# A Stochastic Algorithm for M-estimator Computation

Nabil Rachdi, Jean-Claude Fort

# A Stochastic Algorithm for M-estimator Computation

Nabil Rachdi*, Jean-Claude Fort†

October 2, 2018

**Abstract**

Most of statistical procedures consist in estimating parameters by minimizing (or maximizing) some criterion, a minimizing parameter is also called in the statistical literature *M-estimator*, [4]. So to compute an M-estimator consists in finding a global minimum. Depending on the statistical problem and the available information, the criterion to minimize may be more or less complicated: non convex, no gradient, non smooth etc... Moreover, generally only evaluations of the criterion are reachable. Thus, it can be difficult in practice to compute a *M-estimator*. We propose a new procedure to compute a global minimum, using a stochastic algorithm to take advantage of various smooth versions of the criterion.

**Keywords:** M-estimation, optimization, stochastic algorithms, smoothing methods

## 1 Introduction

Let $H : \Theta \longrightarrow \mathbb{R}$ be some mapping, or criterion function, where $\Theta = \mathbb{R}^k$. Moreover, we assume that $H$ has a unique global minimum $\boldsymbol{\theta}$ over $\Theta$ noted $\widehat{\boldsymbol{\theta}}$. Hence the problem is to compute

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\Theta}{\text{Argmin}}\, H(\boldsymbol{\theta})\,. \tag{1}$$

A classical example is the *maximum likelihood estimator* where $H$ takes the form

$$H(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log(p_{\boldsymbol{\theta}}(Y_i))\,,$$

where $Y_1, ..., Y_n$ are i.i.d random variables drawn from a distribution $Q$ and $\{p_{\boldsymbol{\theta}},\, \boldsymbol{\theta} \in \Theta\}$ is some family of density functions. In the case of a gaussian family of mean $\mu = \boldsymbol{\theta}$ and variance $\sigma^2 = 1$, the function $H$ becomes (see Figure 1(a))

$$H(\boldsymbol{\theta}) = \sum_{i=1}^{n} (Y_i - \boldsymbol{\theta})^2 + C\,,$$

where $C$ is a constant independent of $\boldsymbol{\theta}$. Here, one computes

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\Theta}{\text{Argmin}} \sum_{i=1}^{n} (Y_i - \boldsymbol{\theta})^2 = \frac{1}{n}\sum_{i=1}^{n} Y_i\,.$$

There are also cases where a simple Newton algorithm is enough to compute $\widehat{\boldsymbol{\theta}}$. However, the function $H$ can be complicated, for instance if it is the result of a "complex" statistical modeling. Indeed, let us consider the estimators resulting in the statistical procedures introduced in the work of N. Rachdi et al. [7] :

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\Theta}{\text{Argmin}} \sum_{i=1}^{n} \Psi\left(\sum_{j=1}^{m} \widetilde{\rho}_{\mathcal{F}}(h(\mathbf{X}_j, \boldsymbol{\theta}))\,,\, Y_i\right) \tag{2}$$

*ACTUARIS, 13/15 Boulevard de la Madeleine, 75001 Paris, nabil.rachdi@actuaris.com

†Université Paris Descartes, 45 rue des saints pères, 75006 Paris, jean-claude.fort@parisdescartes.fr

where $Y_1, ..., Y_n$ are i.i.d random variables drawn from a distribution $Q$, $\mathbf{X}_1, ..., \mathbf{X}_m$ are i.i.d random variables drawn from a distribution $P^{\mathbf{x}}$, $\mathcal{F}$ is some feature space, $\Psi : \mathcal{F} \to L_1(Q)$ is a $\mathcal{F}$-contrast, $\widetilde{\rho}_{\mathcal{F}} : \mathcal{Y} \to \mathcal{F}$ is some weight function and $h$ is a computer code. Here, the function $H$ is

$$H(\boldsymbol{\theta}) = \sum_{i=1}^{n} \Psi \left( \sum_{j=1}^{m} \widetilde{\rho}_{\mathcal{F}}(h(\mathbf{X}_j, \boldsymbol{\theta})) \, , \, Y_i \right) .$$

