A Stochastic Gradient Type Algorithm for Closed Loop Problems

Kengy Barty*, Jean-Sébastien Roy†, Cyrille Strugarek‡

13th May 2005

Abstract

We focus on solving closed-loop stochastic problems, and propose a perturbed gradient algorithm to achieve this goal. The main hurdle in such problems is the fact that the control variables are infinite dimensional, and have hence to be represented in a finite way in order to solve the problem numerically. In the same way, the gradient of the criterion is itself an infinite dimensional object. Our algorithm replaces this exact (and unknown) gradient by a perturbed one, which consists in the product of the true gradient evaluated at a random point and a kernel function which extends this gradient to the neighbourhood of the random point. Proceeding this way, we explore the whole space iteration after iteration through random points. Since each kernel function is perfectly known by a finite (and small) number of parameters, say N, the control at iteration k is perfectly known as an infinite dimensional object by at most $N \times k$ parameters.

The main strength of this method is that it avoids any discretization of the underlying space, provided that we can draw as many points as needed in this space. Hence, we can take into account the possible measurability constraints of the problem in a new way.

Moreover, the randomized strategy implemented by the algorithm causes the most probable parts of the space to be the most explored ones, which is a priori an interesting feature.

In this paper, we first show a convergence result of this algorithm in the general case, and then give a few numerical examples showing the interest of this method for solving practical stochastic optimization problems.

Keywords: Stochastic Quasi-Gradient, Perturbed Gradient, Closed-Loop Problems

1. Motivation

The aim of this work is to focus on the way we can use a gradient algorithm for stochastic optimal control problems with closed-loop control variables. The typical problem we will consider is:

$$\min_u \mathcal{J}(u) := \mathbb{E}\left(j(u(\boldsymbol{\xi}), \boldsymbol{\xi})\right),$$
$$\text{s.t. } u \in U^f,$$

with $\boldsymbol{\xi}$ some random variable with values in a space Ξ called the noise space of the problem. The control variable u is searched as a mapping from Ξ to a metric space U, such that $j(u(\cdot), \cdot) : \Xi \to \mathbb{R}$ is measurable. The possible restrictions on the feedback u we are looking for are given by the feasible subset denoted by U^f. Usually, u belongs therefore to an infinite dimensional space. The numerical problem is to find a finite representation of the control variable u.

A classical technique consists in discretizing the space Ξ. By giving to the underlying random variable $\boldsymbol{\xi}$ a discrete probability law, or by quantizing the noise space into a partition, one finds the optimal control for each discretized value of $\boldsymbol{\xi}$. In

[*] **École Nationale des Ponts et Chaussées** (ENPC),
kengy.barty@cermics.enpc.fr

[†] **EDF R&D**
1, avenue du Général de Gaulle
F-92141 Clamart Cedex
jean-sebastien.roy@edf.fr

[‡] **EDF R&D**
cyrille.strugarek@edf.fr
also with the École Nationale Supérieure de Techniques Avancées (ENSTA) and the École Nationale des Ponts et Chaussées (ENPC)

the case of multi-stage stochastic programs, this approach corresponds for example to the representation of the underlying noise space by some scenario trees (see e.g.,[Shapiro and Ruszczynski, 2003] for a good survey of usual stochastic programming techniques, or [Higle and Sen, 1996] for a description of the various possible algorithms for large-scale problems). As soon as the problem becomes a discrete one, every finite dimensional optimization algorithms can be used without difficulties except computational ones. For example, discretized problem can be solved by as many gradient algorithms as points of discretization for $\boldsymbol{\xi}$. Each algorithm is correctly described, and its convergence is well known. But, at the end of these discrete resolutions, one has to build the *continuous* control by some e.g., interpolation step, on the basis of the computed discrete optimal values.
Another possibility is to search for the control u as a linear combination of known functions (see [Holt et al., 1955] or more recently [Bertsekas and Tsitsiklis, 1996]). One has hence to solve a finite dimensional problem, of dimension equal to the cardinality of the function basis. In this approach, one is *restrained* from the beginning to the vector space generated by the given basis, but one avoids any interpolation step.
Our approach is different from the preceding ones. It avoids any interpolation step, and it is not restrained to any a priori subspace of the initial feasible set. Our approach is based on the ideas of stochastic approximations introduced by Robbins and Monro (see [Robbins and Monro, 1951], or [Lai, 2003] for an historical survey of these techniques). We apply the classical stochastic approximation techniques (see e.g., [Delyon, 1996, Borkar, 1998, Granichin, 2002]) to gradient algorithms with stochastic noises,(see e.g., [Bertsekas and Tsitsiklis, 2000, Bertsekas and Tsitsiklis, 1996]). The main idea is to draw at each iteration of our gradient algorithm a realization of the underlying random variable $\boldsymbol{\xi}$, and then, to extend the gradient of $\mathcal{J}$ at the draw, to a given neighbourhood whose size is decreasing along the iterations.
Under regularity assumptions on the cost function j, and under some technical assumptions on the law of $\boldsymbol{\xi}$ and on the kernels used to extend the gradient, we give in subsections 2.2 and 2.3 two convergence proofs for this algorithm in two main cases.
As a particular case, we show in subsection 2.4 how our algorithm generalizes the classical stochastic gradient algorithm for open loop problems.
We then apply in section 3 this algorithm to some optimal control problems.

2. Theoretical framework

2.1. **Algorithm.** Let us denote the euclidean norm in $\mathbb{R}^n$ by $\|\cdot\|_n$, for all $n \in \mathbb{N}$, and analogously by $\langle\cdot,\cdot\rangle_n$ the usual scalar product. We here focus on the problem:

$$\begin{aligned} &\min_u \mathcal{J}(u) := \mathbb{E}\left(j(u(\boldsymbol{\xi}),\boldsymbol{\xi})\right), \\ &\text{s.t. } u \in U^f. \end{aligned} \tag{1}$$

where:

- $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space,
- $\boldsymbol{\xi}$ is a random variable on Ω with values in $\mathbb{R}^m$, and with law μ, admitting a density with respect to the Lebesgue measure,
- $j : \mathbb{R}^p \times \mathbb{R}^m \to \mathbb{R}$ is a normal integrand, i.e., j is such that for all measurable mapping $u : \mathbb{R}^m \to \mathbb{R}^p$, $j(u(\cdot),\cdot) : \mathbb{R}^m \to \mathbb{R}$ is measurable,
- U^f is a closed convex subset, or a sub vector space of $L^2(\mathbb{R}^m, \mathbb{R}^p) := \{u : \mathbb{R}^m \to \mathbb{R}^p \ : \ \mathbb{E}\left(\|u(\boldsymbol{\xi})\|_p^2\right) < \ \infty\}$. This space is an Hilbert space, equipped

with the scalar product:

$$\langle u, v\rangle = \mathbb{E}\left(\langle u(\boldsymbol{\xi}), v(\boldsymbol{\xi})\rangle_p\right).$$

- $\Pi_{U^f}(\cdot)$ denotes the projection onto U^f.

We write the classical gradient algorithm for problem (1), i.e.,

$$u^{k+1} = \Pi_{U^f}\left(u^k - \rho^k \nabla \mathcal{J}(u^k)\right), \text{ a.s.} \tag{2}$$

where (ρ^k) is a given sequence of nonnegative numbers. This algorithm is of course not a practical one, mainly from a numerical point of view. Indeed, the update formula is an equality in $L^2(\mathbb{R}^m, \mathbb{R}^p)$... and the gradient of $\mathcal{J}$ is given by the following formula:

$$\forall u \in L^2(\mathbb{R}^m, \mathbb{R}^p),\ \nabla \mathcal{J}(u)(\cdot) = \nabla_u j(u(\cdot), \cdot). \tag{3}$$

Stochastic approximation algorithms usually aim at estimating an expectation on the basis of successive draws of random variables. Since the gradient is here not an expectation, we cannot use a priori such techniques, though problem (1) is a stochastic problem.

Depending on the feasible set U^f, the stochasticity of the problem (1) is more or less effective. Typically, if the feasible set is of the type $u(\xi) \in \Gamma(\xi)$ a.s. for some mapping Γ, the cost depends on the probability law of $\boldsymbol{\xi}$, but not the optimal value u^* of the problem (which depends only on the support of the random variable $\boldsymbol{\xi}$), since we can invert the expectation and the infimization operators in problem (1).

But we might want to solve such problems with the help of the underlying random variable $\boldsymbol{\xi}$. It leads us to propose the following stochastic algorithm to solve problem (1):

Algorithm 2.1. Step k:

- Draw $\boldsymbol{\xi}^{k+1}$ independently from the past draws according to μ,
- Update:

$$u^{k+1}(\cdot) = \Pi_{U^f}\left(u^k(\cdot) - \rho^k \epsilon^k \nabla_u j(u^k(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}^{k+1}, \cdot)\right),$$

where K^k is a bounded mapping from $\mathbb{R}^m \times \mathbb{R}^m$ to $\mathbb{R}$, and $\epsilon^k > 0$. In the following, we will call the mappings K^k kernels, by analogy with the theory of functional estimation.

Our algorithm 2.1 is hence a stochastic algorithm, but it differs from the classical stochastic gradient algorithm, in that our *noisy gradient* is not an unbiased estimator of the true gradient, but a biased one. Indeed, by denoting $\mathcal{F}^k = \sigma(\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^k)$ the sigma field generated by the past draws:

$$\begin{aligned}\mathbb{E}\left(\nabla_u j(u^k(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}^{k+1}, \cdot) | \mathcal{F}^k\right) =& \mathbb{E}\left(\nabla_u j(u^k(\boldsymbol{\xi}), \boldsymbol{\xi}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}, \cdot)\right), \\ \neq& \nabla_u j(u^k(\cdot), \cdot).\end{aligned}$$

One of the main interests of this algorithm is to provide an estimate of the optimal feedback without any *interpolation* step involving a lot of calculations. Typically, the kernels are determined at each iteration by essentially two parameters: their window size ϵ^k and their center $\boldsymbol{\xi}^{k+1}$, as will be developped further. At iteration k, the feedback u^{k+1} will be perfectly known on its possibly continuous domain by at most $2 \times (k+1)$ parameters. It may therefore be an interesting alternative to the classical one which consists in two steps: a discretization of the underlying noise space, and an interpolation step.

Before going to the convergence proofs, let us give some notations. For all mapping $G \in L^2(\mathbb{R}^m \times \mathbb{R}^m, \mathbb{R}^m)$, we will write $\mathbb{E}\left(G(\boldsymbol{\xi}, \cdot)\right)$ for $\int_\Omega G(\boldsymbol{\xi}(\omega), \cdot) d\mathbb{P}(\omega)$, which is in $L^2(\mathbb{R}^m, \mathbb{R}^m)$. Furthermore, let $v : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^p$, such that for almost all ξ, $v(\xi, \cdot) \in L^2(\mathbb{R}^m, \mathbb{R}^p)$. It will hold in the following :

$$\forall k \in \mathbb{N},\ \|v(\boldsymbol{\xi}^{k+1}, \cdot)\|^2 = \int_\Omega \|v(\boldsymbol{\xi}^{k+1}, \boldsymbol{\xi}(\omega))\|_p^2 d\mathbb{P}(\omega), \text{ with } \boldsymbol{\xi} \text{ independent from } \boldsymbol{\xi}^{k+1}.$$

2.2. **A first convergence proof.** We use here the classical Robbins-Siegmund's scheme to prove our convergence result (see [Robbins and Siegmund, 1971]).

