2003

Covariant Policy Search

J. Andrew Bagnell
Carnegie Mellon University

Jeff Schneider
Carnegie Mellon University

Covariant Policy Search

J. Andrew Bagnell and Jeff Schneider
Robotics Institute
Carnegie-Mellon University
Pittsburgh, PA 15213
{*dbagnell,schneide*}*@ri.cmu.edu*

Abstract

We investigate the problem of non-covariant behavior of policy gradient reinforcement learning algorithms. The policy gradient approach is amenable to analysis by information geometric methods. This leads us to propose a natural metric on controller parameterization that results from considering the manifold of probability distributions over paths induced by a stochastic controller. Investigation of this approach leads to a covariant gradient ascent rule. Interesting properties of this rule are discussed, including its relation with actor-critic style reinforcement learning algorithms. The algorithms discussed here are computationally quite efficient and on some interesting problems lead to dramatic performance improvement over non-covariant rules.

1 Introduction

Much recent work in reinforcement learning and stochastic optimal control has focused on algorithms that search directly through a space of policies rather than building approximate value functions. Policy search has numerous advantages: domain knowledge may be easily encoded in a policy, the policy may require less representational power than a value-function approximation, there are simple extensions to the multi-agent domain, and convergent algorithms are known. Furthermore, policy search approaches have recently scored some encouraging successes [Bagnell and Schneider, 2001] [Baxter *et al.*, 1999] including helicopter control and game-playing.

One interesting aspect of existing gradient search algorithms, however, is that they are **non-covariant;** that is, a simple re-parameterization of the policy typically leads to a different gradient direction. (Not that found by applying the Jacobian of the re-parameterization to the gradient). This is a odd result; it is intuitively difficult to justify the gradient computation as actually indicating the direction of steepest descent. This problem is well recognized in the pattern-recognition and statistics communities and is a very active area of research. [Kakade, 2002] was, to the best of our knowledge the first to identify this problem in reinforcement learning and to suggest that techniques from information geometry may prove valuable in its solution.

Inspired by the work of [Amari and Nagaoka, 2000], Kakade proposed a "natural gradient" algorithm. In particular, [Kakade, 2002] proposed a scheme for generating a metric on parameter space that has interesting theoretical properties. Most convincingly, Kakade showed strong empirical evidence for the algorithm's usefulness. In particular, [Kakade, 2002] applies the algorithm to the Tetris problem of [Bertsekas and Tsitsiklis, 1996]. This problem is particularly interesting as value function methods (as for example described in [Bertsekas and Tsitsiklis, 1996]) demonstrate non-monotone performance; first policy iteration improves the policy dramatically, but after a small number of iterations the policy gets much worse. Normal gradient methods, including second-order and conjugate methods also prove very ineffective on this problem; even after a tremendous number of rounds they only mildly increase the performance of the game-player. The method presented in [Kakade, 2002] shows rapid performance improvement and achieves a significantly better policy than value-function methods (at their peak) in comparable time.

However, despite recognizing an interesting defect in the general approach and intuiting an apparently powerful algorithm, Kakade concludes that his method also fails to be covariant leaving the problem open. We present here what we believe to be an appropriate solution.

In Kakade's work, there is no proper probability manifold, but rather a collection of such (one for each state) based on the policy. As such, Kakade must rely on an ad-hoc method for generating a metric. Here instead we take a proper manifold, the distribution over paths induced by the controller, and compute a metric based on that. In the special case appropriate to the average reward RL formulation, Kakade's scheme is shown to give rise to a bona-fide natural metric, despite the observation in the paper that the learning was non-covariant. We believe this is an artifact– perhaps of step-length. Further, we note that parametric invariance does **not** require this metric– there are numerous covariant metrics. Rather, a stronger probabilistically natural invariance demands the metric used. We describe this result to motivate our method.

Importantly, the notion of the metric on the path-distribution allows us to provide very simple and natural extensions to Kakade's algorithm that cover the finite horizon, discounted start-state, and partially-observed reinforcement learning problems as well as the average reward one. Finally, for completeness, we describe the result discovered by Kakade relating the invariant metric and compatible actor-critic methods [Sutton *et al.*, 1999].

