methods which to date is not fully understood. However, the employment of the right cost function be the reason here, i.e., if we take a second order taylor expansion of $J\left(\pi'\right)$ in $p^{\pi'}(\tau)$, we obtain

$$\nabla_{p^{\pi'}(\tau)} J\left(\pi'\right) = R(\tau) + \frac{1}{\eta}\left(1 + \log\frac{p^{\pi'}(\tau)}{q^{\pi}(\tau)}\right), \tag{18}$$

$$\nabla^2_{p^{\pi'}(\tau)} J\left(\pi'\right) = \frac{1}{\eta}\frac{1}{p^{\pi'}(\tau)}. \tag{19}$$

If we assume that we only incrementally update with a $p^{\pi'}(\tau) = q^{\pi}(\tau) + \delta p\left(\tau\right)$, then we can give this cost function in the form

$$\begin{aligned} J\left(\pi'\right) &= J\left(\pi\right) + \sum\nolimits_{\tau} \nabla_{p^{\pi'}(\tau)} J\left(\pi'\right)^T \delta p\left(\tau\right) \qquad (20)\\ &+ \frac{1}{2}\sum\nolimits_{\tau,\hat{\tau}} \nabla^2_{p^{\pi'}(\tau)} J\left(\pi'\right) \delta p\left(\tau\right) \delta p\left(\hat{\tau}\right), \\ &= J\left(\pi\right) + \sum\nolimits_{\tau}\left[R(\tau) + \frac{1}{\eta}\left(1 + \log\left(1 + \delta p\left(\tau\right)\right)\right)\right]^T \delta p\left(\tau\right) \qquad (21)\\ &+ \frac{1}{2}\sum\nolimits_{\tau,\hat{\tau}}\left[\frac{1}{\eta}\frac{1}{p^{\pi'}(\tau)}\right]\delta p\left(\tau\right)\delta p\left(\hat{\tau}\right). \end{aligned}$$

Note, that this step was exact. If we assume make this step approximate, we can use

$$\nabla_{p^{\pi'}(\tau)} J\left(\pi'\right)\Big|_{p^{\pi'}(\tau) = q^{\pi}(\tau)} = R(\tau) + \frac{1}{\eta}, \tag{22}$$

and assume updates parametrized by θ, i.e., with

$$\delta p\left(\tau\right) = \nabla_\theta p^{\pi'}(\tau)^T \delta\theta. \tag{23}$$

Then, using the vanishing property of $\sum_{\hat{\tau}} \nabla_\theta p^{\pi'}(\hat{\tau}) = 0$, we have

$$J\left(\pi_{\theta+\delta\theta}\right) = J\left(\pi_\theta\right) + \underbrace{\sum\nolimits_{\tau} \left(R(\tau) + \frac{1}{\eta}\right) \nabla_\theta p^{\pi'}(\tau)}_{\text{Policy Gradient } \nabla_\theta J(\pi_\theta) \text{ with baseline } 1/\eta} \delta\theta \tag{24}$$

$$+ \frac{1}{2}\frac{1}{\eta} \sum\nolimits_{\tau,\hat{\tau}} \delta\theta^T \underbrace{\nabla_\theta p^{\pi'}(\tau) \left[\frac{1}{p^{\pi'}(\tau)}\right] \nabla_\theta p^{\pi'}(\tau)^T}_{\text{Fisher Information Matrix } F(\theta)} \delta\theta.$$

Solving for $\delta\theta$ yields the Natural Policy Gradient, i.e.,

$$\delta\theta = 2\eta F^{-1}(\theta) \nabla_\theta J\left(\pi_\theta\right). \tag{25}$$

From a policy search point of view, this result is surprising.

3 Conclusion

While not fully developed, the REPS approach may offer an alternative to the EM-like policy search methods. It basically optimizes the same cost function as the Natural Actor-Critic but allows larger steps than a small gradient updates. Its covariance properties are of course an open question. So is its efficient implementation and application to real problems.