For instance, let us consider the case of the log-contrast $\Psi(\rho, y) := -\log(\rho(y))$ and the weight function $\widetilde{\rho}_{\mathcal{F}}(y)(\cdot) := \frac{1}{\sqrt{2\pi}\, b} \, e^{(\frac{\cdot - y}{b})^2}$ with a bandwidth $b$. The bandwidth $b$, in fact $b_{\boldsymbol{\theta}}$, is computed from the sample $h(\mathbf{X}_j, \boldsymbol{\theta})$, $j = 1, ..., m$ for $\boldsymbol{\theta} \in \Theta$, by Silverman's rule-of-thumb :

$$b_{\boldsymbol{\theta}} = 1.06 \, m^{-1/5} \, \hat{\sigma}_{\boldsymbol{\theta}} \, ,$$

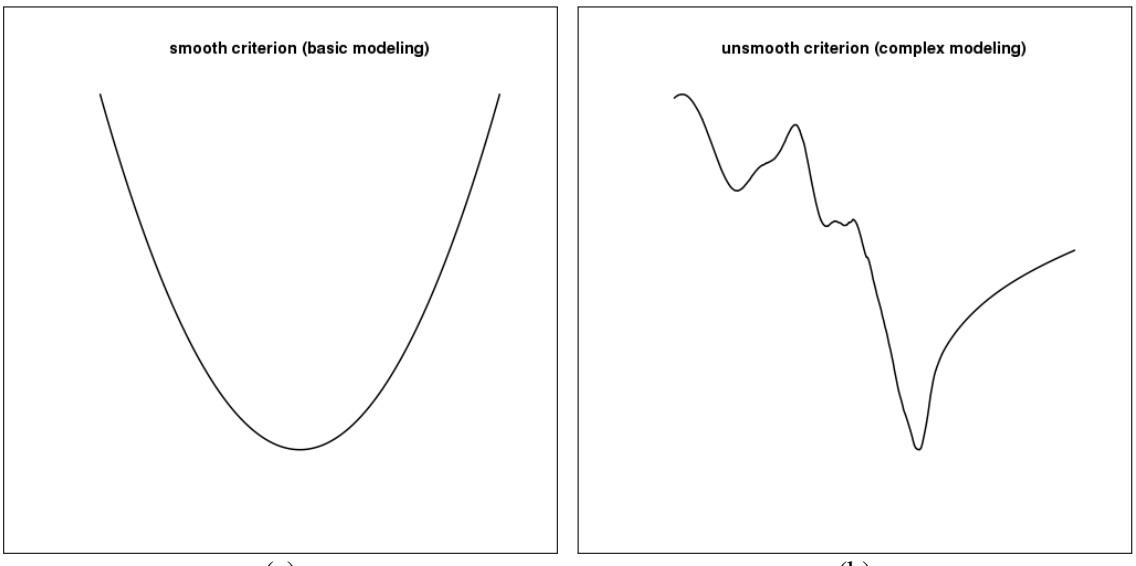
where $\hat{\sigma}_{\boldsymbol{\theta}}$ is the empirical standard deviation of the sample $h(\mathbf{X}_j, \boldsymbol{\theta})$, $j = 1, ..., m$.
Finally $H$ becomes

$$H(\boldsymbol{\theta}) = -\sum_{i=1}^{n} \log \left( \sum_{j=1}^{m} e^{(h(\mathbf{X}_j, \boldsymbol{\theta}) - Y_i)^2 / b_{\boldsymbol{\theta}}^2} \right) + C, \tag{3}$$

where $C$ is a constant independent of $\boldsymbol{\theta}$ .
Let us recall that $h$ is a computer code, viewed as a *black-box*[1], which represents a physical phenomenon. Typically, $h$ gives solutions of differential equations etc... The computation of $\widehat{\boldsymbol{\theta}}$ is given by

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \Theta}{\text{Argmin}} \; -\sum_{i=1}^{n} \log \left( \sum_{j=1}^{m} e^{(h(\mathbf{X}_j, \boldsymbol{\theta}) - Y_i)^2 / b_{\boldsymbol{\theta}}^2} \right) . \tag{4}$$

We should take into account two important issues. First, the function can be highly non-convex (with many local minima), second we don't have the analytical expression of the gradient. Figure 1(b) shows an example in dimension one of function $H$ resulting of this modeling.




