Journal of Machine Learning Research 7 (2006) 771–791 Submitted 5/05; Published 5/06

Policy Gradient in Continuous Time

Rémi Munos REMI.MUNOS@POLYTECHNIQUE.FR
Centre de Mathématiques Appliquées
Ecole Polytechnique
91128 Palaiseau, France

Editor: Michael Littman

Abstract

Policy search is a method for approximately solving an optimal control problem by performing a parametric optimization search in a given class of parameterized policies. In order to process a local optimization technique, such as a gradient method, we wish to evaluate the sensitivity of the performance measure with respect to the policy parameters, the so-called *policy gradient*. This paper is concerned with the estimation of the policy gradient for continuous-time, deterministic state dynamics, in a *reinforcement learning* framework, that is, when the decision maker does not have a model of the state dynamics.

We show that usual likelihood ratio methods used in discrete-time, fail to proceed the gradient because they are subject to variance explosion when the discretization time-step decreases to 0. We describe an alternative approach based on the approximation of the pathwise derivative, which leads to a policy gradient estimate that converges almost surely to the true gradient when the time-step tends to 0. The underlying idea starts with the derivation of an explicit representation of the policy gradient using pathwise derivation. This derivation makes use of the knowledge of the state dynamics. Then, in order to estimate the gradient from the observable data only, we use a stochastic policy to discretize the continuous deterministic system into a stochastic discrete process, which enables to replace the unknown coefficients by quantities that solely depend on known data. We prove the almost sure convergence of this estimate to the true policy gradient when the discretization time-step goes to zero.

The method is illustrated on two target problems, in discrete and continuous control spaces.

Keywords: optimal control, reinforcement learning, policy search, sensitivity analysis, parametric optimization, gradient estimate, likelihood ratio method, pathwise derivation

1. Introduction and Statement of the Problem

We consider an optimal control problem with continuous state $(x_t \in \mathbb{R}^d)_{t\geq 0}$ whose state dynamics is defined according to the controlled differential equation:

$$\frac{dx_t}{dt} = f(x_t, u_t), \tag{1}$$

where the control $(u_t)_{t\geq 0}$ is a Lebesgue measurable function with values in a control space U. Note that the state-dynamics f may also depend on time, but we omit this dependency in the notation, for simplicity. We intend to maximize a functional J that depends on the trajectory $(x_t)_{0\leq t\leq T}$ over a finite-time horizon $T > 0$. For simplicity, in the paper, we illustrate the case of a terminal reward

only:

$$J(x;(u_t)_{t\geq 0}) := r(x_T), \tag{2}$$

where $r : \mathbb{R}^d \to \mathbb{R}$ is the reward function. Extension to the case of general functional of the kind

$$J(x;(u_t)_{t\geq 0}) = \int_0^T r(t,x_t)dt + R(x_T), \tag{3}$$

with r and R being current and terminal reward functions, would easily follow, as indicated in Remark 1.

The optimal control problem of finding a control $(u_t)_{t\geq 0}$ that maximizes the functional is replaced by a parametric optimization problem for which we search for a good feed-back control law in a given class of parameterized policies $\{\pi_\alpha : [0,T] \times \mathbb{R}^d \to U\}_\alpha$, where $\alpha \in \mathbb{R}^m$ is the parameter. The control $u_t \in U$ (or action) at time t is $u_t = \pi_\alpha(t,x_t)$, and we may write the dynamics of the resulting feed-back system as

$$\frac{dx_t}{dt} = f_\alpha(x_t), \tag{4}$$

where $f_\alpha(x_t) := f(x,\pi_\alpha(t,x))$. In the paper, we will make the assumption that f_α is $\mathcal{C}^2$, with bounded derivatives. Let us define the **performance measure**

$$V(\alpha) := J(x;\pi_\alpha(t,x_t)_{t\geq 0}),$$

where its dependency with respect to (w.r.t.) the parameter α is emphasized. One may also consider an average performance measure according to some distribution μ for the initial state: $V(\alpha) := \mathbb{E}[J(x;\pi_\alpha(t,x_t)_{t\geq 0})|x \sim \mu]$.

In order to find a local maximum of $V(\alpha)$, one may perform a local search, such as a gradient ascent method

$$\alpha \leftarrow \alpha + \eta \nabla_\alpha V(\alpha), \tag{5}$$

with an adequate step η (see for example (Polyak, 1987; Kushner and Yin, 1997)). The computation of the gradient $\nabla_\alpha V(\alpha)$ is the object of this paper.

A first method would be to approximate the gradient by a finite-difference quotient for each of the m components of the parameter:

$$\partial_{\alpha_i} V(\alpha) \simeq \frac{V(\alpha+\varepsilon e_i) - V(\alpha)}{\varepsilon},$$

for some small value of ε (we use the notation ∂_α instead of ∇_α to indicate that it is a single-dimensional derivative). This finite-difference method requires the simulation of $m+1$ trajectories to compute an approximation of the true gradient. When the number of parameters is large, this may be computationally expensive. However, this simple method may be efficient if the number of parameters is relatively small.

In the rest of the paper we will not consider this approach, and will aim at computing the gradient using one trajectory only.

Pathwise estimation of the gradient. We now illustrate that if the decision-maker has access to a model of the state dynamics, then a pathwise derivation would directly lead to the policy gradient. Indeed, let us define the gradient of the state with respect to the parameter: $z_t := \nabla_\alpha x_t$ (i.e. z_t is defined as a $d \times m$-matrix whose (i,j)-component is the derivative of the ith component of x_t w.r.t. α_j). Our smoothness assumption on f_α allows to differentiate the state dynamics (4) w.r.t. α, which provides the dynamics on (z_t):

$$\frac{dz_t}{dt} = \nabla_\alpha f_\alpha(x_t) + \nabla_x f_\alpha(x_t) z_t, \tag{6}$$

where the coefficients $\nabla_\alpha f_\alpha$ and $\nabla_x f_\alpha$ are, respectively, the derivatives of f w.r.t. the parameter (matrix of size $d \times m$) and the state (matrix of size $d \times d$). The initial condition for z is $z_0 = 0$. When the reward function r is smooth (i.e. continuously differentiable), one may apply a pathwise differentiation to derive a gradient formula (see e.g. (Bensoussan, 1988) or (Yang and Kushner, 1991) for an extension to the stochastic case):

$$\nabla_\alpha V(\alpha) = \nabla_x r(x_T) z_T. \tag{7}$$

Remark 1 *In the more general setting of a functional (3), the gradient is deduced (by linearity) from the above formula:*

$$\nabla_\alpha V(\alpha) = \int_0^T \nabla_x r(t, x_t) z_t \, dt + \nabla_x R(x_T) z_T.$$

What is known from the agent? The decision maker (call it the agent) that intends to design a good controller for the dynamical system may or may not know a model of the state dynamics f. In case the dynamics is known, the state gradient $z_t = \nabla_\alpha x_t$ may be computed from (6) along the trajectory and the gradient of the performance measure w.r.t. the parameter α is deduced at time T from (7), which allows to perform the gradient ascent step (5).

However, in this paper we consider a *Reinforcement Learning* (Sutton and Barto, 1998) setting in which the state dynamics is unknown from the agent, but we still assume that the state is fully observable. The agent knows only the response of the system to its control. To be more precise, the available information to the agent at time t is its own control policy π_α and the trajectory $(x_s)_{0 \le s \le t}$ up to time t. At time T, the agent receives the reward $r(x_T)$ and, in this paper, we assume that the gradient $\nabla r(x_T)$ is available to the agent.

From this point of view, it seems impossible to derive the state gradient z_t from (6), since $\nabla_\alpha f$ and $\nabla_x f$ are unknown. The term $\nabla_x f(x_t)$ may be approximated by a least squares method from the observation of past states $(x_s)_{s \le t}$, as this will be explained later on in subsection 3.2. However the term $\nabla_\alpha f(x_t)$ cannot be calculated analogously.

In this paper, we introduce the idea of using stochastic policies to approximate the state (x_t) and the state gradient (z_t) by discrete-time stochastic processes (X_t^Δ) and (Z_t^Δ) (with Δ being some discretization time-step). We show how Z_t^Δ can be computed without the knowledge of $\nabla_\alpha f$, but only from information available to the agent.

We prove the convergence (with probability one) of the gradient estimate $\nabla_x r(X_T^\Delta) Z_T^\Delta$ derived from the stochastic processes to $\nabla_\alpha V(\alpha)$ when $\Delta \to 0$. Here, almost sure convergence is obtained using the *concentration of measure phenomenon* (Talagrand, 1996; Ledoux, 2001).

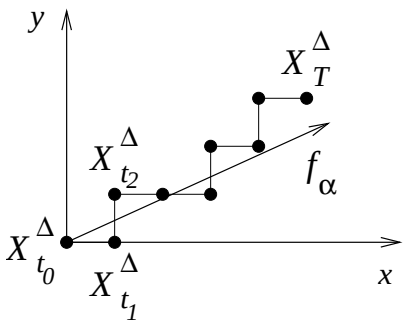


Figure 1: A trajectory $(X^{\Delta}_{t_n})_{0\leq n\leq N}$ and the state dynamics vector f_α of the continuous process $(x_t)_{0\leq t\leq T}$.