1.1 Problem Setup and Notation

A stochastic control problem consists of paths ξ (also called system trajectories) in a space Ξ, a distribution over path-space, $p(\xi)$, that is a function of a sequence of (finite) controls $(a_t)_{t\in\{0,\ldots,T\}}$, indexed by time, from a space A. Throughout this paper we will be considering Partially Observed Markov Decision Problems, in which there are state-variables (x_t) of the system that compose a path ξ and render the past and future independent. In a POMDP, $p(\xi)$ is defined by an initial state and next state transition probabilities $p(x_t|x_{t-1}, a_{t-1})$. We also typically have a sequence of outputs (observations) $(y_t)_{t\in\{0\ldots T\}}$ measurable with respect to x_t. A controller (or policy), π, usually parameterized by θ, is a feedback-type strategy that maps the history of observations $(y_t)_{t\in\{0..\tau\}}$ to a distribution over controls a_τ. Most of the derivations will be given in terms of memoryless stochastic controllers that map the current state to a distribution over actions, although they can be easily extended to finite window or recurrent representations that operate on only the observations. The goal in a control problem is to maximize the expected reinforcement $J(\theta) = \sum_\Xi p_\theta(\xi)r(\xi)$ with respect to θ. The sum is always assumed to exist for all controllers. The reward function on paths in a POMDP is additive in time (or in the infinite time case, discounted or averaged), and a function $R(x)$ of state, although the algorithms discussed here are not necessarily predicated on that assumption. Reinforcement learning is the adaptive version of the control problem where we attept to maximize the expected reinforcment by sampling trajectories and then improving our policy. We use the notation ∂_i throughout to denote $\frac{\partial}{\partial\theta_\iota}$where the parameter θ should be clear from context. For vectors x and y, we use the notation $(x||y)_M$ to denote the inner product with respect to metric M. The notation $\langle f(x)\rangle_{p(x)}$ indicates the expected value of the function f with respect to the distribution p.

2 Covariance and Riemannian manifolds

2.1 Meaning of steepest ascent

For direct policy search methods, we are interested in finding the direction of *steepest ascent* with respect to our current parameters. That is, we wish to maximize the reward function $J(\theta + \Delta\theta)$ subject to $(\Delta\theta||\Delta\theta) = \epsilon$, an infinitesimal. If we reparameterize our controller in terms of, e.g. ζ, and express the same effective infinitesimal policy change in terms of these parameters, ΔJ will of course remain the same. However, if we measure lengths using the naive dot product, the same effective change will not have the same size. It is this problem that leads to the odd behavior of “steepest descent” rules.

2.2 Non-covariant learning rule

To demonstrate the problem we first consider a simple two state, two action MDP described in [Kakade02]. (Figure 1) In state 0 executing action 0 returns to the same state and gets a reward of 1. Executing action 1 get no reward but transits to state 1. In state 1, action 0 returns to state 1 and gets reward 2. Action 1 transits to state 0 and achieves no reward.

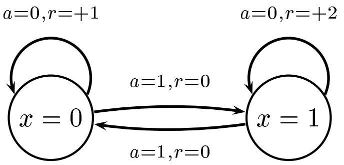


Figure 1: A two-state MDP. Each state has two controls one which self transitions (and earns the reward that labels the state) and another that simply transitions to the other state. Rewards occur only on the transitions between different states.

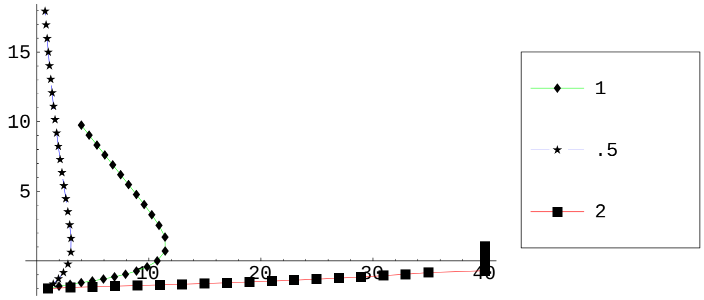


Figure 2: Log odds of actions for policies $logp(a = 0|x)/p(a = 1|x)$ on the two-state MDP (horizontal axis corresponds to state 0). Different curves correspond to varying λ. Notice the non-covariant policy behavior.