Theorem 2.2. *(i) Assume that for almost all $\xi \in \mathbb{R}^m$, $u \mapsto j(u, \xi)$ is strongly convex with modulus B, uniformly in ξ, lower semicontinuous. Assume that j is a normal integrand on U^f which is a closed convex subset of $L^2(\mathbb{R}^m, \mathbb{R}^p)$. Then* (1) *has a unique solution denoted by u^*.*
(ii) Assume that there exist $b_1, b_2 > 0$ such that:

(4a)

$$\forall k \in \mathbb{N},\ r^k(\cdot) := \nabla_u j(u^k(\cdot), \cdot),\ \|r^k - \mathbb{E}\left(r^k(\boldsymbol{\xi}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}, \cdot)\right)\| \leq b_1 \epsilon^k \left(1 + \|r^k\|\right),$$

$$\forall x \in \mathbb{R}^m,\ \mathbb{E}\left((K^k(x, \boldsymbol{\xi}))^2\right) \leq b_2 \epsilon^k, \tag{4b}$$

with u^k generated by Algorithm 2.1.
(iii) Assume that $\nabla_u j(\cdot, \xi)$ is Lipschitz continuous with modulus L uniformly in ξ.
(iv) Assume that the sequences (ϵ^k) and (ρ^k) are such that:

$$\epsilon^k, \rho^k > 0, \quad \sum_{k \in \mathbb{N}} \rho^k \epsilon^k = +\infty, \quad \sum_{k \in \mathbb{N}} \rho^k (\epsilon^k)^2 < +\infty. \tag{5}$$

(v) Assume that

$$\nabla \mathcal{J}(u^*) = 0, \ \textit{and}\ \forall k \in \mathbb{N},\ 0 < \rho^k < \frac{B}{2 b_2 L^2}, \tag{6a}$$

or that

$$\nabla \mathcal{J}(u^*) \neq 0, \ \textit{and} \sum_{k \in \mathbb{N}} \epsilon^k (\rho^k)^2 < +\infty. \tag{6b}$$

Then (u^k) generated by Algorithm 2.1 a.s. strongly converges to the unique optimal solution of (1), *and $\mathcal{J}(u^k) \to \mathcal{J}(u^*)$ a.s., as k goes to infinity.*

Proof : We use the classical scheme of Robbins-Siegmund. For the simplicity of notation, let us define Π to be the projection on the closed convex U^f, and $r^k(\cdot) = \nabla_u j(u^k(\cdot), \cdot)$.

$$\begin{aligned}
\|u^{k+1} - u^*\|^2 =& \|\Pi\left(u^k - \rho^k r^k(\boldsymbol{\xi}^{k+1}) K^k(\boldsymbol{\xi}^{k+1}, \cdot)\right) - \Pi(u^*)\|^2, \\
\leq& \|u^k - u^* - \rho^k r^k(\boldsymbol{\xi}^{k+1}) K^k(\boldsymbol{\xi}^{k+1}, \cdot)\|^2, \\
=& \|u^k - u^*\|^2 + (\rho^k)^2 \|r^k(\boldsymbol{\xi}^{k+1})\|_p^2 \mathbb{E}\left(K^k(\boldsymbol{\xi}^{k+1}, \boldsymbol{\xi})^2 | \boldsymbol{\xi}^{k+1}\right) \\
& - 2\rho^k \epsilon^k \langle u^k - u^*, r^k(\boldsymbol{\xi}^{k+1}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}^{k+1}, \cdot)\rangle.
\end{aligned}$$

where those inequalities were obtained thanks to the nonexpansiveness property of the projection, and the Pythagore's equality. We now want to use assumptions (4a)–(4b).

$$\begin{aligned}
\|u^{k+1} - u^*\|^2 \leq& \|u^k - u^*\|^2 + b_2 \epsilon^k (\rho^k)^2 \|r^k(\boldsymbol{\xi}^{k+1})\|_p^2 \\
& - 2\rho^k \epsilon^k \langle u^k - u^*, r^k(\boldsymbol{\xi}^{k+1}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}^{k+1}, \cdot) - r^k\rangle \\
& - 2\rho^k \epsilon^k \langle u^k - u^*, r^k\rangle.
\end{aligned} \tag{7}$$

Recall that since j is strongly convex uniformly in ξ, $\mathcal{J}$ is also strongly convex with same modulus, which yields

$$\underbrace{\mathcal{J}(u^*) - \mathcal{J}(u^k)}_{\leq 0, \text{ by optimality}} + \langle r^k, u^k - u^* \rangle \geq \frac{B}{2} \|u^* - u^k\|^2.$$

Hence, one has:

$$-2\rho^k \epsilon^k \langle r^k, u^k - u^* \rangle \leq -B\rho^k \epsilon^k \|u^* - u^k\|^2.$$

We go back to (7). With the last inequality, we obtain

$$\begin{aligned} \|u^{k+1} - u^*\|^2 \leq & \left(1 - B\rho^k \epsilon^k\right) \|u^k - u^*\|^2 + b_2 \epsilon^k (\rho^k)^2 \|r^k(\boldsymbol{\xi}^{k+1})\|_p^2 \\ & - 2\rho^k \epsilon^k \langle u^k - u^*, r^k(\boldsymbol{\xi}^{k+1}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}^{k+1}, \cdot) - r^k \rangle. \end{aligned} \tag{8}$$

Using the Lipschitz property of $\nabla_u j$, and the classical inequality $(a+b)^2 \leq 2a^2 + 2b^2$, it holds that

$$\begin{aligned} \|r^k(\boldsymbol{\xi}^{k+1})\|_p^2 \leq & 2\|r^k(\boldsymbol{\xi}^{k+1}) - \nabla_u j(u^*(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1})\|_p^2 + 2\|\nabla_u j(u^*(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1})\|_p^2, \\ \leq & 2L^2 \|u^k(\boldsymbol{\xi}^{k+1}) - u^*(\boldsymbol{\xi}^{k+1})\|_p^2 + 2\|\nabla_u j(u^*(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1})\|_p^2. \end{aligned} \tag{9}$$

Gathering (8) and (9) yields

$$\begin{aligned} \|u^{k+1} - u^*\|^2 \leq & \left(1 - B\rho^k \epsilon^k\right) \|u^k - u^*\|^2 + 2b_2 L^2 \epsilon^k (\rho^k)^2 \|u^k(\boldsymbol{\xi}^{k+1}) - u^*(\boldsymbol{\xi}^{k+1})\|_p^2 \\ & + 2b_2 \epsilon^k (\rho^k)^2 \|\nabla_u j(u^*(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1})\|_p^2 \\ & - 2\rho^k \epsilon^k \langle u^k - u^*, r^k(\boldsymbol{\xi}^{k+1}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}^{k+1}, \cdot) - r^k \rangle. \end{aligned} \tag{10}$$

We now take in (10) the conditional expectation with respect to $\mathcal{F}^k := \sigma(\boldsymbol{\xi}^1, \dots \boldsymbol{\xi}^k)$. Since $\boldsymbol{\xi}^{k+1}$ is independent from $\mathcal{F}^k$, it yields:

$$\begin{aligned} \mathbb{E}\left(\|u^{k+1} - u^*\|^2 | \mathcal{F}^k\right) \leq & \left(1 - B\rho^k \epsilon^k + 2b_2 L^2 \epsilon^k (\rho^k)^2\right) \|u^k - u^*\|^2 \\ & + 2b_2 \epsilon^k (\rho^k)^2 \|\nabla \mathcal{J}(u^*)\|^2 \\ & - 2\rho^k \epsilon^k \langle u^k - u^*, \mathbb{E}\left(r^k(\boldsymbol{\xi}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}, \cdot)\right) - r^k \rangle, \\ \leq & \left(1 - B\rho^k \epsilon^k + 2b_2 L^2 \epsilon^k (\rho^k)^2\right) \|u^k - u^*\|^2 \\ & + 2b_2 \epsilon^k (\rho^k)^2 \|\nabla \mathcal{J}(u^*)\|^2 \\ & + 2\rho^k \epsilon^k \|u^k - u^*\| \|\mathbb{E}\left(r^k(\boldsymbol{\xi}) \frac{1}{\epsilon^k} K^k(\boldsymbol{\xi}, \cdot)\right) - r^k\|, \\ \leq & \left(1 - B\rho^k \epsilon^k + 2b_2 L^2 \epsilon^k (\rho^k)^2\right) \|u^k - u^*\|^2 \\ & + 2b_2 \epsilon^k (\rho^k)^2 \|\nabla \mathcal{J}(u^*)\|^2 \\ & + 2b_1 \rho^k (\epsilon^k)^2 \left(1 + \|\nabla \mathcal{J}(u^k)\|\right) \|u^k - u^*\|. \end{aligned} \tag{11}$$

We now use the inequality $a \leq a^2 + 1$, and the Lipschitz property of $\nabla \mathcal{J}$, and we obtain

$$\begin{aligned} \mathbb{E}\left(\|u^{k+1} - u^*\|^2 | \mathcal{F}^k\right) \leq & \left(1 - B\rho^k \epsilon^k + 2b_2 L^2 \epsilon^k (\rho^k)^2 + 2b_1 \rho^k (\epsilon^k)^2 (1 + L + \|\nabla \mathcal{J}(u^*)\|)\right) \|u^k - u^*\|^2 \\ & + 2b_1 \rho^k (\epsilon^k)^2 (1 + \|\nabla \mathcal{J}(u^*)\|) + 2b_2 \epsilon^k (\rho^k)^2 \|\nabla \mathcal{J}(u^*)\|^2. \end{aligned} \tag{12}$$

We now distinguish between two cases:

(1) If (6a) holds, by assumption, $-B\rho^k \epsilon^k + 2b_2 L^2 \epsilon^k (\rho^k)^2 < 0$. Inequality (12) can therefore be rewritten as:

$$\begin{aligned} \mathbb{E}\left(\|u^{k+1} - u^*\|^2 | \mathcal{F}^k\right) \leq & \left(1 + 2b_1 \rho^k (\epsilon^k)^2 (1 + L + \|\nabla \mathcal{J}(u^*)\|)\right) \|u^k - u^*\|^2 \\ & + 2b_1 \rho^k (\epsilon^k)^2 (1 + \|\nabla \mathcal{J}(u^*)\|) \\ & + \left(-B\rho^k \epsilon^k + 2b_2 L^2 \epsilon^k (\rho^k)^2\right) \|u^k - u^*\|^2. \end{aligned} \tag{13}$$

We can then use the Robbins-Siegmund lemma (see [Robbins and Siegmund, 1971]), which shows that (u^k) converges a.s. strongly to u^*.

(2) If (6b) holds, inequality (12) can be rewritten as:

$$\begin{aligned}\mathbb{E}\left(\|u^{k+1}-u^*\|^2|\mathcal{F}^k\right) \leq &\left(1+2b_2L^2\epsilon^k(\rho^k)^2+2b_1\rho^k(\epsilon^k)^2(1+L+\|\nabla\mathcal{J}(u^*)\|)\right)\|u^k-u^*\|^2\\ &+2b_1\rho^k(\epsilon^k)^2(1+\|\nabla\mathcal{J}(u^*)\|)+2b_2\epsilon^k(\rho^k)^2\|\nabla\mathcal{J}(u^*)\|^2\\ &-B\rho^k\epsilon^k\|u^k-u^*\|^2, \end{aligned} \tag{14}$$

and with our assumptions on the sequences, we can also apply the Robbins-Siegmund lemma, which shows that (u^k) converges a.s. strongly to u^*.