Likelihood ratio method? It is worth mentioning that this strong convergence result contrasts with the usual *likelihood ratio method* (also called *score method*) in discrete time (see e.g. (Reiman and Weiss, 1986; Glynn, 1987) or more recently in the reinforcement learning literature (Williams, 1992; Sutton et al., 2000; Baxter and Bartlett, 2001; Marbach and Tsitsiklis, 2003)) for which the policy gradient estimate is subject to variance explosion when the discretization time-step Δ tends to 0. The intuitive reason for that problem lies in the fact that the number of decisions before getting the reward grows to infinity when $\Delta \to 0$ (the variance of likelihood ratio estimates being usually linear with the number of decisions).

Let us illustrate this problem on a simple 2 dimensional process. Consider the deterministic continuous process $(x_t)_{0\leq t\leq 1}$ defined by the state dynamics:

$$\frac{dx_t}{dt} = f_\alpha := \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix}, \tag{8}$$

$(0 < \alpha < 1)$ with initial condition $x_0 = (0\,0)'$ (where $'$ denotes the transpose operator). The performance measure $V(\alpha)$ is the reward at the terminal state at time $T = 1$, with the reward function being the first coordinate of the state $r((xy)') := x$. Thus $V(\alpha) = r(x_{T=1}) = \alpha$ and its derivative is $\nabla_\alpha V(\alpha) = 1$.

Let $(X^{\Delta}_{t_n})_{0\leq n\leq N} \in \mathbb{R}^2$ be a discrete time stochastic process (the discrete times being $\{t_n = n\Delta\}_{n=0\ldots N}$ with the discretization time-step $\Delta = 1/N$) that starts from initial state $X^{\Delta}_0 = x_0 = (0\,0)'$ and makes N random moves of length Δ towards the right (action u_1) or the top (action u_2) (see Figure 1) according to the stochastic policy (i.e., the probability of choosing the actions in each state x) $\pi_\alpha(u_1|x) = \alpha$, $\pi_\alpha(u_2|x) = 1-\alpha$.

The process is thus defined according to the dynamics:

$$X^{\Delta}_{t_{n+1}} = X^{\Delta}_{t_n} + \begin{pmatrix} U_n \\ 1-U_n \end{pmatrix} \Delta, \tag{9}$$

where $(U_n)_{0\leq n<N}$ are N independent Bernoulli random variables that equal 1 with probability α and 0 with probability $1-\alpha$. The stochastic discrete process (X^{Δ}_t) is consistent with the deterministic continuous one (x_t) in the sense that the jump average direction of the former equals the state

dynamics vector of the latter:

$$\mathbb{E}\Big[\frac{X_{t_{n+1}}-X_{t_n}}{\Delta}|X_{t_n}=x\Big]=\pi_\alpha(u_1,x)\begin{pmatrix}1\\0\end{pmatrix}+\pi_\alpha(u_2,x)\begin{pmatrix}0\\1\end{pmatrix}=\begin{pmatrix}\alpha\\1-\alpha\end{pmatrix}.$$

Thus, when the discretization time-step Δ tends to 0, the process (X_t^Δ) converges almost surely to (x_t) (this statement will be proved in Section 2).

Now, write $V^\Delta(\alpha)$ the performance measure of the discrete process, taken as the expected reward at the terminal state: $V^\Delta(\alpha):=\mathbb{E}[r(X_1^\Delta)]=\frac{1}{N}\sum_{n=0}^{N-1}U_n$. The likelihood ratio estimate $g(\Delta)$ of the gradient $\nabla_\alpha V^\Delta(\alpha)=\mathbb{E}[g(\Delta)]$ is

$$\begin{aligned}g(\Delta) &= r(X_1^\Delta)\sum_{n=0}^{N-1}\frac{\nabla_\alpha\pi_\alpha(u_{t_n}|X_{t_n}^\Delta)}{\pi_\alpha(u_{t_n}|X_{t_n}^\Delta)}\\ &= \Big(\frac{1}{N}\sum_{n=0}^{N-1}U_n\Big)\sum_{n=0}^{N-1}\Big(\frac{U_n}{\alpha}-\frac{1-U_n}{1-\alpha}\Big).\end{aligned}\tag{10}$$

The expectation and variance of this estimate are given now (a proof is provided in Appendix A).

Proposition 2 *The expectation and variance of the estimate (10) are*

$$\begin{aligned}\mathbb{E}\big[g(\Delta)\big] &= 1,\\ Var\big[g(\Delta)\big] &= \frac{1-5(1-\alpha)+(2-3\alpha)\alpha N+\alpha^2N^2}{\alpha(1-\alpha)N}.\end{aligned}\tag{11}$$

Thus $g(\Delta)$ is an unbiased estimated of the true gradient $\nabla_\alpha V(\alpha)=1$. However we notice that the dominant term (when N is large) of the variance is $\frac{\alpha}{1-\alpha}N$, with N being the number of decisions before getting the reward, which grows to infinity when the discretization time-step $\Delta=1/N$ tends to 0. Therefore it is impossible to use this likelihood ratio estimate whenever the time discretization is too fine. In contrast, the gradient estimate introduced in this paper has a variance that decreases to 0 when Δ tends to 0 (this will be illustrated on this same example in subsection 3.4).

Outline of the paper. The paper is organized as follows: in Section 2, we state a general approximation result of a continuous deterministic process by a consistent stochastic discrete process and apply it to prove the convergence of the discretized state and state gradient processes when using a stochastic policy. In Section 3, we establish the convergence of the policy gradient estimate and describe a reinforcement learning algorithm that replaces the unknown coefficients about the state dynamics by information available to the agent. In the last Section, we illustrate the method on two (6 dimensional) target problems in both a discrete and a continuous control space cases. All proofs are in the Appendices.

2. Discretized Stochastic Processes

In this section, we start with a general result for approximating a deterministic continuous process by a stochastic discrete one. This is subsequently applied to the convergence analysis of processes (the state X_t^Δ and the state gradient Z_t^Δ) related to the introduction of stochastic policies.

2.1 A General Convergence Result

Let $(x_t)_{0\leq t\leq T}$ be a deterministic continuous process defined by some dynamics

$$\frac{dx_t}{dt} = f(x_t)$$

with some initial condition x_0. We assume that f is of class $\mathcal{C}^2$ with bounded derivatives. The following result state the almost sure convergence of a consistent discrete stochastic process.

Theorem 3 *Let $\Delta = T/N$ be a discretization time-step (with N being the number of steps) and write $\{t_n = n\Delta\}_{0\leq n\leq N}$ the discrete times. Let $(U_{t_n})_{0\leq n<N}$ be a sequence of independent random variables with values in a set U. We define a discrete stochastic process $(X^\Delta_{t_n})_{0\leq n\leq N}$, starting at $X^\Delta_0 = x_0$, according to some discrete state dynamics $f^\Delta : \mathbb{R}^d \times U \to \mathbb{R}^d$, assumed to be bounded: for $t \in \{t_n\}_{0\leq n<N}$,*

$$X^\Delta_{t+\Delta} = X^\Delta_t + f^\Delta(X^\Delta_t, U_t). \tag{12}$$

If f^Δ satisfies the consistency property*:*

$$\mathbb{E}[f^\Delta(x, U_t)] = f(x)\Delta + o(\Delta), \tag{13}$$

and the following bounding condition:

$$f^\Delta = O(\Delta), \tag{14}$$

(where the notation $O(\cdot)$ is to be understood in the sense uniformly w.r.t. the variable of f^Δ) then, the random variable X^Δ_T converges almost surely to (the deterministic) x_T when $\Delta \to 0$. We write

$$\lim_{\Delta\to 0} X^\Delta_T = x_T, \text{ with probability } 1.$$

Appendix B gives a proof of this result. Note that a weaker convergence result (i.e. convergence in probability) may be obtained from general results in approximation of diffusion processes by Markov chains (Kloeden and Platen, 1995). Here, almost sure convergence is obtained using the *concentration of measure phenomenon* (Talagrand, 1996; Ledoux, 2001), detailed in Appendix B.

Remark 4 *If we assume a slightly better consistency error of $O(\Delta^2)$ instead of $o(\Delta)$ in (13), then we may prove (straightforwardly from the Appendix) that $\mathbb{E}[X^\Delta_T] = x_T + O(\Delta)$ and $\mathbb{E}[||X^\Delta_T - x_T||^2] = O(\Delta)$.*

2.2 Discretization of the State

Let us go back to our initial control problem (1). We consider the case of a finite control space U (extension to a continuous control space is straightforward and is detailed in subsection 3.5). Let π_α be a **stochastic policy** , i.e. $\pi_\alpha(u|t,x)$ denotes the probability of choosing action $u \in U$ at time t in state x. We write $u \sim \pi_\alpha(\cdot|t,x)$ a random choice of an action u according to such a policy.