Consider parameterized probabilistic policies of the form $\pi(a = 0|x;\theta) = \frac{1}{1+e^{\lambda\theta_0\,\delta x=0+\theta_1\,\delta x=1}}$. λ is an arbitrary scale parameter that we use to demonstrate that even mildly different parameterization lead to dramatically different behavior. Below we plot the resulting track through the space of policies using the log ratio of probabilities of the actions for each of the possible states. We start from a policy that makes it somewhat more likely to choose action 0 in state 0 and action 1 in state 1. We then scale λ to 1, .5, and 2 and plot the log odds of the policy from state 1 (Figure 2).

The non-covariant behavior is quite clear in this graph. Further, we note that it is *very* difficult to achieve a good policy using this algorithm from this starting point as the *wrong* action becomes overwhelmingly likely to be chosen from state 0. If sampling were used to compute the gradient, we would nearly never get samples from state 1– which we need to improve the policy.

2.3 Path distribution manifolds

The control problem is essentially a coupled one of optimization and integration over the space of possible paths, Ξ. This motivates the idea that instead of considering the (arbitrary) distance in terms of parameterization of the policy, we may consider distance in terms of the change in distribution over paths resulting from the policy change. We may view the distribution over paths $p(\xi;\theta)$ as a parameterized manifold (nominally embedded in $R^{|\Xi|}$) of dimension $|\theta|$. This takes some work to visualize; as an example consider a space of three possible paths. All possible distributions over these three paths can be smoothly represented with two parameters. In Figure 3 (left) we see one visualization with the embedding

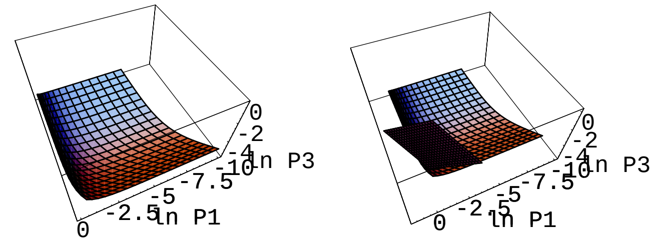


Figure 3: An example probability manifold over paths. Each axis represents the log probability of one of the three possible paths. The manifold is formed as we vary the parameters defining it throughout their domain. On the right we attach the tangent space at a particular point in parameter space.

$\log p(\xi;\theta))$:

With n parameters, assuming no redundancy, we generate an n dimensional manifold. In the case pictured, it is the set of all distributions on 3 paths, but in general, the dimensionality of the path probability manifold will be tremendously less than the number of possible paths. It is important to understand the manifold under consideration is that of the probability distribution over paths and not of paths themselves.

The study of parameterized manifolds like that pictured above is the domain of differential geometry. Here we are interested in establishing a *Riemannian* structure on the manifold of paths. By that we mean we wish to establish a metric on the tangent space (the local linear approximation to the manifold at a point ξ), so we can measure small parameter changes. We do this with the metric $(X||Y)_{p_\theta(\xi)}$ on the tangent space (which is spanned by the partials with respect to each parameter) as $\sum_{i,j} G^{\text{ij}}(\xi) X_i Y_j$, where G is a positive definite matrix. This is a very natural thing to do– instead of just the standard dot product, we have a dot product that allows us to represent rotations and scalings (exactly what a positive definite matrix can represent) and that can vary throughout the manifold. In Figure 3 (right) we depict the tangent space at a point in our parameterization as the local linear approximation at that point.