Since (u^k) strongly converges, it is bounded. Using the Lipschitz property of $\mathcal{J}$ over all bounded set, we get that $\mathcal{J}(u^k)\to\mathcal{J}(u^*)$ as k goes to infinity, and this completes the proof. □

Theorem 2.2 shows the role played by the constraints. It is quite strange to notice that we are able to prove the convergence without a decreasing sequence for the gradient steps ρ^k, provided that the gradient at the optimum is equal to zero. Other assumptions on the cost function j are usual ones, and assumptions on the so called kernels are not restrictive at all, as will be discussed in the next subsection. We provide in the next subsection another convergence proof, which leads to a quite similar result, with slightly different assumptions.

2.3. **Projection on a vector subspace.**

We prove the convergence of our algorithm in the case of a projection on a vector subspace, with slightly different assumptions, and a result even in the convex case.

Theorem 2.3. *(i) Assume that for almost all $\xi\in\mathbb{R}^m$, $u\mapsto j(u,\xi)$ is convex, lower semicontinuous. Assume that j is a normal integrand. If moreover $\mathcal{J}$ is coercive on U^f which is a closed sub vector space of $L^2(\mathbb{R}^m,\mathbb{R}^p)$, then (1) has solutions, and we denote by U^* the set of solutions.*
(ii) Assume that there exist $b_1,b_2>0$ such that:

(15a)

$$\forall k\in\mathbb{N},\ r^k(\cdot):=\nabla_u j(u^k(\cdot),\cdot),\ \|r^k-\mathbb{E}\left(r^k(\boldsymbol{\xi})\frac{1}{\epsilon^k}K^k(\boldsymbol{\xi},\cdot)\right)\|\leq b_1\epsilon^k\left(1+\|r^k\|\right),$$

$$\forall x\in\mathbb{R}^m,\ \mathbb{E}\left((K^k(x,\boldsymbol{\xi}))^2\right)\leq b_2\epsilon^k, \tag{15b}$$

(iii) Assume that the sequences (ϵ^k) and (ρ^k) are such that:

$$\epsilon^k,\rho^k>0,\quad \sum_{k\in\mathbb{N}}\epsilon^k\rho^k=+\infty,\quad \sum_{k\in\mathbb{N}}\rho^k(\epsilon^k)^2<+\infty,\quad \sum_{k\in\mathbb{N}}(\rho^k)^2\epsilon^k<+\infty. \tag{16}$$

If moreover j has linearly bounded gradients, i.e. there are $c,d>0$, such that for all $u\in\mathbb{R}^p$,

$$\forall\xi\in\mathbb{R}^m,\ \|\nabla_u j(u,\xi)\|_p\leq c\|u\|_p+d, \tag{17}$$

then the sequence (u^k) generated by Algorithm 2.1 is such that:

$$\lim_{k\to\infty}\mathcal{J}(u^k)=\mathcal{J}(u^*),\ a.s.$$

with $u^*\in U^*$, and every cluster point of (u^k) in the weak topology is in U^*.
(iv) Moreover, if j is strongly convex (in u) with modulus $B>0$, then U^* reduces to a singleton and (u^k) a.s. strongly converges to the unique optimal solution of (1).

Proof : The proof follows the scheme used by [Cohen and Culioli, 1990].
Let us denote by u^* some optimal solution of (1). Let us define a Lyapunov function $\Lambda: L^2(\mathbb{R}^m,\mathbb{R}^p)\to\mathbb{R}$ by:

$$\forall u\in L^2,\ \Lambda(u):=\frac{1}{2}\|u-u^*\|^2.$$

We study now the variation of the Lyapunov function between two iterations k and $k+1$.

$$\begin{aligned}\delta^{k+1} := \Lambda(u^{k+1}) - \Lambda(u^k) =& \frac{1}{2}\|u^{k+1} - u^*\|^2 - \frac{1}{2}\|u^k - u^*\|^2, \\ =& \frac{1}{2}\|u^{k+1} - u^k\|^2 + \langle u^{k+1} - u^k, u^k - u^* \rangle,\end{aligned}$$

using Pythagore's equality. By the nonexpansiveness property of the projection, it holds by definition of u^{k+1} that:

$$\|u^{k+1} - u^k\| \leq \rho^k \|\nabla_u j(u^k(\boldsymbol{\xi}^{k+1}), \boldsymbol{\xi}^{k+1}) K^k(\boldsymbol{\xi}^{k+1}, \cdot)\|.$$

We now define $G^k(\cdot,\cdot) = \frac{1}{\epsilon^k} K^k(\cdot,\cdot)$, $r^k(\cdot) = \nabla_u j(u^k(\cdot), \cdot)$, and $f^k(\cdot) = \epsilon^k r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot)$. We focus on $\langle u^{k+1} - u^k, u^k - u^* \rangle$. For the simplicity of our notations, we denote by Π the projection on U^f.

$$\begin{aligned}\langle u^{k+1} - u^k, u^k - u^* \rangle =& \langle \Pi\left(u^k - \rho^k f^k\right) - u^k, u^k - u^* \rangle, \\ =& -\rho^k \epsilon^k \langle \Pi\left(r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot)\right), u^k - u^* \rangle, \\ =& -\rho^k \epsilon^k \langle r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot), u^k - u^* \rangle.\end{aligned}$$

Hence,

$$\begin{aligned}\delta^{k+1} \leq& \frac{(\rho^k \epsilon^k)^2}{2} \|r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot)\|^2 - \rho^k \epsilon^k \langle r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot), u^k - u^* \rangle, \\ \leq& \frac{b_2 (\rho^k)^2 \epsilon^k}{2} \|r^k(\boldsymbol{\xi}^{k+1})\|_p^2 + \rho^k \epsilon^k \langle r^k - r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot), u^k - u^* \rangle \\ &+ \rho^k \epsilon^k \langle r^k, u^* - u^k \rangle. \end{aligned} \tag{18}$$

The second inequality is due to the assumption (15b) on the kernels K^k. Using convexity of $\mathcal{J}$, one has:

$$\langle r^k, u^* - u^k \rangle \leq \mathcal{J}(u^*) - \mathcal{J}(u^k) \leq 0. \tag{19}$$

Gathering (18) and (19) yields:

$$\begin{aligned}\Lambda(u^{k+1}) - \Lambda(u^k) \leq& \frac{b_2 (\rho^k)^2 \epsilon^k}{2} \|r^k(\boldsymbol{\xi}^{k+1})\|_p^2 + \rho^k \epsilon^k \langle r^k - r^k(\boldsymbol{\xi}^{k+1}) G^k(\boldsymbol{\xi}^{k+1}, \cdot), u^k - u^* \rangle \\ &+ \rho^k \epsilon^k \left(\mathcal{J}(u^*) - \mathcal{J}(u^k)\right).\end{aligned} \tag{20}$$

We take now the conditional expectation in (20) with respect to $\mathcal{F}^k := \sigma(\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^k)$, i.e., with respect to the past draws. Since $\boldsymbol{\xi}^{k+1}$ is independent from the past draws, it yields

$$\begin{aligned}\mathbb{E}\left(\Lambda(u^{k+1}) - \Lambda(u^k) | \mathcal{F}^k\right) \leq& \frac{b_2 (\rho^k)^2 \epsilon^k}{2} \|r^k\|^2 + \rho^k \epsilon^k \langle r^k - \mathbb{E}\left(r^k(\boldsymbol{\xi}) G^k(\boldsymbol{\xi}, \cdot)\right), u^k - u^* \rangle \\ &+ \rho^k \epsilon^k \left(\mathcal{J}(u^*) - \mathcal{J}(u^k)\right).\end{aligned} \tag{21}$$

Using the linearly bounded gradient assumption on $\mathcal{J}$, one has also, with the classical inequality $(a+b)^2 \leq 2a^2 + 2b^2$

$$\|r^k\|^2 \leq c_1 \|u^k - u^*\|^2 + c_2,$$

with $c_1, c_2 > 0$. Thus,

$$\begin{aligned}\mathbb{E}\left(\Lambda(u^{k+1}) - \Lambda(u^k) | \mathcal{F}^k\right) \leq& \frac{b_2 c_1 (\rho^k)^2 \epsilon^k}{2} \|u^k - u^*\|^2 + \frac{b_2 c_2 (\rho^k)^2 \epsilon^k}{2} \\ &+ \rho^k \epsilon^k \langle r^k - \mathbb{E}\left(r^k(\boldsymbol{\xi}) G^k(\boldsymbol{\xi}, \cdot)\right), u^k - u^* \rangle \\ &+ \rho^k \epsilon^k \left(\mathcal{J}(u^*) - \mathcal{J}(u^k)\right).\end{aligned} \tag{22}$$

We now use Cauchy Schwartz's inequality in (22):

$$\begin{aligned}\mathbb{E}\left(\Lambda(u^{k+1})-\Lambda(u^k)|\mathcal{F}^k\right)\leq&\frac{b_2c_1(\rho^k)^2\epsilon^k}{2}\|u^k-u^*\|^2+\frac{b_2c_2(\rho^k)^2\epsilon^k}{2}\\&+\rho^k\epsilon^k\|r^k-\mathbb{E}\left(r^k(\boldsymbol{\xi})G^k(\boldsymbol{\xi},\cdot)\right)\|\|u^k-u^*\|\\&+\rho^k\epsilon^k\left(\mathcal{J}(u^*)-\mathcal{J}(u^k)\right).\end{aligned}\tag{23}$$

We use the assumption on kernels (15a), and it yields:

$$\begin{aligned}\mathbb{E}\left(\Lambda(u^{k+1})-\Lambda(u^k)|\mathcal{F}^k\right)\leq&\frac{b_2c_1(\rho^k)^2\epsilon^k}{2}\|u^k-u^*\|^2+\frac{b_2c_2(\rho^k)^2\epsilon^k}{2}\\&+b_1\rho^k(\epsilon^k)^2\left(1+\|\nabla\mathcal{J}(u^k)\|\right)\|u^k-u^*\|\\&+\rho^k\epsilon^k\left(\mathcal{J}(u^*)-\mathcal{J}(u^k)\right).\end{aligned}\tag{24}$$

Assumption (17) implies that there exist two scalars $c_3, c_4 > 0$ such that:

$$\forall u\in L^2,\ \|\nabla\mathcal{J}(u)\|\leq c_3\|u\|+c_4.$$

By the last inequality and the classical inequality $x \leq x^2 + 1$, we obtain:

$$\begin{aligned}\mathbb{E}\left(\Lambda(u^{k+1})-\Lambda(u^k)|\mathcal{F}^k\right)\leq&\frac{b_2c_1(\rho^k)^2\epsilon^k}{2}\|u^k-u^*\|^2+\frac{b_2c_2(\rho^k)^2\epsilon^k}{2}\\&+b_1\rho^k(\epsilon^k)^2(1+c_3+c_4)\|u^k-u^*\|^2+b_1\rho^k(\epsilon^k)^2(1+c_4)\\&+\rho^k\epsilon^k\left(\mathcal{J}(u^*)-\mathcal{J}(u^k)\right).\end{aligned}$$