Now, we define the **stochastic discrete state process** $(\mathbf{X^\Delta_{t_n}})_{\mathbf{0\leq n\leq N}}$ (where we use the same notations for the time-steps (t_n) as in the previous subsection), starting at a state $X^\Delta_0 = x$, as follows:

At time $t \in \{(t_n)_{0\leq n<N}\}$, we select an action $u_t \sim \pi_\alpha(\cdot|t, X^\Delta_t)$. Then, $X^\Delta_{t+\Delta}$ is the state at time $t+\Delta$ resulting from keeping the action u_t constant for a period of time Δ. We write:

$$\left\{ \begin{array}{rcl} u_t & \sim & \pi_\alpha(\cdot|t, X^\Delta_t) \\ X^\Delta_{t+\Delta} & := & X^\Delta_t + f^\Delta(X^\Delta_t, u_t) \end{array} \right. \tag{15}$$

where $f^\Delta(x,u)$ represents the jump in the state resulting from the state dynamics (1) with initial condition $x_0 = x$, using a constant control u for a period of time Δ.

The next proposition states the convergence of the discrete stochastic process (X_t^Δ) to the continuous deterministic one (x_t).

Proposition 5 ***Convergence of the discrete state process*** $(\mathbf{X_t^\Delta})$. *When the discretization time-step $\Delta \to 0$, the random variable X_T^Δ converges almost surely to the state x_T defined according to the state dynamics (4) with*

$$f_\alpha(x) := \sum_{u \in U} \pi_\alpha(u|t,x) f(x,u).$$

and initial condition $x_0 = x$.

Proof This is an immediate consequence of Theorem 3 with the discrete state dynamics $f^\Delta(x,u)$. From Taylor's formula,

$$f^\Delta(x,u_t) = f(x,u_t)\Delta + O(\Delta^2),$$

to derive the property on the average jumps:

$$\mathbb{E}[f^\Delta(x,u_t)] \;=\; \sum_{u \in U} \pi_\alpha(u|t,x) f(x,u)\Delta + O(\Delta^2) = f_\alpha(x)\Delta + O(\Delta^2),$$

and the consistency conditions (13) holds, as well as the bound on the jumps (14). ■

2.3 Discretization of the State Gradient

Now, we build an approximation of the state gradient $z_t = \nabla_\alpha x_t$. We define the **stochastic discrete state gradient process** $(\mathbf{Z_{t_n}^\Delta})_{\mathbf{0 \le n \le N}}$, starting with $Z_0^\Delta = 0$, as follows:

At time $t \in \{(t_n)_{0 \le n < N}\}$, let (u_t) and (X_t^Δ) be defined according to (15). Then define

$$Z_{t+\Delta}^\Delta := Z_t^\Delta + f(X_t^\Delta, u_t)\big[l_\alpha(t, X_t^\Delta, u_t)' + l_x(t, X_t^\Delta, u_t)' Z_t^\Delta\big]\Delta + \nabla_x f(X_t^\Delta, u_t) Z_t^\Delta \Delta, \tag{16}$$

where

$$l_\alpha(t,x,u) := \frac{\nabla_\alpha \pi_\alpha(u|t,x)}{\pi_\alpha(u|t,x)} \text{ and } l_x(t,x,u) := \frac{\nabla_x \pi_\alpha(u|t,x)}{\pi_\alpha(u|t,x)}$$

are the likelihood ratios of π_α w.r.t. α and x (defined as vectors of size m and d respectively).

Proposition 6 ***Convergence of the discrete state gradient process*** $(\mathbf{Z_T^\Delta})$***:***

The random variable Z_T^Δ converges almost surely to z_T when $\Delta \to 0$.

Proof The discrete state dynamics (12) for (Z_t^Δ) is defined by the right hand side of (16). Now, from the property

$$\begin{aligned}\mathbb{E}[Z_{t+\Delta}^\Delta - Z_t^\Delta | X_t^\Delta = x, Z_t^\Delta = z] \;&=\; \sum_{u \in U} \pi_\alpha(u|t,x)\Big\{ f(x,u)[l_\alpha(t,x,u)' + l_x(t,x,u)' z] \\ &\qquad + \nabla_x f(x,u) z \Big\}\Delta \\ &=\; \big[\nabla_\alpha f_\alpha(x) + \nabla_x f_\alpha(x) z\big]\Delta,\end{aligned}$$

we deduce that the coupled process (X_t^Δ, Z_t^Δ) is consistent with (x_t, z_t) in the sense of (13):

$$\mathbb{E}\left[\begin{pmatrix} X_{t+\Delta}^\Delta \\ Z_{t+\Delta}^\Delta \end{pmatrix} - \begin{pmatrix} X_t^\Delta \\ Z_t^\Delta \end{pmatrix} \Big| \begin{pmatrix} X_t^\Delta \\ Z_t^\Delta \end{pmatrix} = \begin{pmatrix} x \\ z \end{pmatrix}\right] = \begin{pmatrix} f_\alpha(x) \\ \nabla_\alpha f_\alpha(x) + \nabla_x f_\alpha(x) z \end{pmatrix} \Delta + o(\Delta) \tag{17}$$

and $X_{t+\Delta}^\Delta - X_t^\Delta = O(\Delta)$ and $Z_{t+\Delta}^\Delta - Z_t^\Delta = O(\Delta)$. Thus, as a consequence of Theorem 3, the random variable Z_T^Δ converges almost surely to z_T when $\Delta \to 0$. ■

3. Model-Free Reinforcement Learning Algorithm

We show how to use the approximation results of the previous section to design a model-free reinforcement learning algorithm for estimating the policy gradient $\nabla_\alpha V(\alpha)$ using one trajectory only. First, we state the convergence of the policy gradient estimate computed from the discretized process, then show how to approximate the unknown coefficient $\nabla_x f$ using least-squares regression from the observed trajectory, and finally describe the reinforcement learning algorithm.

3.1 Convergence of the Policy Gradient Estimate

One may use formula (7) to define a gradient estimate of the performance measure w.r.t. the parameter α based on the discrete process (X_t^Δ, Z_t^Δ):

$$g(\Delta) := \nabla_x r(X_T^\Delta) Z_T^\Delta. \tag{18}$$

This estimate converges almost surely to the true gradient, as stated now.

Proposition 7 *Assume that r is continuously differentiable. Then*

$$\lim_{\Delta \to 0} g(\Delta) = \nabla_\alpha V(\alpha) \text{ with probability } 1.$$

Proof This is a direct consequence of the almost sure convergence of (X_T^Δ, Z_T^Δ) to (x_T, z_T) and the continuity of $\nabla_x r$. ■

Now, let us illustrate how Z_t^Δ may be approximated with information available to the agent. The definition (16) of Z_t^Δ requires the term $\nabla_x f(X_t^\Delta, u)$. We now explain how to built a consistent approximation $\widehat{\nabla_x f}(X_t^\Delta, u)$ of this term from the past of the trajectory $(X_s^\Delta)_{0 \le s \le t}$.

3.2 Least-Squares Approximation of $\nabla_x f(X_t^\Delta, u)$

For clarity, in this subsection, we omit reference to Δ, for example writing X_s instead of X_s^Δ. Write $\Delta X_t = X_{t+\Delta} - X_t$ the jump of the state. Let $c > 0$ be a constant (independent of Δ). Define $S(t) := \{s \in [t - c\Delta, t] \,|\, u_s = u_t\}$ the set of past discrete times $t - c\Delta \le s \le t$ when action u_t have been chosen. Note that the cardinality of $S(t)$ is independent from Δ, and solely depends on c and the actual sequence of controls chosen according to the stochastic policy π_α.