2.4 Steepest ascent on Riemannian manifold

There are two questions that then naturally arise– what is steepest descent on a function defined over a Riemannian space, and is there a Riemannian metric on the manifold of the paths that is in some sense natural. We quickly answer the first question, and pursue the second in the next two sections. A Lagrange multiplier argument (given schematically here) makes it easy to see the form of the steepest descent direction.

$$\text{min } L(\theta + \delta\theta) = L(\theta) + \nabla L(\theta) \cdot \delta\theta \tag{1}$$

$$\text{subject to } (\theta||\theta) = \epsilon \tag{2}$$

Form the Lagrangian:

$$\begin{aligned} \mathcal{L}(\theta, \lambda) &= L(\theta) + \nabla L(\theta) \cdot \delta\theta + \\ &\quad \lambda(G^{\text{ij}} \delta\theta_i \delta\theta_j - \epsilon) \end{aligned} \tag{3}$$

and take derivatives with respect to each $\delta\theta$, then set to zero to solve for the optimal direction:

$$\nabla_{\delta\theta} \mathcal{L}(\delta\theta, \lambda) = \nabla L(\theta) + \lambda G^{\text{ij}} \delta\theta_j = 0$$

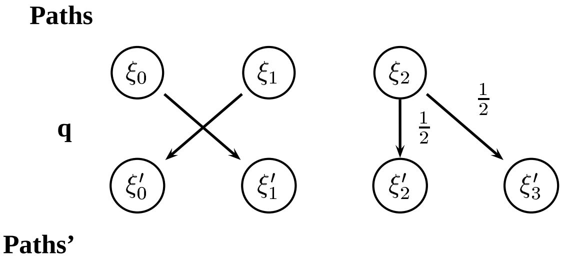
This implies that (since G is positive definite and hence invertible), giving us:

$$\delta\theta \propto G^{-1} \nabla L(\theta) \tag{4}$$

That is, the direction of steepest descent is simply the normal gradient times the inverse of the metric evaluated at the point of tangency. We call this the “natural gradient” for the Reimannian manifold. [Amari and Nagaoka, 2000]

3 Invariance and Chentsov’s Theorem

Some confusion seems to surround the choice of metric on probability manifolds. There is *no* unique answer for this metric, even under the supposition of parametric invariance as suggested by some authors. The issue is rather more subtle. Any metric that transforms under parameter change according to the Jacobian of the function connecting parameters meets this requirement. This type of parametric covariance is a minimum requirement for us. It is natural to suggest a metric that preserves essential probabilistic aspects of the manifold. Consider, for example, functions q (Markov mappings, congruent embeddings, or sufficient statistics, depending on your viewpoint) that carry one distribution over paths to another in a natural and recoverable way. For example, consider the mapping between the two distributions $P(\text{paths})$ and $P(\text{paths}')$given by congruent embedding q depicted below:



The mapping q above interchanges the role of two paths and splits one path into two (with probability 1/2 for each split). In a probabilistically fundamental way, the manifolds $P(\text{path})$ and $P(\text{path}')$ (where we imagine P(path) is a smoothly parameterized set of distributions) are similar to each other. For each path$'$ we can uniquely recover the original probability. In this way, a parameterized distribution over paths can be embedded into a different path space. (This equivalence can actually be phrased in a category theoretic way where the morphisms are these congruent embeddings.) [Chentsov, 1980] General congruent embeddings can be thought of as simply generalizations of the example depicted above to allow arbitrary permutations and arbitrary probabilities for different paths stemming from a single path, as well as compositions of these two. In the control problem this could arise by simple permutation of the state variables or change of co-ordinates to ones with redundant information. It is natural to require that with our metric the congruent embedding is an isometry on the tangent space. (i.e. preserves the length of the vectors). That is, if we make a change of size ϵ (under the metric G_θ) to the distribution $P_\theta(\text{paths})$then carrying that change through q we should measure the same change ϵ using $G_{q(P_\theta)}$. Adding the requirement of invariance with respect to congruent embeddings leads to a unique (up to scale) metric on the manifold. [Chentsov, 1980] This metric is well-known in statistical inference as the Fisher-Rao metric and it can be written as the Fisher information matrix [DeGroot, 1970]:

$$G = \langle \partial_i \mathrm{log} p(\xi;\theta) \partial_j \mathrm{log} p(\xi;\theta) \rangle_{p(\xi;\theta)}$$

Another way to derive the metric is to think about "distances" on probability spaces. The KL-divergence (or relative entropy) between two distributions is a natural divergence on changes in a distribution. It is also manifestly invariant to re-parameterization. If we think about derivatives as small changes in parameters (differentials) then we discover that we also get a unique metric as the second-order Taylor expansion of the KL-divergence. This too agrees with the Fisher information (up to scale). We note that the direction of the KL-divergence is irrelevant as to second-order $\mathrm{KL}(p,q) = \mathrm{KL}(q,p)$ [Amari and Nagaoka, 2000].