(a) (b)

Figure 1: (a) Function $H$ for the mean of a gaussian with known variance. (b) Function $H$ for a one dimensional complex statistical modeling.

[1]black-box: function known only through its input and output values

## 2 Smoothness and stochastic algorithm

In this section, we attempt to overcome irregularities (many critical points) or non-smoothness of the function $H$ by making a convolution with some appropriate function, and we will see how naturally appears a stochastic algorithm.
The smoothness method is taken from [6] where the main idea is to minimize a modified function smoother than $H$ while controlling the degree of smoothness, instead of minimizing directly $H$. However, when the modified function is a convolution of the criterion $H$ with some function, one has to compute multi-dimensional integrals. This limits, in general, the use of such method in high dimensions. Moreover, in many applications, $H$ is not analytically known, but only computable, which makes the computation of a smooth version of $H$ intractable. In this paper we propose optimization procedures based on stochastic algorithms to compute the minimizer, which in any case overcomes the computation of multi-dimensional integrals to get a smooth version of $H$.

Let $g_{\sigma^2}$, $\sigma > 0$, be the normal probability density function ($p.d.f.$) with variance $\sigma^2$.
Let us denote by $H_\sigma$ the convolution of $H$ and $g_{\sigma^2}$

$$H_\sigma(\boldsymbol{\theta}) := \int_\Theta H(\boldsymbol{\theta} - w)\, g_{\sigma^2}(w) dw, \quad \sigma > 0\,. \tag{5}$$

By noting that $g_\sigma(\cdot) = \frac{1}{\sigma^k} g(\frac{\cdot}{\sigma})$, where $g$ is the standard multivariate $p.d.f.$, we show the following basic lemma.

**Lemma 2.1.** *Let $\mathcal{C}(\Theta)$ be the space of all continuous functions on $\Theta$.*
*If $H \in \mathcal{C}(\Theta)$, then*

$$\forall\, \boldsymbol{\theta} \in \Theta\,, \quad H_\sigma(\boldsymbol{\theta}) \underset{\sigma \to 0}{\longrightarrow} H(\boldsymbol{\theta})\,.$$

*If $H$ is integrable, then*

$$\forall\, \boldsymbol{\theta} \in \Theta\,, \quad H_\sigma(\boldsymbol{\theta}) \underset{\sigma \to +\infty}{\longrightarrow} 0\,.$$

The previous lemma shows the smoothing control of the function $H$ by the transformation (5).

**Remark 2.1.** *The convolution with the $p.d.f$ $g_\sigma$ can be generalized to any other $p.d.f.$ $f$ on $\mathbb{R}^k$ by defining*

$$f_\sigma(\cdot) = \frac{1}{\sigma^k} f\left(\frac{\cdot}{\sigma}\right)\,.$$

Let us consider the following function as an academic example

$$H(\boldsymbol{\theta}) = \boldsymbol{\theta}^2 + a \sin(b\, \boldsymbol{\theta})\,, \tag{6}$$

with $a > 0$ and $b > 0$. It is easy to show that

$$H_\sigma(\boldsymbol{\theta}) = \boldsymbol{\theta}^2 + a \sin(b\, \boldsymbol{\theta})\, e^{-(b\,\sigma)^2/2} + \sigma^2\,. \tag{7}$$

**Remark 2.2.** *Notice that we have $H_\sigma(\boldsymbol{\theta}) \underset{\sigma \to 0}{\longrightarrow} H(\boldsymbol{\theta})$ but not $H_\sigma(\boldsymbol{\theta}) \underset{\sigma \to +\infty}{\longrightarrow} 0$ since for large $\sigma > 0$, $H(\boldsymbol{\theta}) \approx \boldsymbol{\theta}^2 + \sigma^2$, which is a very smooth function.*