From Taylor's formula, for all discrete time s,

$$\Delta X_s = X_{s+\Delta} - X_s = f(X_s, u_t)\Delta + \nabla_x f(X_s, u_t) f(X_s, u_t) \frac{\Delta^2}{2} + O(\Delta^3). \tag{19}$$

Now, for $s \in S(t)$ we have $X_t - X_s = O(\Delta)$, thus

$$f(X_s,u_t) = f(X_t,u_t) + \nabla_x f(X_t,u_t)(X_s - X_t) + O(\Delta^2),$$

from which we deduce (using the fact that $\nabla_x f(X_s,u_t) = \nabla_x f(X_t,u_t) + O(\Delta)$) that

$$\begin{aligned}
\Delta X_s &= \Delta X_t + \big[\nabla_x f(X_s,u_t) f(X_s,u_t) - \nabla_x f(X_t,u_t) f(X_t,u_t)\big]\frac{\Delta^2}{2} \\
&\quad + \nabla_x f(X_t,u_t)(X_s - X_t)\Delta + O(\Delta^3) \\
&= \Delta X_t + \nabla_x f(X_t,u_t)[X_s - X_t + \frac{1}{2}(\Delta X_s - \Delta X_t)]\Delta + O(\Delta^3) \qquad (20) \\
&= b + A(X_s + \frac{1}{2}\Delta X_s)\Delta + O(\Delta^3)
\end{aligned}$$

with $b := \Delta X_t - \nabla_x f(X_t,u_t)(X_t + \frac{1}{2}\Delta X_t)\Delta$ and $A := \nabla_x f(X_t,u_t)$. Based on the observation of several jumps $\{\Delta X_s\}_{s\in S(t)}$, one may derive an approximation of $\nabla_x f(X_t,u_t)$ by solving the least-squares problem:

$$\min_{A,b} \frac{1}{n_t} \sum_{s\in S(t)} \Big\| \Delta X_s - b - A\big(X_s + \frac{1}{2}\Delta X_s\big)\Delta \Big\|^2, \qquad (21)$$

where n_t is the cardinality of $S(t)$. Write $X_s^+ := X_s + \frac{1}{2}\Delta X_s = \frac{1}{2}(X_s + X_{s+\Delta})$ and use the simplified notations: $\overline{X}$, $\overline{XX'}$, $\overline{\Delta X}$, and $\overline{\Delta X X'}$, to denote the average values, when $s \in S(t)$, of X_s^+, $X_s^+(X_s^+)'$, ΔX_s, and $\Delta X_s (X_s^+)'$, respectively. For example,

$$\overline{X} := \frac{1}{n_t} \sum_{s\in S(t)} X_s^+.$$

The optimality condition for (21) holds when the matrix $Q_t := \overline{XX'} - \overline{X}\,\overline{X}'$ is invertible, and in that case, the least squares solution provides the approximation $\widehat{\nabla_x f}(X_t,u_t)$ of $\nabla_x f(X_t,u_t)$:

$$\widehat{\nabla_x f}(X_t,u_t) = \frac{1}{\Delta}\big(\overline{\Delta X X'} - \overline{\Delta X}\,\overline{X}'\big)\big(\overline{XX'} - \overline{X}\,\overline{X}'\big)^{-1}. \qquad (22)$$

This optimality condition does not hold when the set of points $(X_s^+)_{s\in S(t)}$ lies in a vector space of dimension $< d$ (then, Q_t is degenerate). In order to circumvent this problem, we assume that the eigenvalues of the matrix Q_t are bounded away from 0, in the sense given in the following proposition (whose proof in provided in Appendix C).

Proposition 8 *The matrix $Q_t = \overline{XX'} - \overline{X}\,\overline{X}'$ is symmetric non-negative. Let $\nu(\Delta) \geq 0$ be the smallest eigenvalue of Q_t, for all $0 \leq t \leq T$. Then, if $\nu(\Delta) > 0$ and $\nu(\Delta)$ satisfies*

$$\frac{1}{\nu(\Delta)} = o(\Delta^{-4}), \qquad (23)$$

then, for all $0 \leq t \leq T$, the least squares estimate $\widehat{\nabla_x f}(X_t,u_t)$ defined by (22) is consistent with the gradient $\nabla_x f(X_t,u_t)$, that is:

$$\lim_{\Delta\to 0} \widehat{\nabla_x f}(X_t,u_t) = \nabla_x f(X_t,u_t).$$

The condition (23) is not easy to check since it depends on the state dynamics and the policy. Note however that, when we use a strict stochastic policy (i.e., $\pi_\alpha > 0$), a sufficient condition for the set of points $(X_s^+)_{s\in S(t)}$ to span a vector space of dimension d is that the system be (at least locally) controllable. In the case of linear systems $dx/dt = Ax + Bu$, where $u \in U = \mathbb{R}^q$, and A and B being $d \times d$ and $d \times q$-matrices respectively, a necessary and sufficient condition for controllability is that the $d \times (qd)$ controllability matrix $[B : AB : A^2B : \cdots : A^{d-1}B]$ has rank d (this is the so-called *Kalman rank condition* (Kalman et al., 1969)). In more general settings, for example when f is a linear combination of vector fields $h_i(x)$ weighted by the control components, i.e. $f(x,u) = \sum_{i=1}^q h_i(x)u_i$, a sufficient condition for controllability is that the dimension of the Lie algebra generated by the fields $\{h_i\}$ is d (see e.g. (LaValle, 2006)). Intuitively, this dimension represents the number of possible independent directions of movement when following any sequence of controls.

In our numerical experiments, we observed the convergence of the $\nabla_x f$ estimate.

Remark 9 *A simple on-line way for approximating $\nabla_x f$ is to consider a weighted least-squares problem using an exponential weight (with some coefficient $\lambda \in (0,1)$) instead of the rectangular window $s \in [t - c\Delta, t]$. The piece of information related to a time $s < t$ is weighted by λ^p, where p is the number of times the control u has been chosen between s and t. It is easy to adapt the proof of Proposition 8 to derive that a such weighted least squares estimate for $\nabla_x f$ is consistent, for any $\lambda \in (0,1)$, under the same condition (23).*

An on-line update rule would consider tables for the average values $\overline{Y}(u)$ (where $\overline{Y}$ means $\overline{X}$, $\overline{XX'}$, $\overline{\Delta X}$, or $\overline{\Delta X X'}$) for all $u \in U$. The values are initialized (at the first time t each action u is encountered) by Y_t, where Y_t means X_t^+, $X_t^+(X_t^+)'$, ΔX_t, and $\Delta X_t(X_t^+)'$, respectively. Then, the values are updated at time t, according to

$$\begin{array}{lll} \overline{Y}(u) \leftarrow \lambda \overline{Y}(u) + (1-\lambda) Y_t & \text{for} & u = u_t, \\ \overline{Y}(u) \text{ stays unchanged} & \text{for} & u \neq u_t. \end{array}$$

The quantities $\overline{X}$, $\overline{XX'}$, $\overline{\Delta X}$, and $\overline{\Delta X X'}$ are easily updated and the estimate $\widehat{\nabla_x f}$ may advantageously be computed from (22) by using an iterative matrix inversion, such as with the Sherman-Morrison formula (see for example (Golub and Loan, 1996)).

Note that for the first discrete times t, the matrix $\overline{XX'} - \overline{X}\,\overline{X}'$ may not be invertible, simply because there is not enough points $(X_s)_{s<t}$ to form a subspace of dimension d. We may simply set $\widehat{\nabla_x f}$ to 0, which has no impact on the general convergence result.

3.3 The Reinforcement Learning Algorithm

Here, we derive a convergent policy gradient estimate in which all information required to build the state gradient Z_t^Δ is the past trajectory $(X_s^\Delta)_{0\le s\le t}$.

Choose a time step Δ. For a given stochastic policy π_α, the algorithm proceeds as follows:

1. At time $t = 0$, initialise $X_0^\Delta = x$ and $Z_0^\Delta = 0$.
2. For each discrete time $t \in \{(t_n)_{0\le n<N}\}$, choose an action $u_t \sim \pi_\alpha(t, X_t^\Delta)$ according to the stochastic policy π_α and keep this action for a period of time Δ, which moves the system from X_t^Δ to $X_{t+\Delta}^\Delta$ (summarized by the dynamics (15)).

3. Update the average values $\overline{X}$, $\overline{XX'}$, $\overline{\Delta X}$, or $\overline{\Delta X X'}$, for all $u \in U$, as described in subsection 3.2, for example by using an exponential trace with parameter $\lambda \in (0,1)$ as mentioned in Remark 9.

4. Compute the state dynamics gradient approximation $\widehat{\nabla_x f}(X_t^\Delta, u_t)$ according to

$$\widehat{\nabla_x f}(X_t^\Delta, u_t) = \frac{1}{\Delta}\big(\overline{\Delta X X'} - \overline{\Delta X}\,\overline{X}'\big)\big(\overline{XX'} - \overline{X}\,\overline{X}'\big)^{-1}.$$

5. Update Z_t^Δ according to

$$\begin{aligned} Z_{t+\Delta}^\Delta &= Z_t^\Delta + \Delta X_t^\Delta \Big[\frac{\big[\nabla_\alpha \pi_\alpha(u_t|t,X_t^\Delta)\big]'}{\pi_\alpha(u_t|t,X_t^\Delta)} + \frac{\big[\nabla_x \pi_\alpha(u_t|t,X_t^\Delta)\big]'}{\pi_\alpha(u_t|t,X_t^\Delta)} Z_t^\Delta \Big] \\ &\quad + \widehat{\nabla_x f}(t,X_t^\Delta,u_t) Z_t^\Delta \Delta. \end{aligned} \tag{24}$$

6. Repeat steps 2-5 until $t = T$. Then return the policy gradient estimate $\nabla_x r(X_T^\Delta) Z_T^\Delta$.

This algorithm returns a consistent approximation of the policy gradient $\nabla_\alpha V(\alpha)$, as stated now.