4 Fisher-Rao Metric on the Path-space Manifold

The issue now becomes how to derive the Fisher metric on the space of path distributions. It turns out in the case of processes with underlying Markovian state to be rather easy, and involves only computations we already make in the likelihood ratio approach standard in gradient-based reinforcement learning.

4.1 Derivation of finite-time path metric

The Fisher information metric involves computing $\langle \partial_i \mathrm{log} p(\xi;\theta) \partial_j \mathrm{log} p(\xi;\theta) \rangle_{p(\xi;\theta)}$. Fortunately, this is easy to do. The essential algorithm within gradient methods like REINFORCE and GPOMDP is a clever method of computing $\langle \partial_i \mathrm{log} p(\xi;\theta) \rangle_{p(\xi;\theta)}$, the expected score function. Thus, while the expected score is the gradient, the correlation of the score is the Fisher matrix. The following simple algorithm is unbiased and converges almost surely (under the same regularity conditions as in [Baxter *et al.*, 1999] as the number of sample paths m goes to infinity:

Algorithm 1 (Finite-horizon metric computation) *For i in 1 to m:*

```
• Sample a path using the policy $\pi(\theta)$: $\xi_i = (x_1, a_1, x_2, a_2, \ldots, x_n)$
• For t in 1 to $n$:
    – Compute $z_j = \frac{t-1}{t} z + \frac{1}{t} \partial_j log \pi(a|x;\theta)$
• $G = \frac{i-1}{i} G + \frac{1}{i} z^T z$
Return G
```

We use the Markov property and the invariance of the transition probabilities (given the actions) in the algorithm above to compute $\partial_i \mathrm{log} p(\xi;\theta)$ simply. These details can be extracted from the proofs below, as well as other, potentially better ways to sample.

To compute the natural gradient, we simply invert this matrix and multiply it with the gradient computed using standard policy search methods.

4.2 Limiting metric for infinite horizon problems

A well-known result in statistics [DeGroot, 1970] gives a different form for the Fisher metric under appropriate regularity conditions. We quickly derive that result, and then show how this gives us a simple form for the path probability metric as $t \to \infty$.

$$\begin{aligned} G_\theta &= \langle \partial_i \mathrm{log} p(\xi;\theta) \partial_j \mathrm{log} p(\xi;\theta) \rangle_{p(\xi;\theta)} \\ &= \sum_{\Xi} \partial_i p(\xi;\theta) \partial_j (\mathrm{log} p(\xi;\theta)) \\ &= \sum_{\Xi} \partial_i (p(\xi;\theta) \partial_j log p(\xi;\theta)) - \\ &\quad \sum_{\Xi} p(\xi;\theta) \partial_i \partial_j \mathrm{log} p(\xi;\theta) \\ &= -\langle \partial_i \partial_j \mathrm{log} p(\xi;\theta) \rangle_{p(\xi;\theta)} + \partial_i \sum_{\Xi} \partial_j p(\xi;\theta) \\ &= -\langle \partial_i \partial_j \mathrm{log} p(\xi;\theta) \rangle_{p(\xi;\theta)} \qquad (5) \end{aligned}$$

The third line follows from the second by integrating by parts and the fifth follows by observing that the total probability is constant.

Now we show that in the limit this leads to a simple metric for the infinite horizon case. To get a convergent metric, we must normalize the metric, in particular by the total length of the path denoted t. Since the metric is defined only up to scale, this is perfectly justified.