By definition of Λ, we have finally:

$$\mathbb{E}\left(\Lambda(u^{k+1})-\Lambda(u^k)|\mathcal{F}^k\right)\leq\alpha^k\Lambda(u^k)+\beta^k+\rho^k\epsilon^k\left(\mathcal{J}(u^*)-\mathcal{J}(u^k)\right),\tag{25}$$

with $\alpha^k := b_2c_1(\rho^k)^2\epsilon^k + 2b_1\rho^k(\epsilon^k)^2(1 + c_3 + c_4)$ and $\beta^k := \frac{b_2c_2(\rho^k)^2\epsilon^k}{2} + b_1\rho^k(\epsilon^k)^2(1 + c_4)$. (α^k) and (β^k) are by assumption summable sequences. Let us now take the expectation in (25), and define $y^k := \mathbb{E}\left(\Lambda(u^k)\right)$. By optimality of u^*,

$$y^{k+1}-y^k\leq\alpha^ky^k+\beta^k.\tag{26}$$

Using Lemma 2.8, it shows that (y^k) is bounded, by, say $M > 0$. We prove that $(\Lambda(u^k))$ is a convergent quasi-martingale. Indeed:

- $(\Lambda(u^k))$ is by definition adapted to $(\mathcal{F}^k)$.
- By definition, $\Lambda(u^k) \geq 0$ for all $k \in \mathbb{N}$, i.e., $\inf_{k\in\mathbb{N}} \mathbb{E}\left(\Lambda(u^k)\right) > -\infty$.
- Let us consider $C_k := \{\mathbb{E}\left(\Lambda(u^{k+1}) - \Lambda(u^k)|\mathcal{F}^k\right) > 0\}$. It is clear that 1_{C_k} is $\mathcal{F}^k$ measurable. Using (25), we have:

$$\begin{aligned}\sum_{k\in\mathbb{N}}\mathbb{E}\left(1_{C_k}\times(\Lambda(u^{k+1})-\Lambda(u^k))\right)\leq&\sum_{k\in\mathbb{N}}\mathbb{E}\left(1_{C_k}\times\mathbb{E}\left(\Lambda(u^{k+1})-\Lambda(u^k)|\mathcal{F}^k\right)\right),\\\leq&\sum_{k\in\mathbb{N}}\mathbb{E}\left(1_{C_k}\times(\alpha^k\Lambda(u^k)+\beta^k)\right),\\\leq&\sum_{k\in\mathbb{N}}(\alpha^kM+\beta^k),\\<&\infty,\end{aligned}$$

  since the sequences are summable.
- It is also clear that $\sup_{k\in\mathbb{N}} \mathbb{E}\left(\Lambda(u^k)^-\right) < \infty$, and consequently, using a result of [Métivier, 1982] (pp. 49-51), $(\Lambda(u^k))$ is a quasi-martingale and converges a.s. to some integrable random variable. Hence, it is a.s. bounded, and by definition, (u^k) and (r^k) are two a.s. bounded sequences in L^2.

We now prove that $(\mathcal{J}(u^k))$ a.s. converges to $\mathcal{J}(u^*)$. Coming back to (25) and taking the expectation gives:

$$\rho^k\epsilon^k\mathbb{E}\left(\mathcal{J}(u^k)-\mathcal{J}(u^*)\right)\leq\alpha^ky^k+\beta^k+y^k-y^{k+1}.$$

We sum this inequality for $k = 0, \dots, n$:

$$\begin{aligned}\sum_{k=0}^{n} \rho^k \epsilon^k \mathbb{E}\left(\mathcal{J}(u^k) - \mathcal{J}(u^*)\right) \leq & y^0 - y^{n+1} + \sum_{k=0}^{n} (\alpha^k M + \beta^k), \\ \leq & M + M \sum_{k=0}^{n} \alpha^k + \sum_{k=0}^{n} \beta^k. \end{aligned} \tag{27}$$

We make $n \to \infty$:

$$\sum_{k \in \mathbb{N}} \rho^k \epsilon^k \mathbb{E}\left(\mathcal{J}(u^k) - \mathcal{J}(u^*)\right) < \infty.$$

By optimality, all the terms under the expectation are a.s. nonnegative. Thus:

$$\sum_{k \in \mathbb{N}} \rho^k \epsilon^k \left(\mathcal{J}(u^k) - \mathcal{J}(u^*)\right) < \infty. \tag{28}$$

We now want to use Lemma 2.9. Let $l \in \mathbb{N}$. By convexity,

$$\begin{aligned}\mathcal{J}(u^l) - \mathcal{J}(u^{l+1}) \leq & \langle \nabla \mathcal{J}(u^l), u^l - u^{l+1} \rangle, \\ = & \rho^l \epsilon^l \langle r^l, \Pi_{U^f}\left(r^l(\boldsymbol{\xi}^{l+1}) G^l(\boldsymbol{\xi}^{l+1}, \cdot)\right) \rangle. \end{aligned} \tag{29}$$

We take now the conditional expectation with respect to $\mathcal{F}^l$:

$$\begin{aligned}\mathcal{J}(u^l) - \mathbb{E}\left(\mathcal{J}(u^{l+1}) | \mathcal{F}^l\right) \leq & \rho^l \epsilon^l \langle r^l, \Pi_{U^f}\left(\mathbb{E}\left(r^l(\boldsymbol{\xi}) \frac{1}{\epsilon^l} K^l(\boldsymbol{\xi}, \cdot)\right)\right) \rangle, \\ \leq & \rho^l \epsilon^l \|r^l\| \left(\|\mathbb{E}\left(r^l(\boldsymbol{\xi}) \frac{1}{\epsilon^l} K^l(\boldsymbol{\xi}, \cdot)\right) - r^l\| + \|r^l\|\right), \\ \leq & \rho^l \epsilon^l R \left(b_1 \epsilon^l (1 + R) + R\right), \\ \leq & \rho^l \epsilon^l \delta, \end{aligned} \tag{30}$$

with $\delta > 0$, since we already know that $(\|r^k\|)$ is bounded by, say, some $R > 0$.Hence, we can apply Lemma 2.9, with (28) and (30), and with our sampling space for probability space, and with $\gamma^k = \epsilon^k \rho^k$. It yields

$$\lim_{k \to \infty} \mathcal{J}(u^k) = \mathcal{J}(u^*) \tag{31}$$

Let $\bar{u}$ be a cluster point of (u^k). Hence there is some subsequence $(u_{\phi(k)})$ which converges to $\bar{u}$. Since U^f is a closed subspace, $\bar{u} \in U^f$, and by lower semi-continuity of $\mathcal{J}$, it holds:

$$\mathcal{J}(\bar{u}) \leq \liminf_{k \to \infty} \mathcal{J}(u^{\phi(k)}) = \mathcal{J}(u^*),$$

hence, $\bar{u} \in U^*$.
Suppose now that j is strongly convex with modulus $B > 0$. In this case, U^* reduces to a singleton $\{u^*\}$. By definition,

$$\mathcal{J}(u^k) - \mathcal{J}(u^*) \geq \langle \nabla \mathcal{J}(u^*), u^k - u^* \rangle + \frac{B}{2} \|u^* - u^k\|^2 \tag{32}$$

By optimality, $\langle \nabla \mathcal{J}(u^*), u^k - u^* \rangle \geq 0$. (31) gives therefore the strong convergence of (u^k) to u^*, and it completes the proof. □

Remark 2.4 (Choice of ϵ^k and ρ^k)**.** For the choice of the two sequences ϵ^k and ρ^k, we can take $\rho^k = k^{-\alpha}$ and $\epsilon^k = k^{-\beta}$, with $\beta \in [1/2, 1]$ and $\alpha \in \left[\frac{1-\beta}{2}, 1 - \beta\right]$, which yields to the assumptions (16). For example, $\alpha = 1/3$ and $\beta = 2/3$ is a good choice.

Remark 2.5 (Measurability of the stepsizes and kernels)**.** The stepsizes (ρ^k) and (ϵ^k), as the kernels K^k can be taken such that they are adapted to the sequences of draws $(\boldsymbol{\xi}^{k+1})$. For all $k \in \mathbb{N}$, if ϵ^k, ρ^k and $K^k(\cdot, \cdot)$ are measurable with respect to $\sigma(\boldsymbol{\xi}^1, \dots, \boldsymbol{\xi}^k)$, the proofs are always true. The Robbins-Siegmund Lemma holds true with this measurability assumptions, and so do the results involving the quasi-martingale result of Métivier.

Remark 2.6 (Choice of Kernels)**.** Analogously, if we take the kernel to be $K^k(x,y) = \delta(x) K\left(\frac{x-y}{\epsilon^{k\,1/m}}\right)$, with $K : \mathbb{R}^m \to \mathbb{R}$ such that

$$\int_{\mathbb{R}^m} K(x)dx = 1, \quad K(x) = K(-x), \text{ for all } x \in \mathbb{R}^m,$$

assumptions (15a)–(15b) will be satisfied with all the usual laws for $\boldsymbol{\xi}$, by taking if $\boldsymbol{\xi}$ has a density $p(\xi)$ over $\mathbb{R}^m$, $\delta(x) = \frac{1}{p(x)}$.

Example 2.7. We here provide an illustration of our assumptions (16) on the two sequences (ρ^k), (ϵ^k). This example has to be taken as an illustration, and absolutely nothing more.
Consider $\Xi = [0,1]$ to be the noise space, and $\boldsymbol{\xi}$ a real random variable with uniform law on Ξ. Let $\epsilon^k = 1/(k+1)$ be a sequence of decreasing steps, and let us define $K^k(x,y) = 1_{|x-y| \leq \epsilon^k/2}$ for all $x, y \in \Xi$. Such kernels will produce controls differentiable almost everywhere.
Let us now define the indexes j_n such that $j_0 = 0$, and for $n \geq 1$, j_n is such that:

$$\sum_{k=j_{n-1}}^{j_n - 1} \epsilon^k \leq 1, \quad \sum_{k=j_{n-1}}^{j_n} \epsilon^k > 1.$$

Since $\sum_{k\in\mathbb{N}} \epsilon^k = +\infty$, this sequence is well defined. Consider now the sequence (ξ^k) such that for all $k \in \mathbb{N}$, $\xi^{j_k+1} = \frac{\epsilon^{j_k}}{2}$, $\xi^{j_k+2} = \frac{\epsilon^{j_k+1}+\epsilon^{j_k}}{2} + \xi^{j_k+1}$, $\xi^{j_k+3} = \frac{\epsilon^{j_k+2}+\epsilon^{j_k+1}}{2} + \xi^{j_k+2}, \ldots, \xi^{j_{k+1}} = \sum_{n=j_k}^{j_{k+1}-2} \epsilon^n + \frac{\epsilon^{j_{k+1}-1}}{2}$, $\xi^{j_{k+1}+1} = \frac{\epsilon^{j_{k+1}}}{2}$, and so on. This sequence is also well defined. This construction is illustrated by figure 1, until the j_{k+1}st point.