Proposition 10 *Assume that the property (23) of Proposition 8 holds, and that the reward function is continuously differentiable. Then the estimate $\nabla_x r(X_T^\Delta) Z_T^\Delta$ returned by the RL algorithm is a consistent approximation of the policy gradient $\nabla_\alpha V(\alpha)$, in the sense that $\nabla_x r(X_T^\Delta) Z_T^\Delta$ converges almost surely to $\nabla_\alpha V(\alpha)$ when $\Delta \to 0$.*

Proof From Proposition 8, $\widehat{\nabla_x f}$ is a consistent approximation of $\nabla_x f$, thus the process (Z_t^Δ) built from (24) also satisfies the consistency condition (17), and the proof follows like in Proposition 7.■

3.4 Illustration on a Simple Example

Let us illustrate this algorithm on the simple example described in the introduction (for which we observed the infinite variance of the likelihood ratio estimate in the continuous time limit).

The continuous process is defined by (8) and the discrete time stochastic process by (9). With the notations used in the introduction, the state gradient dynamics (24) is:

$$Z_{t_{n+1}}^\Delta = Z_{t_n}^\Delta + (X_{t_{n+1}}^\Delta - X_{t_n}^\Delta)\frac{\nabla_\alpha \pi_\alpha(u_{t_n}|t,X_{t_n}^\Delta)}{\pi_\alpha(u_{t_n}|t,X_{t_n}^\Delta)} = Z_t^\Delta + \begin{pmatrix} U_n/\alpha \\ (1-U_n)/(\alpha-1) \end{pmatrix}\Delta.$$

Thus the gradient estimate (18) is

$$g(\Delta) = \nabla r(X_{T=1}^\Delta) Z_{T=1,1}^\Delta = \frac{1}{\alpha N}\Big(\sum_{n=0}^{N-1} U_n\Big).$$

Since $\mathbb{E}[g(\Delta)] = 1$, this is an unbiased estimate of the true gradient $\nabla_\alpha V(\alpha) = \nabla_\alpha r(x_1) = 1$. Moreover, its variance $\mathrm{Var}[g(\Delta)] = \frac{1}{\alpha^2 N}\mathrm{Var}[U_n] = \frac{1-\alpha}{\alpha N}$ decreases to 0 when N goes to infinity, which contrast with the variance of the likelihood ratio estimate (11).

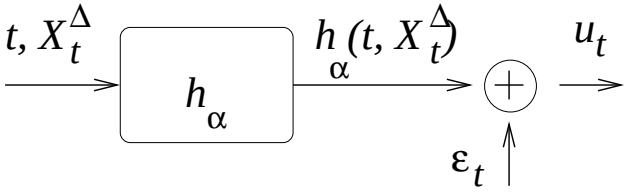


Figure 2: A stochastic policy $u_t = h_\alpha(t, X_t^\Delta) + \varepsilon_t$ with $\varepsilon_t \sim \mathcal{N}(0, v(\Delta))$.

3.5 The Continuous Control Space Case

So far, we have used notations for a finite control space U. However, the same results hold in the case of a continuous control space $U \in \mathbb{R}^q$. Let us illustrate a simple way for defining a stochastic policy based on a parameterized deterministic policy. Let $h_\alpha : [0,T] \times \mathbb{R}^d \to U = \mathbb{R}^q$ be a deterministic policy parameterized by α (which may be implemented by a neural network, or with any other function approximator). We search for a value of the parameter α that maximizes the performance of the corresponding policy.

We build a stochastic policy by perturbing h_α with a centered Gaussian noise of covariance matrix $v(\Delta)$ (i.e. which depends on the discretization time-step Δ). Thus $u_t = h_\alpha(t, X_t^\Delta) + \varepsilon_t$ with $\varepsilon_t \sim \mathcal{N}(0, v(\Delta))$. See Figure 2. We assume that $\lim_{\Delta \to 0} v(\Delta) = 0$.

This stochastic policy admits a probability density representation $\pi_\alpha(u|t,x)$:

$$\pi_\alpha(u|t,x) = \frac{1}{\sqrt{(2\pi)^p |v(\Delta)|}} \exp\Big[-\frac{1}{2}(u - h_\alpha(t,x))' v(\Delta)^{-1} (u - h_\alpha(t,x)) \Big].$$

The stochastic process (X_t^Δ) built according to (15) from this stochastic policy π_α is consistent with the continuous process (x_t) defined by the parameterized deterministic policy h_α:

$$\frac{dx_t}{dt} = f(x, h_\alpha(t,x)).$$

Indeed, from the continuity of f, and the assumption that $v(\Delta) \stackrel{\Delta \to 0}{\longrightarrow} 0$, the average state dynamics vector using the stochastic policy π_α tends to the state dynamics vector using the deterministic policy h_α:

$$\lim_{\Delta \to 0} \int_{\mathbb{R}^q} f(x,u) \pi_\alpha(u|t,x) du = f(x, h_\alpha(t,x)),$$

and the consistency property (13) as well as the bound (14) hold (for the same reasons as those invoked in subsection 2.2). Thus, the reinforcement learning algorithm of subsection 3.3 applies directly.

Note that from the specific form of the policy $\pi_\alpha(u|t,x)$, the likelihood ratios are easily computed: for each parameter α_i, $1 \le i \le m$, $\frac{\partial_{\alpha_i} \pi_\alpha(u|t,x)}{\pi_\alpha(u|t,x)} = \partial_{\alpha_i} h_\alpha(t,x) v(\Delta)^{-1} (u - h_\alpha(t,x))$, and for each coordinate x_i, $1 \le i \le d$, $\frac{\partial_{x_i} \pi_\alpha(u|t,x)}{\pi_\alpha(u|t,x)} = \partial_{x_i} h_\alpha(t,x) v(\Delta)^{-1} (u - h_\alpha(t,x))$.

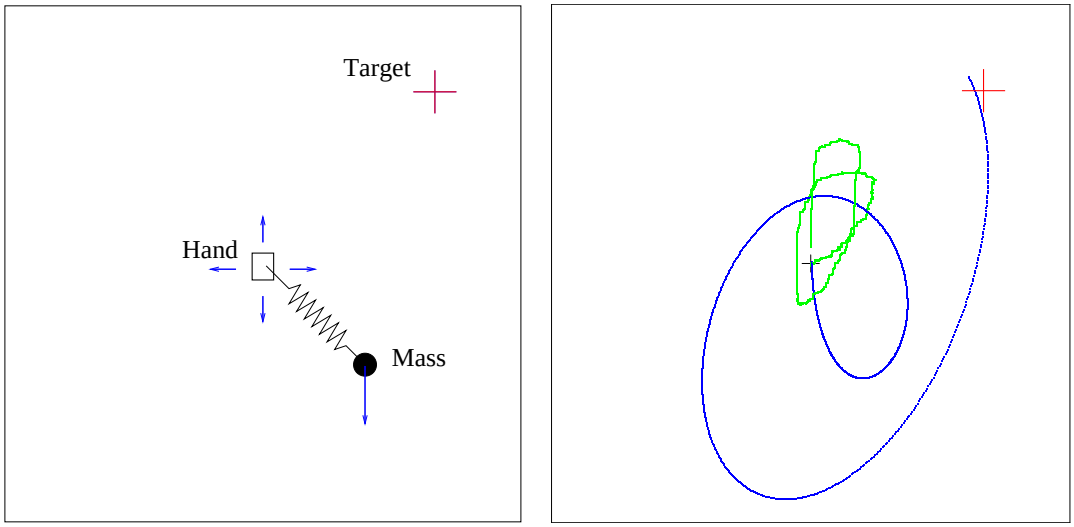
4. Numerical Experiments

We provide two experiments, a target problem and an inverted pendulum, that illustrate the reinforcement learning algorithm described in subsection 3.3 in the case of a finite and a continuous control space, respectively.

4.1 A Target Problem

This is a 6 dimensional system $(x_0, y_0, x, y, v_x, v_y)$ that represents a hand ((x_0, y_0) position) holding a spring to which is attached a mass (defined by its position (x, y) and velocity (v_x, v_y)) subject to gravitation. The control is the movement of the hand, in any 4 possible directions (up, down, left, right). The goal is to reach a target (x_G, y_G) with the mass at a specific time T (see Figure 3a), while keeping the hand close to the origin. For that purpose, the terminal reward function is defined by

$$r = -x_0^2 - y_0^2 - (x - x_G)^2 - (y - y_G)^2.$$



(a) The physical system

(b) A trajectory (the mass and the hand) starting from the origin

Figure 3: (a) the physical system. (b) A trajectory obtained after 1000 gradient steps. For that specific trajectory, the performance (terminal reward) was -0.087.