Figure 2 shows the behavior of the function $H_\sigma$ (with $a = 1$ and $b = 6$) with respect to the parameter $\sigma$.
Now the challenge is to compute the minimizer of $H_\sigma$ which would be, a priori, more tractable than the minimizer of $H$. However, $H_\sigma(\boldsymbol{\theta})$ requires the knowledge of $H$ which is supposed to be unknown analytically. Also, the computation of $H_\sigma(\boldsymbol{\theta})$ needs to integrate on $\Theta \in \mathbb{R}^k$ which can be difficult in general, especially if $k$ is large. The following remark is the key of this work.
Notice that

$$H_\sigma(\boldsymbol{\theta}) = \int_\Theta H(\boldsymbol{\theta} - w)\, g_{\sigma^2}(w) dw = \mathbb{E}_{W^\sigma \sim g_{\sigma^2}} \left(H(\boldsymbol{\theta} - W^\sigma)\right)\,, \tag{8}$$

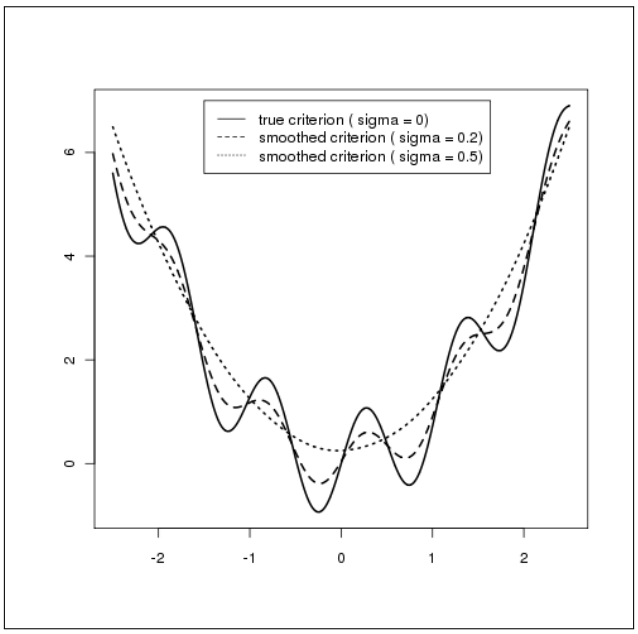



Figure 2: Illustration of the transformation (7) for different values of $\sigma$, with $a = 1$ and $b = 6$.

where $W^\sigma \sim g_{\sigma^2}$ means that $W^\sigma$ is a random variable with $p.d.f.\ g_{\sigma^2}$. To use classical notation from the stochastic algorithms literature, we will write

$$H(\boldsymbol{\theta} - w) = H(\boldsymbol{\theta}, w)\,.$$

By the display (8), i.e $H_\sigma(\boldsymbol{\theta}) = \mathbb{E}_{W^\sigma \sim g_{\sigma^2}}\left(H(\boldsymbol{\theta}, W^\sigma)\right)$, we propose to use stochastic algorithms to compute the minimizer of $H_\sigma$. Given a sequence of random variables $(W_t^\sigma)_{t\geq 1}$ i.i.d from the distribution $g_{\sigma^2}$, and sequences of real numbers $(\gamma_t)_{t\geq 0}$, $(\delta_t)_{t\geq 0}$ decreasing to zero (both may depend on $\sigma$), we form the following Kiefer-Wolfowitz algorithms ( [1] p. 53 ) for each $\sigma > 0$:

$$(KW)\begin{cases} \boldsymbol{\theta}_0^\sigma \in \Theta \\ \left(\widehat{\nabla}_{t+1} H(\boldsymbol{\theta}_t^\sigma)\right)_l = \dfrac{H(\boldsymbol{\theta}_t^\sigma + \delta_{t+1}\mathbf{e}^l, W_{t+1}^\sigma) - H(\boldsymbol{\theta}_t^\sigma - \delta_{t+1}\mathbf{e}^l, W_{t+1}^\sigma)}{2\,\delta_{t+1}} \\ \boldsymbol{\theta}_{t+1}^\sigma = \boldsymbol{\theta}_t^\sigma - \gamma_{t+1}\,\widehat{\nabla}_{t+1} H(\boldsymbol{\theta}_t^\sigma) \end{cases} \tag{9}$$