Theorem 1 (Infinite-Horizon Metric) *For an ergodic Markov process the Fisher information matrix limits to the expected Fisher information of the policy for each state and control under stationary distribution of states and actions. Proof: We use G^t_θ to indicate the t-step finite-horizon metric.*

$$\begin{aligned} \lim_{t\to\infty} \frac{1}{t} G^t_\theta &= \lim_{t\to\infty} -\frac{1}{t} \langle \partial_i \partial_j log p(\xi;\theta) \rangle_{p(\xi;\theta)} \\ &= \lim_{t\to\infty} -\frac{1}{t} \sum_{\Xi} p(\xi;\theta) \partial_i \frac{\partial_j p(\xi;\theta)}{p(\xi;\theta)} \qquad (6) \end{aligned}$$

For a Markov process we can write the likelihood ratio $\frac{\partial_i p(\xi;\theta)}{p(\xi;\theta)}$ as

$$\frac{\partial_i p(\xi;\theta)}{p(\xi;\theta)} = \sum\nolimits_t \frac{\partial_i p(x_{t+1}|x_t;\theta)}{p(x_{t+1}|x_t;\theta)}$$

Using the chain rule we continue the derivation with $d^{\pi}(x)$ indicating the stationary distribution:

$$\begin{aligned}
\lim_{t\to\infty}\frac{1}{t}G_{\theta} &= \lim_{t\to\infty} -\frac{1}{t}\langle\sum_{t}\frac{\partial_i\partial_j p(x_{t+1}|x_t;\theta)}{p(x_{t+1}|x_t;\theta)} - \\
&\quad \frac{\partial_i p(x_{t+1}|x_t;\theta)\partial_j p(x_{t+1}|x_t;\theta)}{p(x_{t+1}|x_t;\theta)p(x_{t+1}|x_t;\theta)}\rangle_{p(\xi;\theta)} \\
&= -\sum_{x,a} d^{\pi}(x)\pi(a|x;\theta)(\frac{\partial_i\partial_j\pi(a_t|x_t;\theta)}{\pi(a_t|x_t;\theta)}) + \\
&\quad \frac{\partial_i\pi(a_t|x_t;\theta)\partial_j\pi(a_t|x_t;\theta)}{\pi(a_t|x_t;\theta)\pi(a_t|x_t;\theta)}) \\
&= \sum_{x,a} d^{\pi}(x)\pi(a|x;\theta)\frac{\partial_i\pi(a_t|x_t;\theta)\partial_j\pi(a_t|x_t;\theta)}{\pi(a_t|x_t;\theta)^2} \\
&\quad -\sum_{x} d^{\pi}(x)\sum_{a}\partial_i\partial_j\pi(a_t|x_t;\theta) \\
&= \langle G_{x;\theta}\rangle_{d^{\pi}(x)}
\end{aligned}$$

where the second equality follows by noting that the likelihood ratio is independent of transition probability matrix given the action probability, and by application of the ergodic theorem. The last line follows (with $G_{x;\theta}$ the Fisher information for the policy at a given state) as the second term in line 3 vanishes since the total probability is constant with respect to θ. ■

It is also interesting to consider what happens in the start-state, discounted formulation of the problem. In this case we would like our metric, using the general definition over path distributions given above, to naturally weigh the start start more than it necessarily is in the infinite horizon average case. It is well-known that a discount factor γ is equivalent to an undiscounted problem where each trajectory terminates with probability $1-\gamma$ at each step. We can use this fact to derive a metric more appropriate for the discounted formalism.

Theorem 2 (Start-State Metric) *For a discounted Markov process the Fisher information matrix equals the Fisher information of the policy for each state and control under the limiting distribution of states and actions.* *Proof: The proof is very similiar to the infinite horizon case and so we simply sketch it:*

$$\begin{aligned}
G_{\theta} &= -\langle\partial_i logp(\xi;\theta)\partial_j logp(\xi;\theta)\rangle_{p(\xi;\theta)} \\
&= -\sum_{\Xi} p(\xi;\theta)\partial_i\frac{\partial_j p(\xi;\theta)}{p(\xi;\theta)} \\
&= -\sum_{\Xi} p(\xi;\theta)\partial_i\sum_{t}\frac{\partial_j\pi(a_t|x_t)}{\pi(a_t|x_t)} \\
&= \sum_{\Xi} p(\xi;\theta)\sum_{t}\frac{\partial_i\pi(a_t|x_t)\partial_j\pi(a_t|x_t)}{\pi(a_t|x_t)^2} - \\
&\quad \sum_{\Xi} p(\xi;\theta)\sum_{t}\frac{\partial_i\partial_j\pi(a_t|x_t)}{\pi(a_t|x_t)} \\
&= \langle\frac{\partial_i\partial_j\pi(a|x)}{\pi(a|x)}\rangle_{d^{\pi}(x)\pi(a_t|x_t)}
\end{aligned}$$