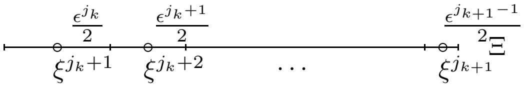

$\frac{\epsilon^{j_k}}{2}$
$\frac{\epsilon^{j_k+1}}{2}$
$\frac{\epsilon^{j_{k+1}-1}}{2}$
Ξ
ξ^{j_k+1}
ξ^{j_k+2}
$\ldots$
$\xi^{j_{k+1}}$

FIGURE 1. Quasi-Monte Carlo construction to cover $\Xi = [0,1]$

Consider now the following algorithm, with a given nonnegative sequence (ρ^k):

$$\forall \xi \in \Xi, \; u^{k+1}(\xi) = u^k(\xi) - \rho^k f(\xi^{k+1}) K^k(\xi^{k+1}, \xi),$$

with $f : \Xi \to \mathbb{R}$ a Lipschitz continuous mapping with modulus L. By definition of kernels K^k, this algorithm modifies the function u^k only on the little ball of radius $\epsilon^k/2$ and centered on ξ^{k+1}. This algorithm is a quasi Monte Carlo version of our preceding algorithm 2.1.
We now study this algorithm only on $\xi = 0$, and denote for simplicity $u^k(0)$ by v^k. It consists by definitions in:

$$v^{j_{k+1}} = v^{j_k} - \rho_{j_k} r^{j_k}, \tag{33}$$

with $r^{j_k} = f(\xi^{j_{k+1}}) K^{j_k}(\xi^{j_{k+1}}, 0)$. Notice first that r^{j_k} can be seen as a perturbation of the function f, taken in 0. To ensure the convergence of algorithm (33), we can therefore use the general convergence theorems for stochastic algorithms (see e.g. [Bertsekas and Tsitsiklis, 1996]). A common condition is

$$\sum_{k\in\mathbb{N}} \rho^{j_k} = +\infty, \quad \sum_{k\in\mathbb{N}} (\rho^{j_k})^2 < +\infty. \tag{34}$$

We now try to have $\rho_{j_k} = \frac{1}{k}$, which would be sufficient. By definition, we have $\sum_{n=j_k}^{j_{k+1}} \epsilon^k \simeq 1$.

$$\begin{aligned}\sum_{n=j_k}^{j_{k+1}} \epsilon^n \simeq & \int_{j_k}^{j_{k+1}} \frac{1}{x} dx, \\ = & \left[\log(x)\right]_{j_k}^{j_{k+1}}, \\ = & \log\left(\frac{j_{k+1}}{j_k}\right).\end{aligned}$$

Hence, we want to obtain $\log(\frac{j_{k+1}}{j_k}) = 1$, i.e., $j_{k+1} = j_k e$.With $j_0 = 1$, it yields $j_k = e^k$. Hence, $\rho^n = \frac{1}{\log(n)}$, for all $n \in \mathbb{N}$. With $\epsilon^n = \frac{1}{n}$ for all $n \in \mathbb{N}$, the assumption (16) is therefore satisfied.

2.4. **A generalization of the open-loop stochastic gradient algorithm.** In this section, we consider again the same problem (1), with the particular feasible set:

$$\begin{aligned}U^f := \{u : \mathbb{R}^m \to \mathbb{R}^p \ & : u \ \sigma(\{\emptyset, \mathbb{R}^m\}) - \text{measurable}, \\ & u \in L^2(\Xi, \mathbb{R}^p), \ u(\xi) \in \Gamma(\xi) \ \text{ a.s.}\}.\end{aligned} \tag{35}$$

with Γ a closed convex measurable mapping from $\mathbb{R}^m$ to R^p. U^f defines therefore the constant controls $u \in \mathbb{R}^p$ such that $u \in \Gamma$, where Γ abusively defines the range of the mapping Γ, which is a closed convex of $\mathbb{R}^p$. Problem (1) becomes therefore an open-loop problem, equivalent to:

$$\begin{aligned}& \min_{u \in \mathbb{R}^p} \mathbb{E}\left(j(u, \boldsymbol{\xi})\right), \\ & \text{s.t. } u \in \Gamma,\end{aligned} \tag{36}$$

On the other hand, the updating step of algorithm 2.1 becomes:

$$u^{k+1} = \Pi_\Gamma\left(u^k - \rho^k \nabla_u j(u^k, \boldsymbol{\xi}^{k+1}) \mathbb{E}\left(K^k(\boldsymbol{\xi}^{k+1}, \boldsymbol{\xi}) | \boldsymbol{\xi}^{k+1}\right)\right). \tag{37}$$

Assume now that $\boldsymbol{\xi}$ has a smooth density p w.r.t. the Lebesgue measure. Take a kernel K^k defined as in Remark 2.6, by:

$$\forall (x, y) \in (\mathbb{R}^m)^2, \ K^k(x, y) = \frac{1}{p(x)} K\left(\frac{x - y}{(\epsilon^k)^{1/m}}\right),$$

with K verifying the kernel assumptions of Remark 2.6, and $K(x) = 0$ for all x s.t. $\|x\|_m > 1$. Then, we obtain:

$$\begin{aligned}\forall x \in \mathbb{R}^m, \ \mathbb{E}\left(K^k(x, \boldsymbol{\xi})\right) = & \int_{\mathbb{R}^m} \frac{p(\xi)}{p(x)} K\left(\frac{\xi - x}{(\epsilon^k)^{1/m}}\right) d\xi, \\ = & \epsilon^k \int_{\mathbb{R}^m} \frac{p(x + (\epsilon^k)^{1/m} y)}{p(x)} K(y) dy, \\ = & \epsilon^k + o\left((\epsilon^k)^2\right),\end{aligned}$$

where the last equations are obtained through a change of variables and the Taylor formula applied to p. Hence, we obtain the classical stochastic gradient algorithm,

$$u^{k+1} = \Pi_\Gamma\left(u^k - \rho^k \epsilon^k \nabla_u j(u^k, \boldsymbol{\xi}^{k+1}) + o\left(\rho^k (\epsilon^k)^2\right)\right).$$

with decreasing steps $\epsilon^k \rho^k$, and some additional perturbation converging quickly to 0.

2.5. **Technical Lemmas.** We here provide two technical lemmas we use in the preceding convergence proof of theorem 2.3.

Lemma 2.8. *Let $(x_k)_{k\in\mathbb{N}}$ be a sequence of nonnegative real numbers. Let $(\alpha_k)_{k\in\mathbb{N}}$ and $(\beta_k)_{k\in\mathbb{N}}$ be sequences of nonnegative real numbers such that $\sum_{k\in\mathbb{N}}\alpha_k < +\infty$ and $\sum_{k\in\mathbb{N}}\beta_k < +\infty$. If we have:*

$$\forall k \in \mathbb{N},\ x_{k+1} - x_k \leq \alpha_k x_k + \beta_k,$$

then the sequence $(x_k)_{k\in\mathbb{N}}$ is bounded.

The proof can be found in [Cohen, 1984].

Lemma 2.9. *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be some probability space, equipped with a filter $(\mathcal{F}^k)$. Let $\mathcal{J}$ be a real valued mapping from an Hilbert space H. Let $(u^k)_{k\in\mathbb{N}}$ be a sequence of random variables with values in H, such that for all $k \in \mathbb{N}$, u^k is $\mathcal{F}^k$-measurable, and $(\gamma^k)_{k\in\mathbb{N}}$ a sequence of nonnegative real numbers such that:*

(i) $\sum_{k\in\mathbb{N}}\gamma^k = +\infty$,
(ii) $\exists\mu \in \mathbb{R}$, $\sum_{k\in\mathbb{N}}\gamma^k\left(\mathcal{J}(u^k) - \mu\right) < +\infty$, *and* $\forall k \in \mathbb{N}$, $\mathcal{J}(u^k) - \mu \geq 0$, *a.s.*
(iii) $\exists\delta > 0$, $\forall k \in \mathbb{N}$, $\mathcal{J}(u^k) - \mathbb{E}\left(\mathcal{J}(u^{k+1})|\mathcal{F}^k\right) \leq \delta\gamma^k$, *a.s.*

Then $(\mathcal{J}(u^k))_{k\in\mathbb{N}}$ a.s. converges to μ.

Proof : For all $\alpha \in \mathbb{R}$, let us define the subset N_α of $\mathbb{N}$ such that:

$$N_\alpha := \left\{k \in \mathbb{N}\ :\ \mathcal{J}(u^k) - \mu \leq \alpha,\ \text{a.s.}\right\}.$$

We will also denote by N_α^c the complementary set of N_α in $\mathbb{N}$. Assumptions $(i - ii)$ imply that N_α is not finite.
Following (ii), we have:

$$+\infty > \sum_{k\in\mathbb{N}}\gamma^k\left(\mathcal{J}(u^k) - \mu\right) \geq \sum_{k\in N_\alpha^c}\gamma^k\left(\mathcal{J}(u^k) - \mu\right) \geq \alpha\sum_{k\in N_\alpha^c}\gamma^k.$$

It proves that for all $\beta > 0$, there is some $n_\beta \in \mathbb{N}$ such that $\sum_{k\in N_\alpha^c,\ k\geq n_\beta}\gamma^l \leq \beta$.
Let $\epsilon > 0$. Take $\alpha = \epsilon/2$ and $\beta = \epsilon/(2\delta)$. For all $k \geq n_\beta$, we have two possibilities:

- If $k \in N_\alpha$, then $\mathcal{J}(u^k) - \mu \leq \alpha < \epsilon$.
- If $k \in N_\alpha^c$, let m be the smallest element of N_α such that $m \geq k$ (we know that it exists since N_α is not finite). We can hence write:

$$\begin{aligned}\mathcal{J}(u^k) - \mu =& \mathcal{J}(u^k) - \mathbb{E}\left(\mathcal{J}(u^m)|\mathcal{F}^k\right) + \mathbb{E}\left(\mathcal{J}(u^m)|\mathcal{F}^k\right) - \mu\\ =&\mathbb{E}\left(\sum_{l=k}^{m-1}\mathcal{J}(u^l) - \mathbb{E}\left(\mathcal{J}(u^{l+1})|\mathcal{F}^l\right)|\mathcal{F}^k\right) + \mathbb{E}\left(\mathcal{J}(u^m)|\mathcal{F}^k\right) - \mu,\\ \leq&\delta\left(\sum_{l=k}^{m-1}\gamma^l\right) + \alpha \leq \delta\left(\sum_{l\in N_\alpha^c,\ l\geq n_\beta}\gamma^l\right) + \alpha \leq \epsilon.\end{aligned}$$

□

3. Numerical applications

We now give a few numerical applications of our algorithm. The first thing to be decided is to give a stopping test to Algorithm 2.1. Many tests can be implemented, but a good one is to give a maximal number of iterations... or in the *not projected* case to compute the norm of the true gradient $\|\nabla\mathcal{J}(u^k)\|$ and to compare it with a given threshold.

In the following, we will consider the kernels and sequences for all $k \in \mathbb{N}$ as:

- $\forall x \in \mathbb{R}^m,\ K^k(x, \cdot) := \frac{1}{\sqrt{\pi}}e^{-\left(\frac{\cdot - x}{\epsilon^k}\right)^2}$,
- $\epsilon^k = \frac{1}{k^\alpha}$,

- $\rho^k = \frac{1}{k^\beta}$,

with uniform laws on the noises. Our algorithm is parametrized by two nonnegative numbers: α and β respectively for the window size and the descent size of the stochastic gradient.

Remark 3.1 (Convergence speed)**.** For all the following examples, the *convergence speed* stands for the graph representing the difference $(\mathcal{J}(u^k) - \mathcal{J}(u^*))$ along the iterations.