The state dynamics is:

$$\begin{aligned} \dot{x_0} &= u_x, \quad \dot{x} = v_x, \quad \dot{v_x} = -\tfrac{k}{m}(x - x_0), \\ \dot{y_0} &= u_y, \quad \dot{y} = v_y, \quad \dot{v_y} = -\tfrac{k}{m}(y - y_0) - g, \end{aligned}$$

with k being the spring constant, m the mass, g the gravitational constant, and $(u_x, u_y) = u \in U := \{(1,0),(0,1),(-1,0),(0,-1)\}$ the control. We consider a Boltzmann-like stochastic policy

$$\pi_\alpha(u|t,x) = \frac{\exp Q_\alpha(t,x,u)}{\sum_{u' \in U} \exp Q_\alpha(t,x,u')}$$

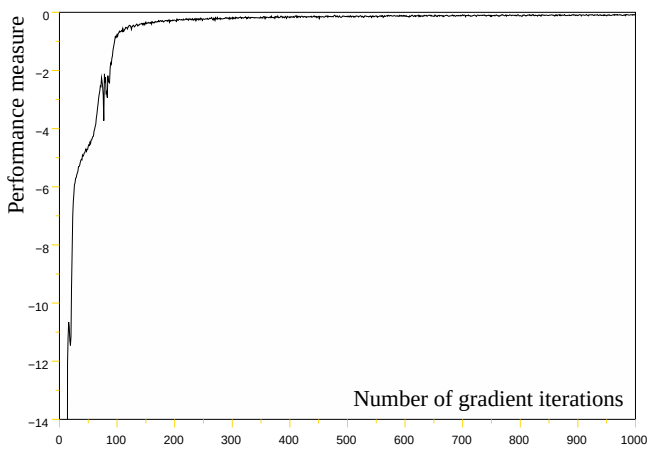


Figure 4: Performance of successive parameterized controllers.

with a linear parameterization of the Q_α values: $Q_\alpha(t,x,u) = \alpha_0^u + \alpha_1^u t + \alpha_2^u x_0 + \alpha_3^u y_0 + \alpha_4^u x + \alpha_5^u y + \alpha_6^u v_x + \alpha_7^u v_y$, for each 4 possible actions $u \in U$. Thus the parameter $\alpha \in I\!R^{32}$. We initialized α with uniform random values in the range $[-0.01, 0.01]$. In our experiments we chose $k=1$, $m=1$, $g=1$, $x_G = y_G = 2$, $\lambda = 0.9$, $\Delta = 0.01$, $T = 10$.

At each iteration, we run one trajectory $(X_t)_{0 \le t \le T}$ using the stochastic policy, and calculate the policy gradient estimate according to the RL algorithm described in subsection 3.3. We then perform a gradient ascent step (5) (with a fixed step $\eta = 0.01$). Figure 4 shows the performance of the parameterized controller as a function of the number of gradient iterations.

For that problem, we chose initial states uniformly distributed over the domain $[-0.1, 0.1]^6$. We found that the randomness introduced in the choice of the initial state helped in not getting stuck in local minima. Here, convergence of the gradient method occurs to a controller close to optimality (for which $r = 0$). We illustrate in Figure 3b the trajectory (where only the hand and the mass positions are shown) obtained after 1000 gradient steps, starting from the initial state $(x_0, y_0, x, y, v_x, v_y)_{t=0} = 0$.

4.2 Double Inverted Pendulum

We illustrate the approach described in subsection 3.5 on this continuous control space problem. This is an double inverted pendulum defined in the 6-dimensions: the position of the cart, its velocity, the two angles, and their angular velocity $x = (y, v, \theta_1, \omega_1, \theta_2, \omega_2)' \in I\!R^6$ (see Figure 5). The control $u \in U = I\!R$ (continuous variable) is the force applied to the cart. The state dynamics are described in (Bogdanov, 2004). The goal is to reach the unstable equilibrium $(y, v, \theta_1, \omega_1, \theta_2, \omega_2) = 0$ at time $T = 5$. We consider the quadratic reward function $r(x) = -(y^2 + v^2 + \theta_1^2 + \omega_1^2 + \theta_2^2 + \omega_2^2)$.

Like in subsection 3.5, we build a stochastic policy by adding a Gaussian noise of variance $v(\Delta) = \Delta I$ (where I is the identity matrix) to a linearly parameterized (time independent) determin-

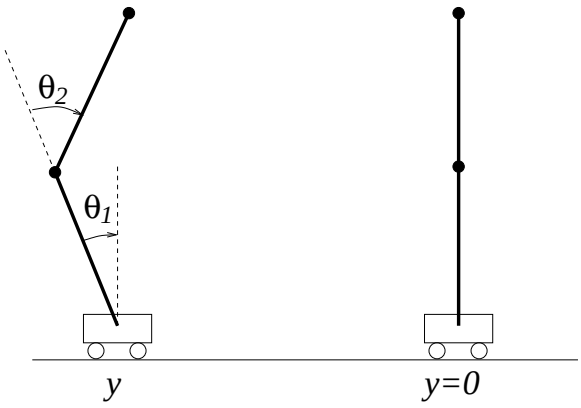


Figure 5: The double inverted pendulum. Current position and target position.

istic policy $h_\alpha(t,x) = \alpha_1 + \alpha_2 y + \alpha_3 v + \alpha_4\theta_1 + \alpha_5\omega_1 + \alpha_6\theta_2 + \alpha_7\omega_2$, i.e. the control at time t is $u_t \sim h_\alpha(t,x_t) + \mathcal{N}(0, v(\Delta))$.

We wish to find a local maximum of the performance measure $V(\alpha) = r(x_T)$ in the space of the policy parameters $\alpha \in \mathbb{R}^7$. We initialized α with uniform random values in the range $[-0.01, 0.01]$, and perform a stochastic gradient algorithm (5) where the gradient $\nabla_\alpha V(\alpha)$ is computed according to the reinforcement learning algorithm defined in subsection 3.3.

A gradient step update (5) is performed (with $\eta = 1$) at the end of each sample trajectory starting from an initial state, chosen uniformly randomly in the domain defined by $y \in [-1,1]$, $\theta_1 \in [-0.3, 0.3]$, $\theta_2 \in [-0.3, 0.3]$, and $v = 0$, $\omega_1 = 0$, $\omega_2 = 0$. We use a discretization time-step $\Delta = 10^{-3}$ which is low enough to provide a very good approximation of the true gradient, that is the gradient that would be obtained from the continuous (but unknown from the agent) state dynamics by using the deterministic policy $h_\alpha(t,x)$.

Figure 6 shows (in bold) the performance measure (terminal reward) at the end of each trajectory as a function of the number of gradient iterations. The other curves give the values of the $(\alpha_1, \ldots, \alpha_7)$ during simulations.

After 1000 gradient iterations, the obtained policy is $h_\alpha(t,x) = -0.0023 - 5.31y - 1.74v + 11.16\theta_1 + 0.92\omega_1 - 7.77\theta_2 - 3.94\omega_2$, and the resulting average performance is -0.097 for trajectories starting randomly from the same domain as during learning. In this problem, a linear controller is sufficient to derive a controller close to optimality. However, we should mention that for initial states in another domain (say, if the angles were not close to 0, and loops would be required to reach the target position), the problem would not possibly be solved with such a simple class of policies.

5. Conclusion

We described a reinforcement learning method for approximating the gradient of the performance measure of a continuous-time deterministic problem, with respect to the control parameters. This was obtained by using a stochastic policy to approximate the continuous system by a consistent stochastic discrete process. We showed how using a perturbated parameterized deterministic policy enables to process a consistent (when the perturbation goes to 0) gradient estimate only from the observable data.

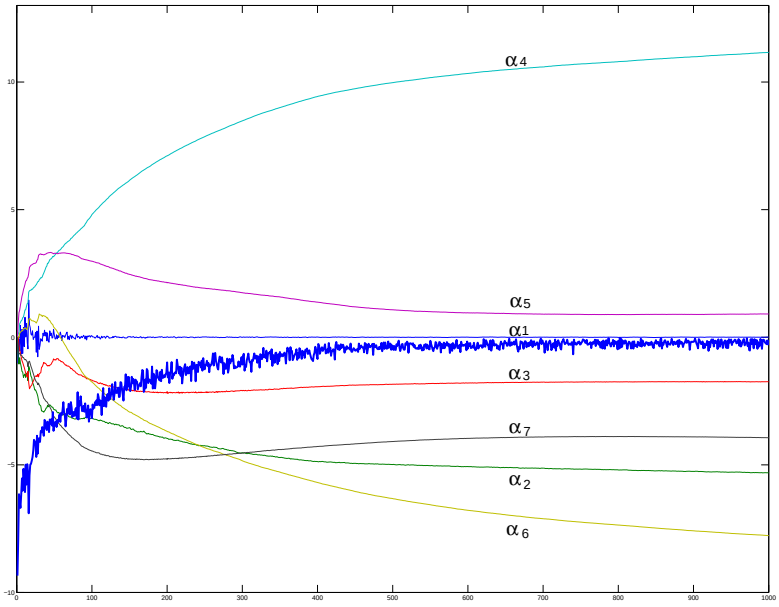


Figure 6: The bold curve shows the performance measure $V(\alpha)$, and the other curves the values of $(\alpha_1, \ldots, \alpha_7)$, as a function of the number of gradient iterations.