where $(\mathbf{e}^l)_{l=1,\ldots,k}$ is the canonical basis of $\mathbb{R}^k$. Let us notice that we use a single sequence $(W_t^\sigma)_{t\geq 1}$ and not a two independent sequences $(W_t^\sigma)_{t\geq 1}$ and $(\widetilde{W}_t^\sigma)_{t\geq 1}$ as the Kiefer-Wolfowitz algorithms are classically introduced. Of course a simple adaptation would produce a SPSA algorithm, but this is not a key point in this article. We recall the seminal result of Kiefer & Wolfowitz:

**Theorem 2.1** (Classical Kiefer-Wolfowitz theorem (see Proposition 1.4.28 in [2]))**.** *Let $f : \mathbb{R} \to \mathbb{R}$ be a function defined as $f(x) = \mathbb{E}(U(x))$ where $U(\cdot)$ is some random function. Let us define the iterative Kiefer-Wolfowitz procedure as follows*

$$x_{t+1} = x_t - \gamma_{t+1}\frac{U(x_t + \delta_{t+1}) - U(x_t - \delta_{t+1})}{\delta_{t+1}}\,,$$

*for some sequences $(\gamma_t)_t$ and $(\delta_t)_t$ decreasing to zero as $t \to +\infty$. If the three assumptions are satisfied*

- $\displaystyle\sum_{t\geq 0}\gamma_t = +\infty,\ \sum_{t\geq 0}\left(\frac{\gamma_t}{\delta_t}\right)^2 < +\infty\ (\mathcal{H}_1).$
- $\mathbb{E}(U^2(x)) \leq K(1 + x^2)$ *for some constant $K$*
- *$f$ has a unique global minimum noted $x^*$, is twice differentiable and strictly convex such that*

$$|f''(x)| \leq K(1 + |x|)\,,$$

*then* $x_t \underset{t\to+\infty}{\longrightarrow} x^*$ *almost surely* $(a.s)$.

**Theorem 2.2.** *Let us consider the (KW) algorithm in (9). Suppose that* $H_\sigma \in \mathcal{C}^2(\Theta)$ *and* $\nabla^2 H_\sigma$ *is Lipschitz and positive definite, and that the sequences* $\gamma = (\gamma_t)_{t\geq 0}$, $\delta = (\delta_t)_{t\geq 0}$ *decreasing to zero satisfy* $\mathcal{H}_1$.

*Then, for all* $\sigma > 0$

$$\boldsymbol{\theta}_t^\sigma \underset{t\to+\infty}{\longrightarrow} \widehat{\boldsymbol{\theta}^\sigma} := \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{Argmin}}\ H_\sigma(\boldsymbol{\theta}) \quad (a.s.) \ .$$

# 3 Stochastic Algorithm with smoothness.

We are not interested in finding the global minimizer of $H_\sigma$ but that of $H$. For this purpose, let us consider the previous (KW) algorithm (9) with the parameter $\sigma$ depending on $t$. Let us denote by $\sigma := (\sigma_t)_{t\geq 0}$, $\gamma := (\gamma_t)_{t\geq 0}$, and $\delta := (\delta_t)_{t\geq 0}$ three sequences of real numbers decreasing to zero. We propose the following algorithm.

**Algorithm 1** Stochastic Algorithm

**Require:** $\sigma : t \mapsto \sigma_t$, $\gamma$, $\delta$, $\boldsymbol{\theta}_0$, $\epsilon$, $T_{dyn}$

generate independent $W_1, ..., W_{T_{dyn}}$ with $W_t \sim g_{\sigma_t}$

**while** $\|\widehat{\nabla}_{t+1} H(\boldsymbol{\theta}_t)\| > \epsilon$ & $t \leq T_{dyn}$ **do**

$$\left(\widehat{\nabla}_{t+1} H(\boldsymbol{\theta}_t)\right)_l = \frac{H(\boldsymbol{\theta}_t + \delta_{t+1} e^l, W_{t+1}) - H(\boldsymbol{\theta}_t - \delta_{t+1} e^l, W_{t+1})}{2\,\delta_{t+1}}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \gamma_{t+1}\, \widehat{\nabla}_{t+1} H(\boldsymbol{\theta}_t)$$

**end while**

**return** $\boldsymbol{\theta}_t$

**Remark 3.1.** *This algorithm is "dynamic" in that the function* $H_{\sigma_t}$ *changes in time, converging toward the "true" function* $H$.