3.1. **Least-Square problem.** Let us here consider the case of estimating on $[0, 1]$ the real function $x \mapsto \sin\left(\frac{100}{x+1}\right)$. We consider the following cost function:

$$\forall u, x \in \mathbb{R},\ j(u, x) = \left(u - \sin\left(\frac{100}{x+1}\right)\right)^2 \tag{38}$$

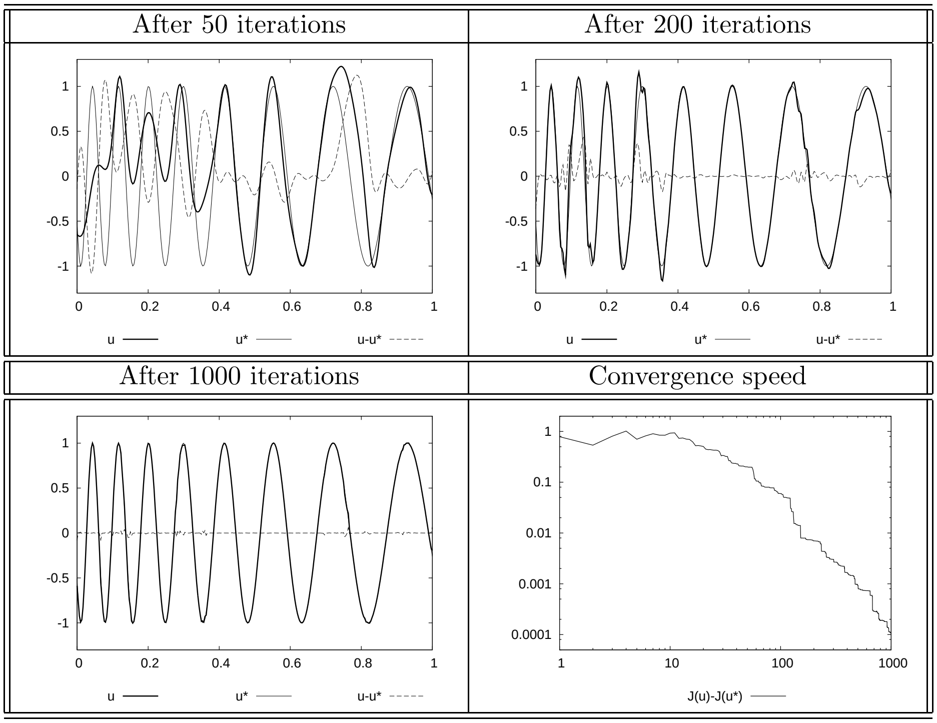
Let $\boldsymbol{\xi}$ be a real random variable following the uniform law on $[0, 1]$. We define $\mathcal{J}$ to be:

$$\forall u \in L^2([0, 1], \mathbb{R}),\ \mathcal{J}(u) = \mathbb{E}\left(j(u(\boldsymbol{\xi}), \boldsymbol{\xi})\right).$$

The gradient of j with respect to u is:

$$\forall u, x \in \mathbb{R},\ \nabla_u j(u, x) = 2\left(u - \sin\left(\frac{100}{x+1}\right)\right).$$

We now apply our algorithm to the problem of minimizing $\mathcal{J}$. Figure 2 shows u^k obtained by algorithm 2.1 respectively after 50, 200 and 1000 iterations. It also shows the optimal feedback called u^* and the error $\|u^k(\xi) - u^*(\xi)\|_p$. The last graph shows the convergence speed of the algorithm.



After 50 iterations

After 200 iterations

After 1000 iterations

Convergence speed

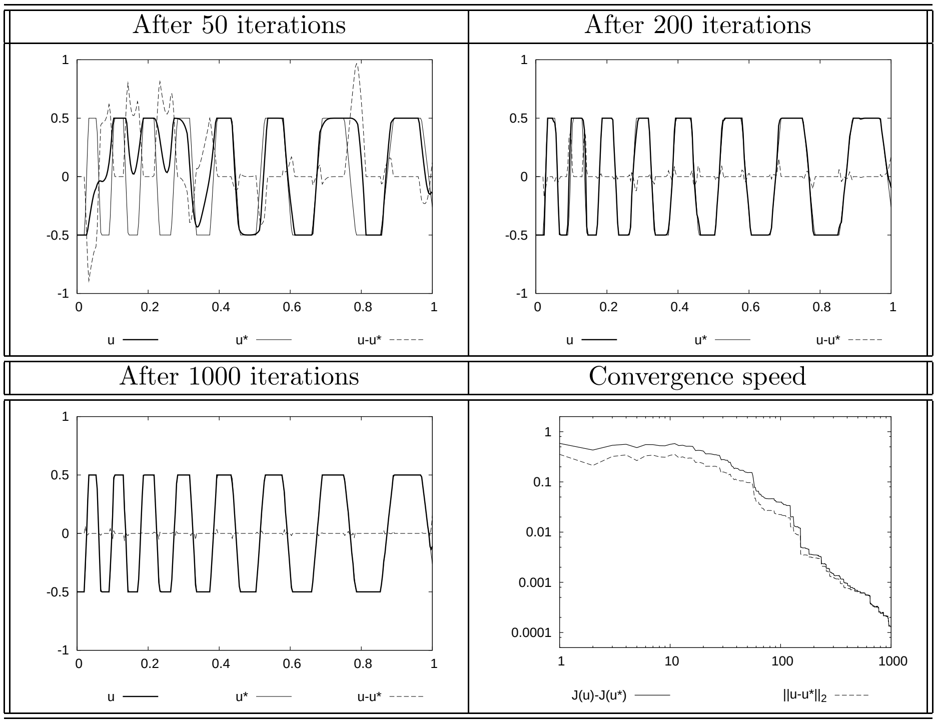
Figure 2. Least Square Problem, feedback along the iterations, and convergence speed

It is clear that with the iterations, $[0,1]$ becomes more and more correctly explored by our random draws of $\boldsymbol{\xi}$, and hence the feedback converges. Very quickly, the general behaviour of u^* is well captured, and then the finest behaviour is fitted with more iterations.

3.2. **Constrained Least-Square problem.** We take exactly the same problem, but with an additional bounding constraint. $U^f = \{u : [0,1] \to \mathbb{R} \ : \ -1/2 \leq u(\cdot) \leq 1/2\}$. The projection consists therefore only in bounding the maximal values of u^k. Let us denote $[x]_a^b = \min(\max(x,a),b)$. Then it holds:

$$\forall u \in L^2([0,1],\mathbb{R}), \ \Pi_{U^f}(u)(\cdot) = [u(\cdot)]_{-1/2}^{1/2}.$$

The evolution of the control is showed by Figure 3. Exactly as before, we see that our algorithm provides a good solution to the minimization problem.


After 50 iterations

After 200 iterations

After 1000 iterations

Convergence speed

Figure 3. Truncated Least Square Problem, feedback along the iterations, and convergence speed

3.3. **Reservoir management.** After the previous academic examples, we give a more practical example. We want to manage a reservoir. We give two examples, the first one with a single time period, and the second one with two time periods, and two sequential decisions.

3.3.1. *Single time period.* The cost function is here given by:

$$\forall \xi \in [\underline{x}, \overline{x}] \,, \ \forall u \in [0,s], \ j\,(u,\xi) = -\xi u - \sqrt{\epsilon + s - u}, \tag{39}$$

with given thresholds $\underline{x}, \overline{x}, s$. s represents the available stock in the reservoir, u is the control, i.e., the quantity we will produce, and ξ is a proportional selling price (which will be the random part of the system). j is therefore composed of two terms: $u\xi$ which represents the sales profit, and $\sqrt{\epsilon + s - u}$ which is the value at the end of the game. We take $\boldsymbol{\xi}$ a real random variable with uniform law on $[\underline{x}, \overline{x}]$, and assume s to be fixed. Hence the criterion to be minimized is given by:

$$\forall u \in L^2([\underline{x}, \overline{x}], \mathbb{R}),\ \mathcal{J}(u) = \mathbb{E}\left(j(u(\boldsymbol{\xi}), \boldsymbol{\xi})\right).$$

The gradient is:

$$\nabla \mathcal{J}(u)(\xi) = -\xi + \frac{1}{2\sqrt{\epsilon + s - u(\xi)}},$$

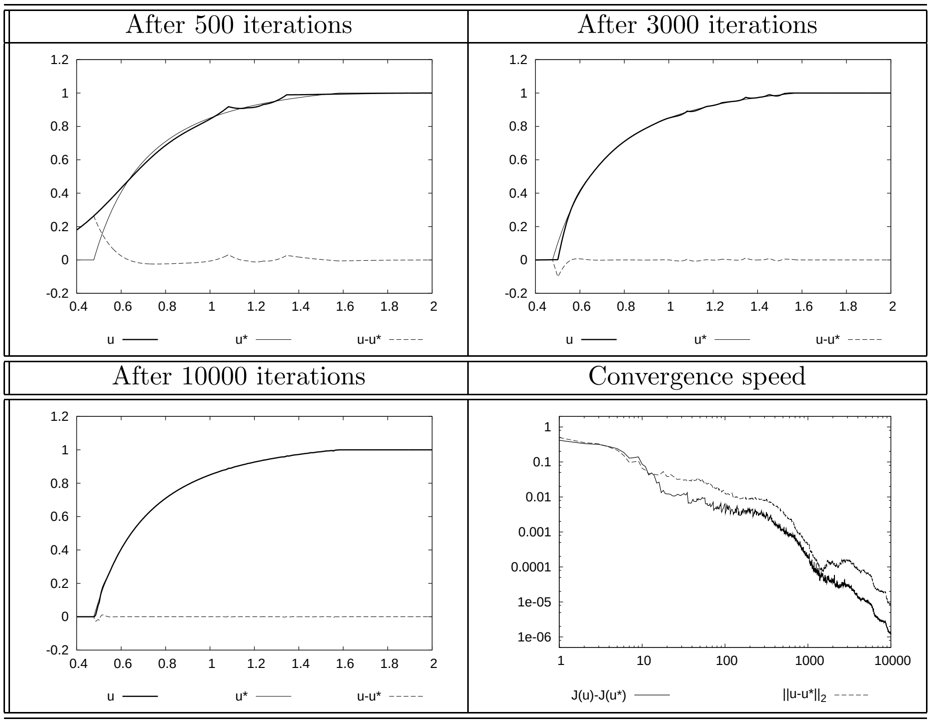
The optimal control is therefore:

$$\forall \xi \in [\underline{x}, \overline{x}],\ u^*(\xi) = \left[s + \epsilon - \frac{1}{4\xi^2}\right]_0^s.$$

It can be rewritten as follows, for all $\xi \in [\underline{x}, \overline{x}]$:

$$u^*(\xi) = \begin{cases} 0 & \text{if } \xi < \frac{1}{2\sqrt{\epsilon+s}} \\ s + \epsilon - \frac{1}{4\xi^2} & \text{if } \frac{1}{2\sqrt{\epsilon+s}} \leq \xi \leq \frac{1}{2\sqrt{\epsilon}} \\ s & \text{if } \xi > \frac{1}{2\sqrt{\epsilon}} \end{cases}$$

For the numerical application of our algorithm, we take $s = 1, \epsilon = 0.1, [\underline{x}, \overline{x}] = [0.4, 2]$. Figure 4 represents the control obtained after 500, 3000 and 10000 iterations, the optimal one, and the error term for each possible value of the price.


After 500 iterations

After 3000 iterations

After 10000 iterations

Convergence speed

FIGURE 4. Reservoir Problem with one time period and one noise, feedback along the iterations and convergence speed

Once again, it is clear that our algorithm provides a good solution to this problem.

We can also consider s to be stochastic, denoted by $\boldsymbol{s}$, independent from $\boldsymbol{\xi}$. We assign $\boldsymbol{s}$ to follow a uniform law on $[0, 1]$. Hence, the cost function j becomes a function of the stock level $\boldsymbol{s}$, and we consider now the problem of minimizing the following criterion:

$$\forall u \in L^2([\underline{x}, \overline{x}] \times [0,1], \mathbb{R}),\ \mathcal{J}(u) = \mathbb{E}\left(j(u(\boldsymbol{\xi}, \boldsymbol{s}), \boldsymbol{\xi}, \boldsymbol{s})\right).$$

The theoretical computations lead to the same optimal control, which from now on is a function of ξ the price level and s the stock level. The optimal control is represented in Figure 5.