■

where $d^{\pi}(x)=\sum_t \gamma^t Pr^{\pi}(x_t=x)$ is the limiting distribution of states. Thus the infinite horizon (undiscounted) and start-state metrics give essentially the same metric, with only the effective weighting by states differing.

4.3 Metrics for Partially Observed Problems

For policies that map the observation space of a partially-observed markov decision process into distributions over actions, it is just as easy to derive appropriate metric using our approach. The tuple (x_t, a_t, y_t) is also a markov chain, and with only subtle changes to the arguments above we end up with the same metric except using the limiting distribution of observations instead of states.

5 Relation to Compatible Value Function Actor Critic

Kakade noticed a fascinating connection between the limiting metric given in Theorem 1 and the improvement direction computed by a class of actor-critic methods that use a special compatible function approximation technique.[Sutton *et al.*, 1999; Konda and Tsitsiklis, 2002] Following Kakade we let $\psi(s,a)=\partial_i \text{log} p(a|x;\theta)$ and let the compatible function approximator be linear in ψ:

$$f(x,a;\omega)=\sum\nolimits_i \omega_i\psi_i$$

This type of value function approximator was initially suggested as it may be used to compute the true gradient. In practice, it has not been clear what advantage this brings over the gradient estimation routines of [Baxter *et al.*, 1999]. However, "folk-wisdom" has it that performing an that infinitesimal policy iteration (moving in the direction of the best policy according to $f^{\pi}(x,a;\omega)$) using this approximation has very good properties and often significantly outperforms standard gradient methods. The natural gradient provides insight into this behavior. Let ω minimize the squared value-function error:

$$\epsilon(\omega;\theta)=\sum\nolimits_{x,a} d^{\pi}(x)\pi(a|x)(f(x,a;\theta)-Q(x,a;\theta))^2$$

where $Q(x,a;\theta)$ is the exact advantage function. [Sutton *et al.*, 1999] It is easy to check that ([Kakade, 2002], Theorem 1) $\omega \propto {G_{\theta}}^{-1}\partial_i J(\theta)$. That is, the direction of maximum policy improvement is exactly the natural gradient direction. This can be seen by simply differentiating $\epsilon(\omega;\theta)$ to minimize it with respect to ω and noting the result of [Sutton *et al.*, 1999] that

$$\partial_i J(\theta)=\sum\nolimits_{x,a} d^{\pi}(x)Q(x,a;\theta)\partial_i p(a|x;\theta)$$

6 Demonstration

As a consequence of the results of section 5 and [Kakade, 2002], experimental results already exist demonstrating the effectiveness of the natural gradient method. That is, all results using policy improvement with compatible function approximators are implicitly computing this result. As a demonstration, we computed analytically the natural gradient for the

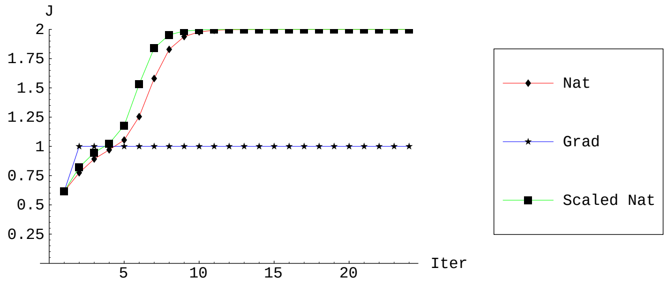


Figure 4: Performance comparison of covariant (nat and scaled) and non-covariant (grad) on the 2-state MDP. The covariant learning algorithm dramatically outperforms the standard gradient algorithm. The standard method achieves J=2 after more than 1000 iterations.