In future work, it would be interesting to extend this method to the case of stochastic dynamics, and to non-smooth reward functions (or in case the reward gradient is unknown from the agent), by using integration-by-part formula for the gradient estimate, such as the *likelihood ratio method* of (Yang and Kushner, 1991) or the *martingale approach* of (Gobet and Munos, 2005).

Appendix A. Proof of Proposition 2

The likelihood ratio estimate (10) may be rewritten

$$\begin{aligned} g(\Delta) &= \frac{1}{\alpha(1-\alpha)N}\Big(\sum_{n=0}^{N-1} U_n\Big)\sum_{n=0}^{N-1}\big(U_n-\alpha\big) \\ &= \frac{1}{\alpha(1-\alpha)N}\Big[\Big(\sum_{n=0}^{N-1} V_n\Big)^2 + \alpha N\sum_{n=0}^{N-1} V_n\Big], \end{aligned}$$

with $V_n := U_n - \alpha$. From the fact that $\mathbb{E}[V_n^2] = \alpha(1-\alpha)$, the expectation of the estimate is

$$\mathbb{E}[g(\Delta)] = \frac{1}{\alpha(1-\alpha)N}\mathbb{E}\Big[\Big(\sum_{n=0}^{N-1} V_n\Big)^2\Big] = 1.$$

Now its variance $\mathrm{Var}[g(\Delta)]$ is

$$\frac{1}{[\alpha(1-\alpha)N]^2}\mathrm{Cov}\Big[\Big(\sum_{n=0}^{N-1} V_n\Big)\Big(\sum_{p=0}^{N-1} V_p\Big) + \alpha N\sum_{n=0}^{N-1} V_n, \Big(\sum_{n'=0}^{N-1} V_{n'}\Big)\Big(\sum_{p'=0}^{N-1} V_{p'}\Big) + \alpha N\sum_{n=0}^{N-1} V_{n'}\Big]. \qquad (25)$$

Notice that from the independence of the Bernoulli random variables (U_n), all terms $\mathrm{Cov}(V_n, V_{n'}) = 0$ for $n \neq n'$, and $\mathrm{Cov}(V_n, V_n) = \mathbb{E}[(U_n - \alpha)^2] = \alpha(1-\alpha)$.

The terms $\mathrm{Cov}(V_n, V_{n'}V_{p'}) = \mathbb{E}\big[V_n(V_{n'}V_{p'} - \mathbb{E}[V_{n'}V_{p'}])\big] = \mathbb{E}[V_nV_{n'}V_{p'}]$ (because V_n is centered) equal 0 whenever $n \neq n'$ or $n \neq p'$. And $\mathrm{Cov}(V_n, V_n^2) = \mathbb{E}[V_n^3] = \alpha(1-\alpha)(1-2\alpha)$.

Now, $\mathrm{Cov}(V_nV_p, V_{n'}V_{p'}) = 0$ when $n \neq n'$, $n \neq p'$, $p \neq n'$, and $p \neq p'$ (because the variables V_nV_p and $V_{n'}V_{p'}$ are independent). The terms $\mathrm{Cov}(V_nV_p, V_nV_{p'}) = \mathbb{E}\big[(V_nV_p - \mathbb{E}[V_nV_p])(V_nV_{p'} - \mathbb{E}[V_nV_{p'}])\big] = \mathbb{E}[V_nV_pV_nV_{p'}] = 0$ for $n \neq p$, $n \neq p'$, and $p \neq p'$ (independence of V_p and $V_n^2V_{p'}$). Now, $\mathrm{Cov}(V_nV_p, V_nV_p) = \mathbb{E}[(V_nV_p)^2] = \alpha^2(1-\alpha)^2$ when $n \neq p$. Finally, $\mathrm{Cov}(V_n^2, V_n^2) = \mathbb{E}[V_n^4] - \big(\mathbb{E}[V_n^2]\big)^2 = \alpha(1-\alpha)(1-3\alpha+3\alpha^2) - \alpha^2(1-\alpha)^2 = \alpha(1-\alpha)(1-4\alpha+4\alpha^2)$.

Thus, the covariance term in (25) is

$$N\alpha(1-\alpha)(1-4\alpha+4\alpha^2) + N(N-1)\alpha^2(1-\alpha)^2 + \alpha N^2\alpha(1-\alpha)(1-2\alpha) + \alpha^2N^3\alpha(1-\alpha)$$

and the variance of the likelihood ratio estimate is

$$\mathrm{Var}[g(\Delta)] = \frac{1 - 5(1-\alpha) + (2-3\alpha)\alpha N + \alpha^2N^2}{\alpha(1-\alpha)N}.$$

Appendix B. Proof of Theorem 3

For convenience, we write x_n for x_{t_n}, X_n for $X^\Delta_{t_n}$, u_n for u_{t_n}, and U_n for U_{t_n}, $0 \leq n \leq N$. Let us define the average approximation errors $m^\Delta_n = \mathbb{E}[||X_n - x_n||]$ and the squared errors $v^\Delta_n = \mathbb{E}[||X_n - x_n||^2]$. Here, we prove the convergence at the terminal time T, i.e. that $X_T \to x_T$ almost surely when $\Delta \to 0$.

B.1 Convergence of the Squared Error $\mathbb{E}[||X^\Delta_T - x_T||^2]$:

We use the decomposition:

$$\begin{aligned} v^\Delta_{n+1} &= \mathbb{E}[||X_{n+1} - X_n||^2] + \mathbb{E}[||X_n - x_n||^2] + \mathbb{E}[||x_n - x_{n+1}||^2] \\ &\quad + 2\mathbb{E}[(X_n - x_n)'(X_{n+1} - X_n + x_n - x_{n+1})] \\ &\quad + 2\mathbb{E}[(X_{n+1} - X_n)'(x_n - x_{n+1})]. \end{aligned} \tag{26}$$

From the bounded jumps property (14), $\mathbb{E}[||X_{n+1} - X_n||^2] = O(\Delta^2)$. From Taylor's formula,

$$x_{n+1} - x_n = f(x_n)\Delta + O(\Delta^2), \tag{27}$$

thus $\mathbb{E}[||x_n - x_{n+1}||^2] = O(\Delta^2)$ (since f is Lipschitz, and x_t and $f(x_t)$ are uniformly bounded on $[0,T]$) and from Cauchy-Schwarz inequality, $|\mathbb{E}[(X_{n+1} - X_n)'(x_n - x_{n+1})]| = O(\Delta^2)$. From (13) and (27),

$$\mathbb{E}[X_{n+1} - X_n + x_n - x_{n+1}|X_n] = [f(X_n) - f(x_n)]\Delta + o(\Delta). \tag{28}$$

Now, from (14) we deduced that $||X_n - x_0|| = O(1)$ thus X_n is bounded (for all n and N), as well as x_n. Let B a constant such that $||X_n|| \leq B$ and $||x_n|| \leq B$ for all $n \leq N$, $N \geq 0$. Since f is $\mathcal{C}^2$, from Taylor's formula, there exists a constant k, such that, for all $n \leq N$,

$$||f(X_n) - f(x_n) - \nabla_x f(x_n)(X_n - x_n)|| \leq k||X_n - x_n||^2. \tag{29}$$

We deduce that

$$\begin{aligned}|\mathbb{E}[(X_n - x_n)'(X_{n+1} - X_n + x_n - x_{n+1})]| &= \left|\mathbb{E}\left[(X_n - x_n)'(f(X_n) - f(x_n))\right]\right|\Delta + o(\Delta) \\ &\leq \left|\mathbb{E}\left[(X_n - x_n)'\nabla_x f(x_n)(X_n - x_n)\right]\right|\Delta + 2kBv_n\Delta + o(\Delta) \\ &\leq Mv_n^\Delta\Delta + o(\Delta)\end{aligned}$$

with $M = \sup_{||x||\leq B} ||\nabla_x f(x)|| + 2kB$. Thus, (26) leads to the recurrent bound

$$v_{n+1}^\Delta \leq (1 + M\Delta)v_n^\Delta + o(\Delta).$$

This actually means that there exists a function $e(\Delta) \to 0$ when $\Delta \to 0$, such that $v_{n+1}^\Delta \leq (1 + M\Delta)v_n^\Delta + e(\Delta)\Delta$. Thus,

$$v_N^\Delta \leq \frac{(1+M\Delta)^N - 1}{(1+M\Delta) - 1}e(\Delta)\Delta \leq (e^{NM\Delta} - 1)\frac{1}{M}e(\Delta)$$

thus $v_N^\Delta = o(1)$, that is $\mathbb{E}[||X_T^\Delta - x_T||^2] \xrightarrow{\Delta\to 0} 0$.