The function $\sigma : t \mapsto \sigma_t$ will be called *smoothing function* and we will see in the next section that its behavior is crucial for the convergence of our algorithm.

**Remark 3.2.** *The stochastic process* $\{\boldsymbol{\theta}_t, t \geq 0\}$ *provided by the Algorithm 1 is a Markov Chain.*

# 4 Simulated examples.

In this section, we test our algorithm on the 1D function (6) (with $a = 1$ and $b = 6$), on the 2D Rosenbrock function and on some particular multi-optima function. In all what follows we will consider the following parameters:

$$\epsilon = 10^{-5}, \quad T_{dyn} = 3500, \quad \Delta t = 5.10^{-2}.$$

## 4.1 1D example.

$$H(\boldsymbol{\theta}) = \theta^2 + \sin(6\,\boldsymbol{\theta})\,.$$

$H$ has a unique global minimum at $\boldsymbol{\theta} = -0.2424938$ .
In order to be in the practical conditions mentioned in the introduction, see (2) and (4), we suppose that we don't have at disposal the gradient of $H$, because $H$ is a black box, and that we cannot compute $H_\sigma$ (which is in fact given by (7)).

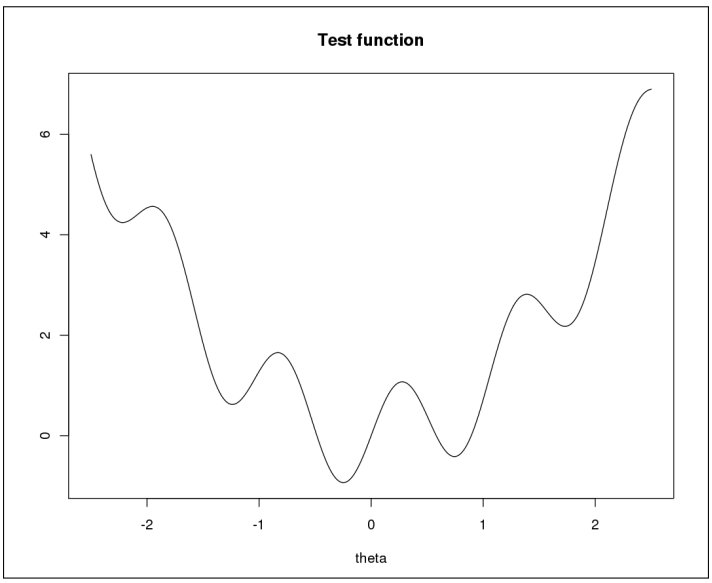



Figure 3: test function $H(\boldsymbol{\theta}) = \theta^2 + \sin(6\,\boldsymbol{\theta})$.

Let us consider the following sequences: $\gamma_t = \dfrac{10^{-1}}{t}$ and $\delta_t = \dfrac{10^{-1}}{t^{0.4}}$ (notice that these sequences satisfy conditions of Theorem 2.1).

We present the evolution of $\boldsymbol{\theta}_t$ versus $t$ at 10 different starting points $\boldsymbol{\theta}_0$, for three smoothing functions.

In Figure 4, we consider the trivial smoothing function $\sigma_t = 0$ for all $t \in [0, T_{dyn}]$, i.e there is no *dynamic* and $H_{\sigma=0} = H$ . It amounts to local methods (we see that $\boldsymbol{\theta}_t$ converges to the nearest minimum). In Figure 5 and Figure 6 we consider two others smoothing functions, where the first function decreases rapidly and the other one decreases slowly. It appears that for a suitable function $\sigma$ (which does not decrease too fast), the process $\{\boldsymbol{\theta}_t,\, t \in [0, T_{dyn}]\}$ converges (in some sense) to the minimum for all starting points (Figure 6 right).