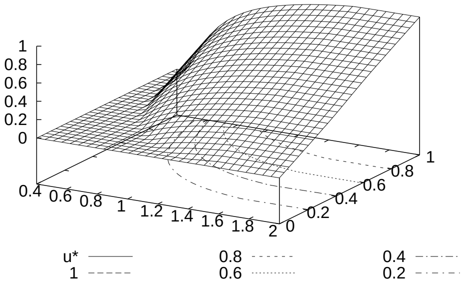


Figure 5. Reservoir Problem with one time period and two noises, Optimum

The application of our algorithm, with the same parameters as before yields Figure 6 showing at the current iteration the feedback (top) and the corresponding error term (bottom), along the iterations. Figure 7 shows the convergence speed for this problem.

3.3.2. *Two time periods.* We finally consider an even more practical problem, which is exactly the same as before, with two successive random prices, and two associated controls. The problem is more complicated in the sense that there is a measurability constraint on the first control: the first decision has to be taken prior to any knowledge of the second price, except its conditional law with respect to the first one. Mathematically, we consider the following cost function:

$$j(u_1, u_2, \xi_1, \xi_2) = -u_1\xi_1 - u_2\xi_2 - \sqrt{\epsilon + s - u_1 - u_2}, \tag{40}$$

for all $(\xi_1, \xi_2) \in [\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2]$, and for all $u_1 \in [0, s],\ u_2 \in [0, s - u_1]$. We take for $i = 1, 2$, $\boldsymbol{\xi}_i$ to be a real random variable with uniform law on $[\underline{x}_i, \overline{x}_i]$, such that $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$ are independent. Classically, the criterion to be minimized is given by:

$$\mathcal{J}(u_1, u_2) = \mathbb{E}\left(j(u_1(\boldsymbol{\xi}_1), u_2(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2), \boldsymbol{\xi}_1, \boldsymbol{\xi}_2)\right),$$

with $u_1 \in L^2([\underline{x}_1, \overline{x}_1], \mathbb{R})$ and $u_2 \in L^2(\Pi_{i=1,2}[\underline{x}_i, \overline{x}_i], \mathbb{R})$. This way of stating the problem expresses itself the measurability conditions on the sequential controls u_1 and u_2.

We now come to the theoretical solution of this problem. We solve it recursively, using a classical dynamic programming procedure. We first compute the second optimal feedback u_2^*, as a function of the two first prices ξ_1 and ξ_2 and of the first feedback u_1. It is exactly the same calculation as before, and it yields:

$$u_2^*(\xi_1, \xi_2, u_1) = \left[\epsilon + s - u_1 - \frac{1}{4(\xi_2)^2}\right]_0^{s-u_1},$$

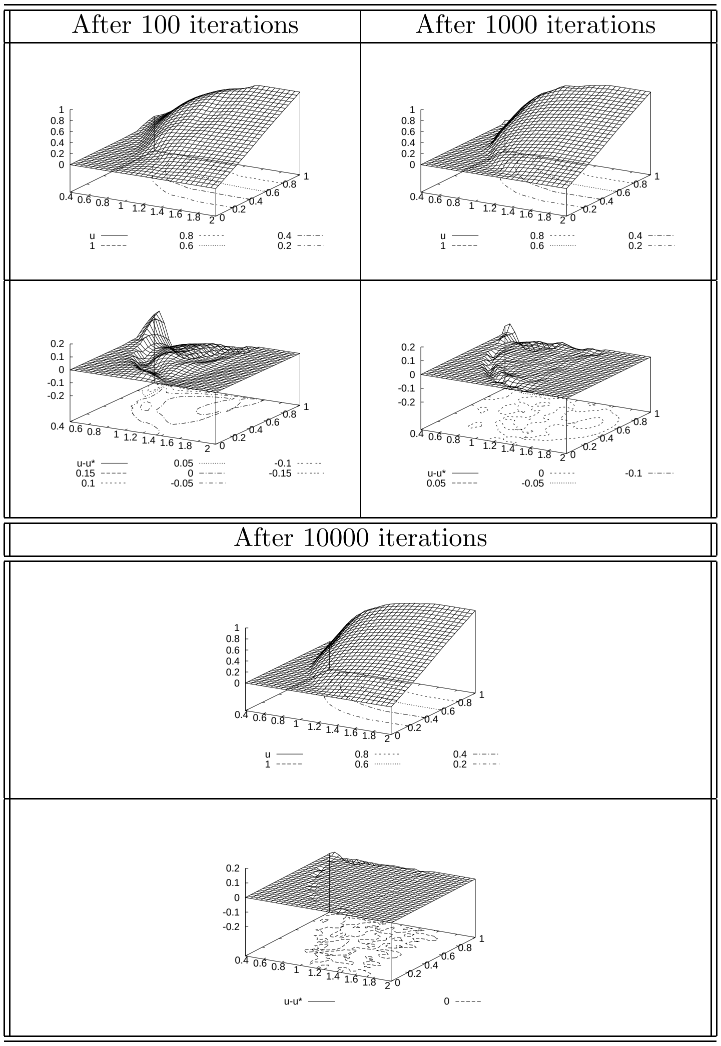


Figure 6. Reservoir Problem with one time period and two noises, feedback (top) and error (bottom) along the iterations

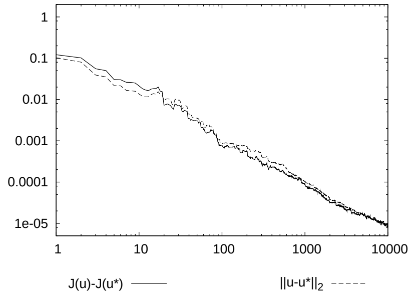


Figure 7. Reservoir Problem with one time period and two noises, convergence speed

which does not depend directly on ξ_1. More explicitly:

$$u_2^*(\xi_2, u_1) = \begin{cases} s - u_1 & \text{if } \xi_2 > \frac{1}{2\sqrt{\epsilon}}, \\ \epsilon + s - u_1 - \frac{1}{4(\xi_2)^2} & \text{if } \frac{1}{2\sqrt{\epsilon+s-u_1}} \leq \xi_2 \leq \frac{1}{2\sqrt{\epsilon}}, \\ 0 & \text{if } \xi_2 < \frac{1}{2\sqrt{\epsilon+s-u_1}} \end{cases}$$

We can moreover compute the gradient of u_2^* with respect to u_1:

$$\nabla_{u_1} u_2^*(u_1, \xi_2) = \begin{cases} -1 & \text{if } \xi_2 \geq \frac{1}{2\sqrt{\epsilon+s-u_1}} \\ 0 & \text{else} \end{cases}$$

We have to solve the following problem for all ξ_1, by independence:

$$\min_{u_1 \in [0,s]} -u_1\xi_1 - \mathbb{E}\left(u_2^*(\boldsymbol{\xi}_2, u_1)\boldsymbol{\xi}_2 + \sqrt{\epsilon + s - u_1 - u_2^*(\boldsymbol{\xi}_2, u_1)} \right)$$

We hence compute the gradient of this new cost function with respect to u_1 and equalize it to zero:

$$-\xi_1 + \mathbb{E}\left(\boldsymbol{\xi}_2 1_{[\frac{1}{2\sqrt{\epsilon+s-u_1}}, \overline{x}_2]}(\boldsymbol{\xi}_2) \right) - \mathbb{E}\left(\frac{1}{2\sqrt{s + \epsilon - u_1 - u_2^*(\boldsymbol{\xi}_2, u_1)}} 1_{[\frac{1}{2\sqrt{\epsilon+s-u_1}}, \overline{x}_2]}(\boldsymbol{\xi}_2) \right)$$
$$+ \mathbb{E}\left(\frac{1}{2\sqrt{s + \epsilon - u_1 - u_2^*(\boldsymbol{\xi}_2, u_1)}} \right) = 0,$$

We assume that $\overline{x}_2 < \frac{1}{2\sqrt{\epsilon}}$. We use now the explicit expression of u_2^*. Hence, the last inequality reads

$$\mathbb{E}\left(\frac{1}{2\sqrt{s + \epsilon - u_1 - u_2^*(\boldsymbol{\xi}_2, u_1)}} \right) = \xi_1.$$

We now compute this expectation ($\boldsymbol{\xi}_2$ follows the uniform law on $[\underline{x}_2, \overline{x}_2]$):

$$\frac{1}{\overline{x}_2 - \underline{x}_2} \int_{\underline{x}_2}^{\overline{x}_2} \frac{d\xi_2}{2\sqrt{s + \epsilon - u_1 - u_2^*(\boldsymbol{\xi}_2, u_1)}} = \xi_1, \text{ i.e.,}$$
$$\int_{\underline{x}_2}^{\frac{1}{2\sqrt{s+\epsilon-u_1}}} \frac{1}{2\sqrt{s + \epsilon - u_1}} d\xi_2 + \int_{\frac{1}{2\sqrt{s+\epsilon-u_1}}}^{\overline{x}_2} \xi_2 d\xi_2 = (\overline{x}_2 - \underline{x}_2)\xi_1,$$

For the simplicity of the computations, we define $r = \frac{1}{2\sqrt{\epsilon+s-u_1}}$. We can then continue our calculus:

$$r(r - \underline{x}_2) + \frac{(\overline{x}_2)^2}{2} - \frac{r^2}{2} = (\overline{x}_2 - \underline{x}_2)\xi_1,$$
$$(r - \underline{x}_2)^2 = \left((\underline{x}_2)^2 - (\overline{x}_2)^2\right) + 2\xi_1(\overline{x}_2 - \underline{x}_2),$$
$$r = \underline{x}_2 + \sqrt{2(\overline{x}_2 - \underline{x}_2)(\xi_1 - \frac{\overline{x}_2 + \underline{x}_2}{2})_+}.$$

We can express the optimal control $u_1^*(\xi_1)$:

$$u_1^*(\xi_1) = \left[\epsilon + s - \frac{1}{4\left(\underline{x}_2 + \sqrt{2(\overline{x}_2 - \underline{x}_2)(\xi_1 - \frac{\overline{x}_2 + \underline{x}_2}{2})_+} \right)^2} \right]_0^s.$$

The optimal control u_2^{**} is then given by $u_2^{**}(\xi_1, \xi_2) = u_2^*(\xi_2, u_1^*(\xi_1))$.

We now give few numerical results, with $s = 1, \epsilon = 0.1, \underline{x}_1 = \underline{x}_2 = 0.4, \overline{x}_1 = \overline{x}_2 = 2$. Figure 8 shows u_2^{**} obtained with these values, as a function of the two noises $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2$.