B.2 Convergence of the Mean $\mathbb{E}[||X_T^\Delta - x_T||]$:

From (28), we have

$$\mathbb{E}[X_{n+1} - x_{n+1}|X_n] = X_n - x_n + [f(X_n) - f(x_n)]\Delta + o(\Delta).$$

Thus from (29),

$$\begin{aligned}m_{n+1}^\Delta = \mathbb{E}[||X_{n+1} - x_{n+1}||] &\leq (1 + ||\nabla_x f(x_n)||\Delta)\mathbb{E}[||X_n - x_n||] + kv_n^\Delta\Delta + o(\Delta) \\ &\leq (1 + M'\Delta)m_n^\Delta + o(\Delta),\end{aligned}$$

since $v_n^\Delta = o(1)$ (with $M' = \sup_{||x||\leq B} ||\nabla_x f(x)||$). Using the same deduction as above, we obtain that $m_N^\Delta = o(1)$, that is $\mathbb{E}[||X_T^\Delta - x_T||] \xrightarrow{\Delta\to 0} 0$.

B.3 Almost Sure Convergence

Here, we use the *concentration-of-measure phenomenon* (Talagrand, 1996; Ledoux, 2001), which states that under mild conditions, a function (say Lipschitz or with bounded differences) of many independent random variables concentrates around its mean, in the sense that the tail probability decreases exponentially fast.

From the definition of the discrete state process (12), one may write the state X_N as a function h of the independent random variables $(U_n)_{0\leq n<N}$, i.e.

$$X_N - x_0 = h(U_0, \ldots, U_{N-1}) := \sum_{n=0}^{N-1}(X_{n+1} - X_n). \tag{30}$$

Observe that $h - \mathbb{E}[h] = \sum_{n=0}^{N-1} d_n$ with $d_n = X_{n+1} - X_n - \mathbb{E}[X_{n+1} - X_n]$ being a martingale difference sequence (that is $\mathbb{E}[d_n|U_0, \ldots, U_{n-1}] = 0$). Now, from (Ledoux, 2001, lemma 4.1), one has:

$$\mathbb{P}(||h - \mathbb{E}[h]|| \geq \varepsilon) \leq 2e^{-\varepsilon^2/(2D^2)} \tag{31}$$

for any $D^2 \geq \sum_{n=0}^{N-1} ||d_n||_\infty^2$. Thus, from (14), and since $f^\Delta(X_n)$ is bounded (for all $n < N$ and all $N > 0$), there exists a constant C that does not depend on N such that $d_n \leq C/N$. Thus we may take $D^2 = C^2/N$.

Now, from the previous paragraph, $||\mathbb{E}[X_N] - x_N|| \leq e(N)$, with $e(N) \to 0$ when $N \to \infty$. This means that $||h - \mathbb{E}[h]|| + e(N) \geq ||X_N - x_N||$, thus

$$\mathbb{P}(||h - \mathbb{E}[h]|| \geq \varepsilon + e(N)) \geq \mathbb{P}(||X_N - x_N|| \geq \varepsilon),$$

and we deduce from (31) that

$$\mathbb{P}(||X_N - x_N|| \geq \varepsilon) \leq 2e^{-N(\varepsilon + e(N))^2/(2C^2)}.$$

Thus, for all $\varepsilon > 0$, the series $\sum_{N \geq 0} \mathbb{P}(||X_N - x_N|| \geq \varepsilon)$ converges. Now, from Borel-Cantelli lemma, we deduce that for all $\varepsilon > 0$, there exists N_ε such that for all $N \geq N_\varepsilon$, $||X_N - x_N|| < \varepsilon$, which proves the almost sure convergence of X_N to x_N as $N \to \infty$ (i.e. $X_T \overset{\Delta \to 0}{\longrightarrow} x_T$ almost surely).

Appendix C. Proof of Proposition 8

First, note that $Q_t = \overline{X\,X'} - \overline{X}\,\overline{X}'$ is a symmetric, non-negative matrix, since it may be rewritten as

$$\frac{1}{n_t} \sum_{s \in S(t)} (X_s^+ - \overline{X})(X_s^+ - \overline{X})'.$$

In solving the least squares problem (21), we deduce $b = \overline{\Delta X} + A\overline{X}\Delta$, thus

$$\begin{aligned} \min_{A,b} \frac{1}{n_t} \sum_{s \in S(t)} \left\| \Delta X_s - b - A(X_s + \frac{1}{2}\Delta X_s)\Delta \right\|^2 &= \min_A \frac{1}{n_t} \sum_{s \in S(t)} \left\| \Delta X_s - \overline{\Delta X} - A(X_s^+ - \overline{X})\Delta \right\|^2 \\ &\leq \frac{1}{n_t} \sum_{s \in S(t)} \left\| \Delta X_s - \overline{\Delta X} - \nabla_x f(\overline{X}, u_t)(X_s^+ - \overline{X})\Delta \right\|^2. \end{aligned} \tag{32}$$

Now, since $X_s = \overline{X} + O(\Delta)$ one may obtain like in (19) and (20) (by replacing X_t by $\overline{X}$) that:

$$\Delta X_s - \overline{\Delta X} - \nabla_x f(\overline{X}, u_t)(X_s^+ - \overline{X})\Delta = O(\Delta^3). \tag{33}$$

We deduce from (32) and (33) that

$$\frac{1}{n_t} \sum_{s \in S(t)} \left\| \left[\widehat{\nabla_x f}(X_t, u_t) - \nabla_x f(\overline{X}, u_t) \right] (X_s^+ - \overline{X})\Delta \right\|^2 = O(\Delta^6).$$

By developing each component,

$$\sum_{i=1}^{d} \left[\widehat{\nabla_x f}(X_t, u_t) - \nabla_x f(\overline{X}, u_t) \right]_{row\,i} Q_t \left[\widehat{\nabla_x f}(X_t, u_t) - \nabla_x f(\overline{X}, u_t) \right]'_{row\,i} = O(\Delta^4).$$

Now, from the definition of $\nu(\Delta)$, for all vector $u \in \mathbb{R}^d$, $u' Q_t u \geq \nu(\Delta)||u||^2$, thus

$$\nu(\Delta) ||\widehat{\nabla_x f}(X_t, u_t) - \nabla_x f(\overline{X}, u_t)||^2 = O(\Delta^4).$$

Condition (23) yields $\widehat{\nabla_x f}(X_t, u_t) = \nabla_x f(\overline{X}, u_t) + o(1)$, and since $\nabla_x f(X_t, u_t) = \nabla_x f(\overline{X}, u_t) + O(\Delta)$, we deduce

$$\lim_{\Delta \to 0} \widehat{\nabla_x f}(X_t, u_t) = \nabla_x f(X_t, u_t).$$

References

J. Baxter and P. L. Bartlett. Infinite-horizon gradient-based policy search. *Journal of Artificial Intelligence Research*, 15:319–350, 2001.

A. Bensoussan. *Perturbation methods in optimal control*. Wiley/Gauthier-Villars Series in Modern Applied Mathematics. John Wiley & Sons Ltd., Chichester, 1988. Translated from the French by C. Tomson.

A. Bogdanov. Optimal control of a double inverted pendulum on a cart. *Technical report CSE-04-006, CSEE, OGI School of Science and Engineering, OHSU*, 2004.

P. W. Glynn. Likelihood ratio gradient estimation: an overview. In A. Thesen, H. Grant, and W. D. Kelton, editors, *Proceedings of the 1987 Winter Simulation Conference*, pages 366–375, 1987.

E. Gobet and R. Munos. Sensitivity analysis using Itô-Malliavin calculus and martingales. application to stochastic optimal control. *SIAM journal on Control and Optimization*, 43(5):1676–1713, 2005.

G. H. Golub and C. F. Van Loan. *Matrix Computations, 3rd ed.* Baltimore, MD: Johns Hopkins, 1996.

R. E. Kalman, P. L. Falb, and M. A. Arbib. *Topics in Mathematical System Theory*. New York: McGraw Hill, 1969.

P. E. Kloeden and E. Platen. *Numerical Solutions of Stochastic Differential Equations*. Springer-Verlag, 1995.

H. J. Kushner and G. Yin. *Stochastic Approximation Algorithms and Applications*. Springer-Verlag, Berlin and New York, 1997.

S. M. LaValle. *Planning Algorithms*. Cambridge University Press, 2006.

M. Ledoux. *The concentration of measure phenomenon*. American Mathematical Society, Providence, RI, 2001.

P. Marbach and J. N. Tsitsiklis. Approximate gradient methods in policy-space optimization of Markov reward processes. *Journal of Discrete Event Dynamical Systems*, 13:111–148, 2003.

B. T. Polyak. *Introduction to Optimization*. Optimization Software Inc., New York, 1987.

M. I. Reiman and A. Weiss. Sensitivity analysis via likelihood ratios. In J. Wilson, J. Henriksen, and S. Roberts, editors, *Proceedings of the 1986 Winter Simulation Conference*, pages 285–289, 1986.

R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction. *Bradford Book*, 1998.

R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. *Neural Information Processing Systems. MIT Press*, pages 1057–1063, 2000.

M. Talagrand. A new look at independence. *Annals of Probability*, 24:1–34, 1996.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

J. Yang and H. J. Kushner. A Monte Carlo method for sensitivity analysis and parametric optimization of nonlinear stochastic systems. *SIAM J. Control Optim.*, 29(5):1216–1249, 1991.