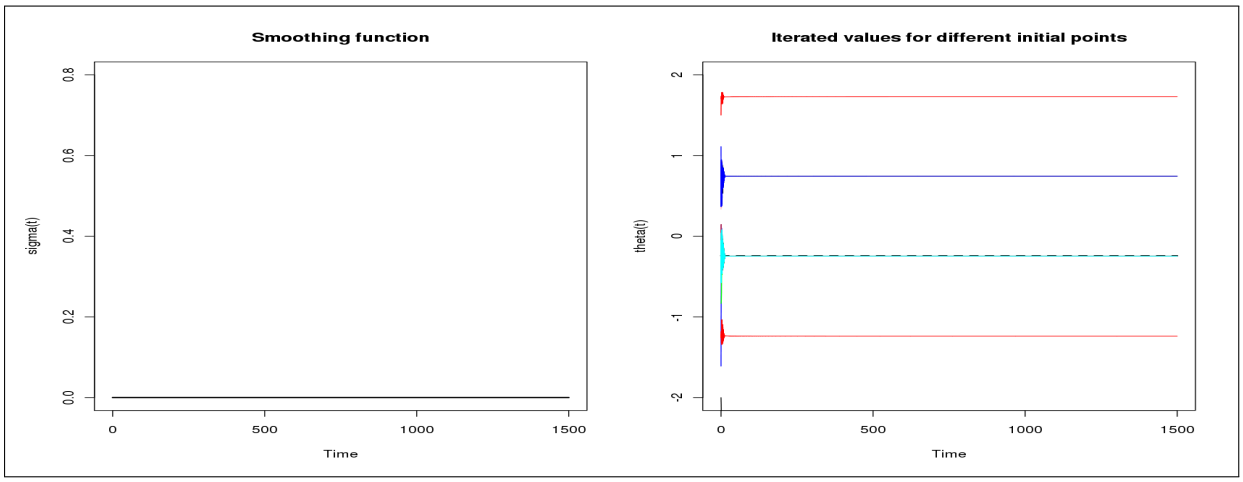



Figure 4: Convergence of the algorithm vs. behaviour of (decreasing rate of) the smoothness function $\sigma$, taken equal to 0. The graph on the right represents the convergence of the stochastic algorithm for 10 initial points.

## 4.2 2D examples.

**Rosenbrock function.**

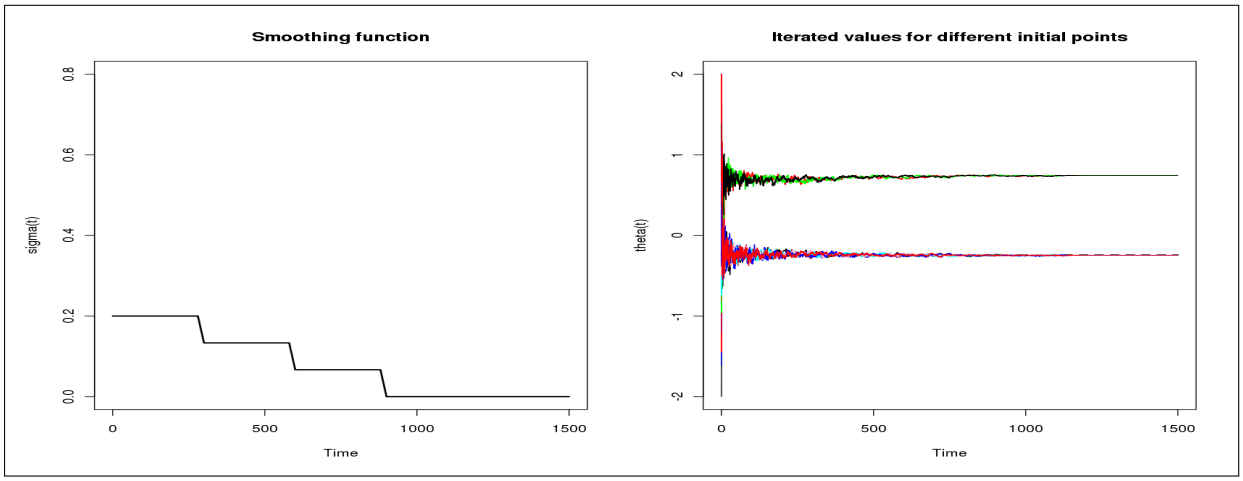



Figure 5: Convergence of the algorithm vs. behaviour of (decreasing rate of) the smoothness function $\sigma$, which decreases "rapidly". The graph on the right represents the convergence of the stochastic algorithm for 10 initial points.