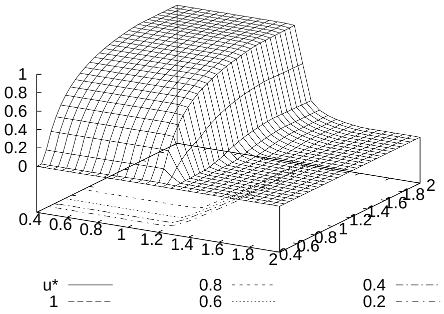


Figure 8. Reservoir Problem with two time periods, Optimal feedback at the second time step

We then apply our algorithm to the solution of this problem, and it yields the graphs given in Figure 9, giving the evolution of u_1 (top), u_2 (middle) and the error on u_2 (bottom) after respectively 1000, 10000, and 100000 iterations. In the previous examples, the simple bound constraints made the projection very easy to perform. On the contrary, in this last example, performing the projection on the subset defined by the constraint $u_2 \leq s - u_1$ is quite difficult, requiring the calculation of an expectation which can only be performed numerically. We overcome this difficulty by solving the equivalent penalized problem where u_2 is only constrained to be in $[0, s]$, and $j\left(u_1, u_2, \xi_1, \xi_2\right) = a_1 u_1 + a_2 u_2$ for all $u_2 \geq s - u_1$, with a_1 and a_2 being positive penalization constants appropriately chosen. The algorithm hence converges correctly, but slowly.

Remark 3.2 (Computational time)**.** The proposed algorithm (2.1) requires one gradient evaluation per iteration, which in turn requires one evaluation of the control, i.e., a possibly projected sum of kernels. As the number of terms of this sum grows over the iterations, this summation may represent the largest part of the computation time. Nevertheless, since the kernel values tend to decrease quickly toward zero away from their center, the use of kd-trees for spatial indexing of the draws $\boldsymbol{\xi}^k$ may greatly improve the performance, resulting in logarithmic time evaluation in the non-measurability constrained case, and sub-linear growth in the other case.

Remark 3.3 (Heuristics for the stepsizes)**.** It is worth noting that stochastic algorithms are very sensitive to the choice of the stepsizes. We can propose a heuristic to fit the steps (ϵ^k) and (ρ^k) in our algorithms. The problem here is to fit the stepsizes ρ^k, ϵ^k for all k with the current draw $\boldsymbol{\xi}^{k+1}$. Our idea is the following. When you draw $\boldsymbol{\xi}^{k+1}$, you will move your control around $\boldsymbol{\xi}^{k+1}$, in a neighbourhood defined by ϵ^k, and with a depth ρ^k. The next time you will fall in this neighbourhood, you may want to have a new neighbourhood and a new depth almost as large as the preceding time, since the draws between thoses two steps did not contribute to the control in this neighbourhood. We hence propose an adaptive way to fit the stepsizes to the draw. Let us define iteratively for all $k \in \mathbb{N}$, the mappings $f^k : \mathbb{R}^m \to \mathbb{R}$ by:

$$f^k(\cdot) = \sum_{l=0}^{k-1} \frac{1}{\epsilon^l} K^l(\boldsymbol{\xi}^{l+1}, \cdot).$$

Hence, for all $k \in \mathbb{N}$, f^k is $\mathcal{F}^k$-measurable, and f^k/k can be considered as an approximation of the density function of the law μ of $\boldsymbol{\xi}$. Let us now define for

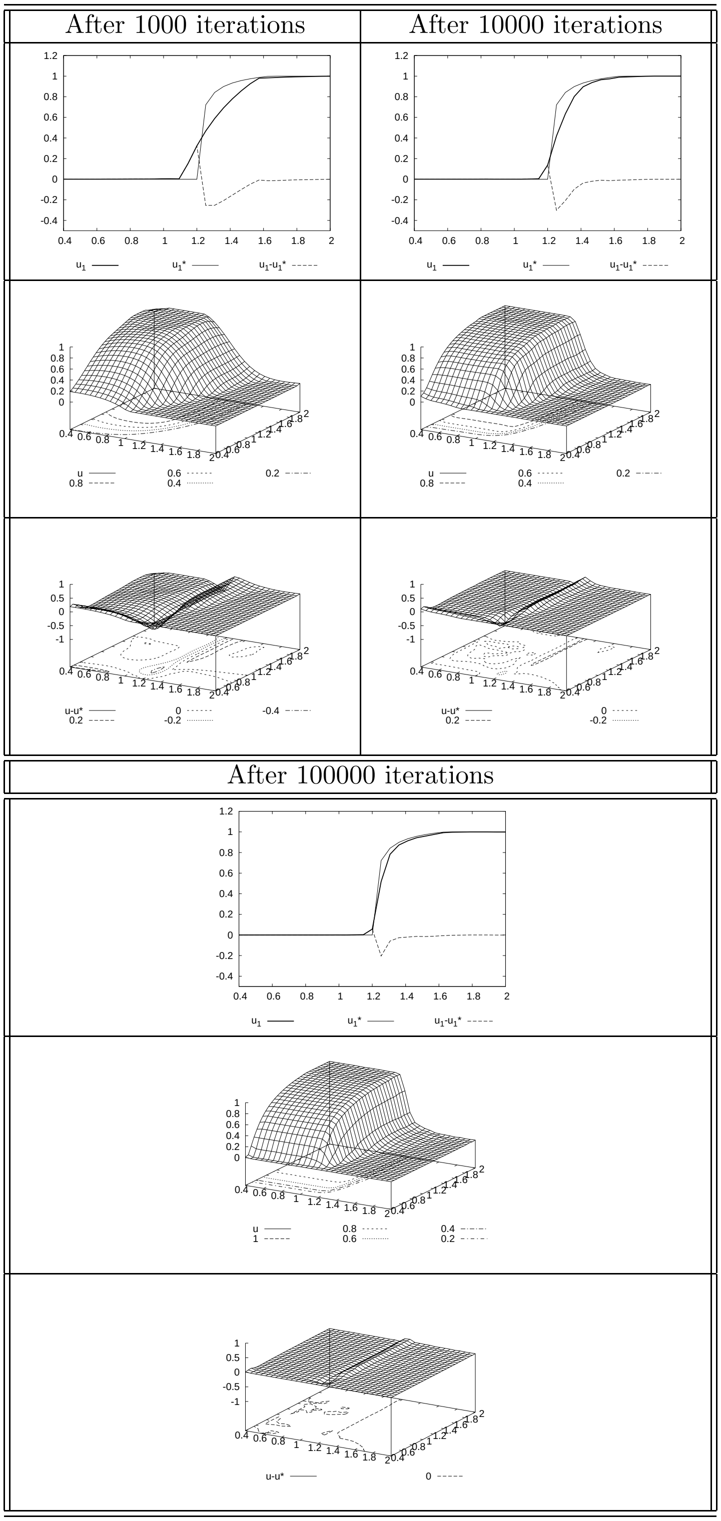


Figure 9. Reservoir Problem with two time periods, feedback at the first time step (top), at the second time step (middle), and error of the second feedback (bottom), along the iterations

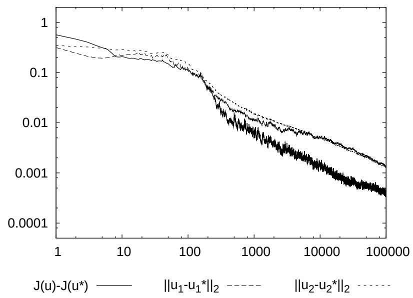


Figure 10. Reservoir Problem with two time periods, convergence speed

all $k \in \mathbb{N}$, $g^k = \lfloor f^k(\boldsymbol{\xi}^{k+1}) \rfloor$. It means that the larger is the g^k, the more the neighbourhood of $\boldsymbol{\xi}^{k+1}$ has been explored in the past steps. Then, one can choose the next stepsizes ρ^k and ϵ^k. Practically, one chooses two nonnegative sequences (η_ρ^k) and (η_ϵ^k) which satisfy assumption (16), and one defines iteratively with $k \in \mathbb{N}$:

$$\epsilon^k = \eta_\epsilon^{g^k}, \quad \text{and,} \quad \rho^k = \eta_\rho^{g^k}.$$

In many cases, (ϵ^k) and (ρ^k) will satisfy assumption (16), but for all $k \in \mathbb{N}$, ϵ^k and ρ^k are $\mathcal{F}^{k+1}$-measurable, i.e., in one sense, anticipative, and we hence fail to prove that this heuristic leads to the convergence of Algorithm (2.1).

To sum up, our idea is to ensure that the stepsizes decrease according to the frequency each neighbourhood has been explored. Therefore, rarely explored regions should have a slower decreasing speed for the corresponding window sizes and depths, and vice versa for frequently visited regions.

4. Conclusion

We propose in this paper a new stochastic gradient type algorithm to solve closed-loop stochastic optimization problems, and provide two convergence proofs for general case with projection on a closed convex subset.

We then give few examples showing that this algorithm is tractable even for multi-stage problems (our example is a three stage program without first open-loop decision). For this kind of application, the projection operations we must do at each iteration can be a computational hurdle.

Our approach can be compared with the approach consisting in a parametrization of the feedback which is searched as a linear combination of given functions (the basis). In our case, when we stop the algorithm, we obtain the feedback as a linear combination of the successive kernels, and we know that it is not the optimal one with respect to this particular sub space, but that it is optimal in an other sense, with respect to the initial functional space (Note that another difference between the two point of view comes from the projections we are doing).

From a theoretical point of view, the need for the gradient steps to be decreasing is not yet completely clear, and we provide a proof (see Theorem 2.2) with constant gradient steps in a particular case. Our work in the future will thus also be continued in this direction.

Another idea to improve the convergence speed is to use when it is possible, average ideas or optimally fitted steps. This way has not yet been developed.

We think also that our algorithm can now be extended to the case of Temporal Difference Learning for Stochastic Dynamic Programming problems, developped by [Bertsekas and Tsitsiklis, 1996]. It will be our main axis in the future.

References

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

[Bertsekas and Tsitsiklis, 2000] Bertsekas, D. and Tsitsiklis, J. (2000). Gradient convergence in gradient methods. *SIAM J. Optim.*, 10(3):627–642.

[Borkar, 1998] Borkar, V. (1998). Asynchronous stochastic approximations. *SIAM J. Control Optimization*, 36(3):840–851.

[Cohen, 1984] Cohen, G. (1984). *Décomposition et Coordination en optimisation déterministe différentiable et non-différentiable*. Thèse de doctorat d'État, Université de Paris IX Dauphine.

[Cohen and Culioli, 1990] Cohen, G. and Culioli, J.-C. (1990). Decomposition Coordination Algorithms for Stochastic Optimization. *SIAM J. Control Optimization*, 28(6):1372–1403.

[Delyon, 1996] Delyon, B. (1996). General results on the convergence of stochastic algorithms. *IEEE Trans. Autom. Control*, 41(9):1245–1255.

[Granichin, 2002] Granichin, O. (2002). Randomized Algorithms for Stochastic Approximation under Arbitrary Disturbances. *Autom. Remote Control*, 63(2):209–219.

[Higle and Sen, 1996] Higle, J. and Sen, S. (1996). *Stochastic Decomposition*. Kluwer, Dordrecht.

[Holt et al., 1955] Holt, C., Modigliani, F., and Simon, H. (1955). A linear decision rule for production and employment scheduling. *Manage. Sci.*, 2(1).

[Lai, 2003] Lai, T. (2003). Stochastic Approximation. *Ann. Stat.*, 31(2):391–406.

[Métivier, 1982] Métivier, M. (1982). *Semimartingales*. De Gruyter, Berlin.

[Robbins and Monro, 1951] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407.

[Robbins and Siegmund, 1971] Robbins, H. and Siegmund, D. (1971). A convergence theorem for nonnegative almost supermartingales and some applications. In Rustagi, J., editor, *Optimizing Methods in Statistics*, pages 233–257. Academic Press, New York.

[Shapiro and Ruszczynski, 2003] Shapiro, A. and Ruszczynski, A. (2003). *Stochastic Programming*. Elsevier, Amsterdam.