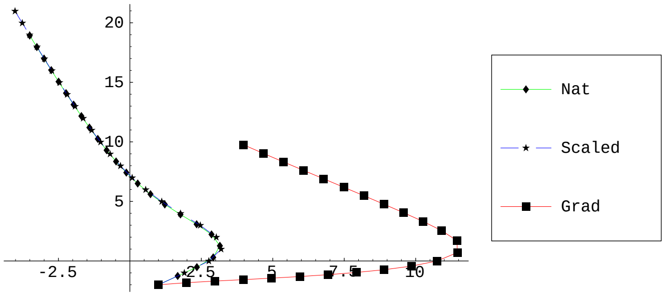


Figure 5: Path through policy space (showing log-odds of the actions with state 0 along the horizontal axis) of covariant (nat and scaled) and non-covariant (grad) algorithms on the 2-state MDP. Note that, as required, the path is independent of scaling for the covariant algorithm.

problem given in Section 2.2. This problem is interesting as it reveals some of the properties of the natural gradient method. First it is easily checked that for a complete Boltzmann policy in an MDP the natural gradient computes:

$$G^{-1}(\theta)\partial_i J(\theta) = \sum\nolimits_{x,a} Q(x,a;\theta)\partial_i p(a|x;\theta)$$

This means that it is very similiar to the standard gradient except that it removes the weighting of states by the stationary distribution. Rather, each state is treated equally. This leads to much more reasonable results in the problem discussed as the partial derivative component for state 1 does not shrink as the policy changes initially.

In Figure 4 we plot the performance of the natural gradient (using two different scalings) and the standard gradient methods in optimizing the policy for this problem.

It is interesting to note that in this graph the behavior of the natural gradient descent algorithm appears to be non-covariant. This is simply due to the step size heuristic not computing equivalent steps in the policy space. The direction however *is* constant as illustrated in Figure 5.

7 Conclusions

[Kakade, 2002] suggested that covariant behavior of gradient algorithms is an important desiderata in reinforcement learning. Unfortunately, working in the space of policies, it was difficult to generate such an algorithm. Here we proposed considering the induced path-distribution manifold and used notions from information geometry to propose a natural covariant algorithm. This leads to interesting insight and a practical algorithm. Fortunately, it agrees with the heuristic suggested by Kakade (despite the suggestion in the paper that the algorithm there was actually not covariant) in the infinite horizon case and extends to cover new problems. [Peters *et al.*, 2002] independently developed theory related to ours (in particular the theorems in Section 4.2) and presented results in the context of robot dynamic control.

Further work will provide more investigation into the experimental behavior of the algorithms presented. Future effort may also yield deeper insight into the relationship between the method presented here and value-function approximations.

References

[Amari and Nagaoka, 2000] S. Amari and H. Nagaoka. *Methods of Information Geometry*. Oxford University Press, 2000.

[Bagnell and Schneider, 2001] J. Bagnell and J. Schneider. Autonomous helicopter control by policy-search based reinforcement learning. In *Proceedings of the 2001 IEEE Int. Conference on Robotics and Automation*. IEEE, 2001.

[Baxter *et al.*, 1999] J. Baxter, L. Weaver, and P. Bartlett. Direct-gradient-based reinforcement learning I: Gradient estimation algorithms. Technical report, Computer Sciences Lab, Australian National University, 1999.

[Bertsekas and Tsitsiklis, 1996] D. Bertsekas and J. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[Chentsov, 1980] N.N. Chentsov. *Statistical Decision Rules and Optimal Inference*. American Mathematical Society, 1980.

[DeGroot, 1970] M. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill, 1970.

[Kakade, 2002] S. Kakade. A natural policy gradient. In *Advances in Neural Information Processing Systems (NIPS-14)*, 2002.

[Konda and Tsitsiklis, 2002] V. Konda and J. Tsitsiklis. Actor-critic algorithms. *to appear in the SIAM Journal on Control and Optimization*, 2002.

[Peters *et al.*, 2002] Jan Peters, Sethu Vijaykumar, and Stefan Schaal. Policy gradient methods for robot control. Technical Report 00-737, USC, 2002.

[Sutton *et al.*, 1999] R. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Neural Information Processing Systems 12*, 1999.