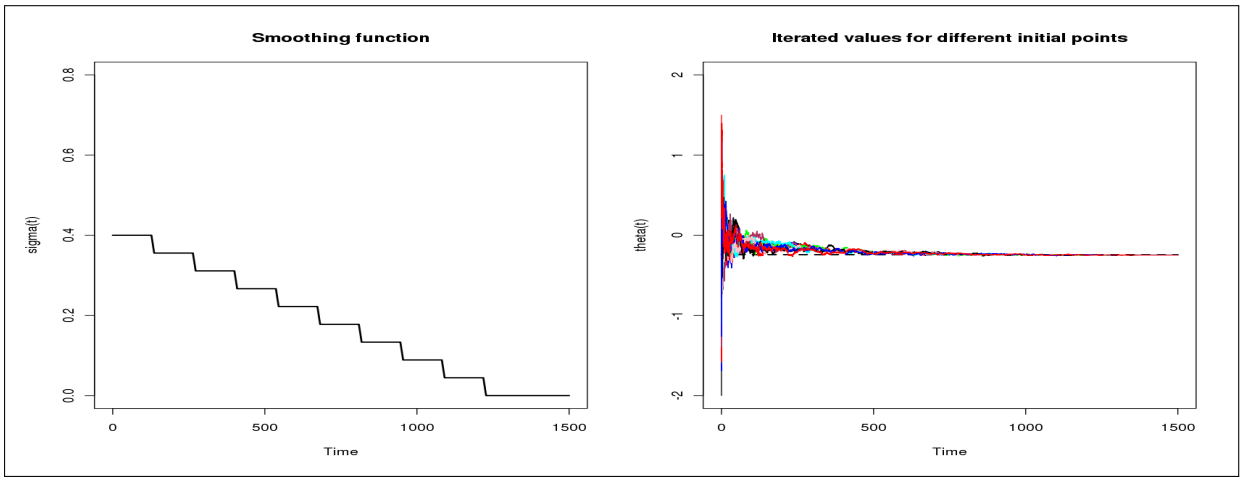



Figure 6: Convergence of the algorithm vs. behaviour of (decreasing rate of) the smoothness function $\sigma$, which decreases "slowly". The graph on the right represents the convergence of the stochastic algorithm for 10 initial points.

Now, let us consider the Rosenbrock function

$$H(\theta_1, \theta_2) = (\theta_1 - 1)^2 + 100\,(\theta_2 - \theta_1^2)^2\,.$$

$H$ has a unique global minimum at $(\theta_1, \theta_2) = (1, 1)$.

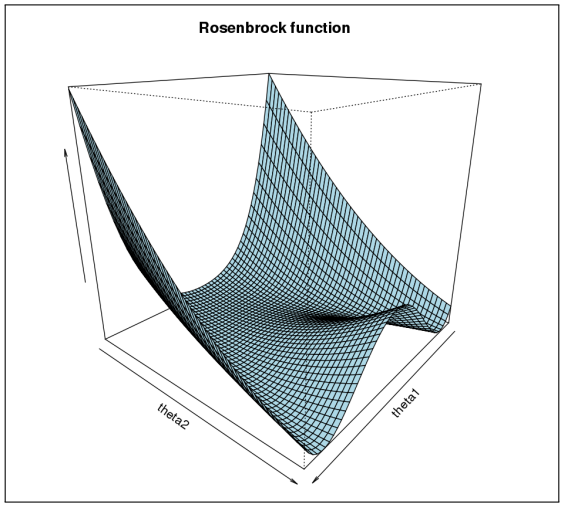
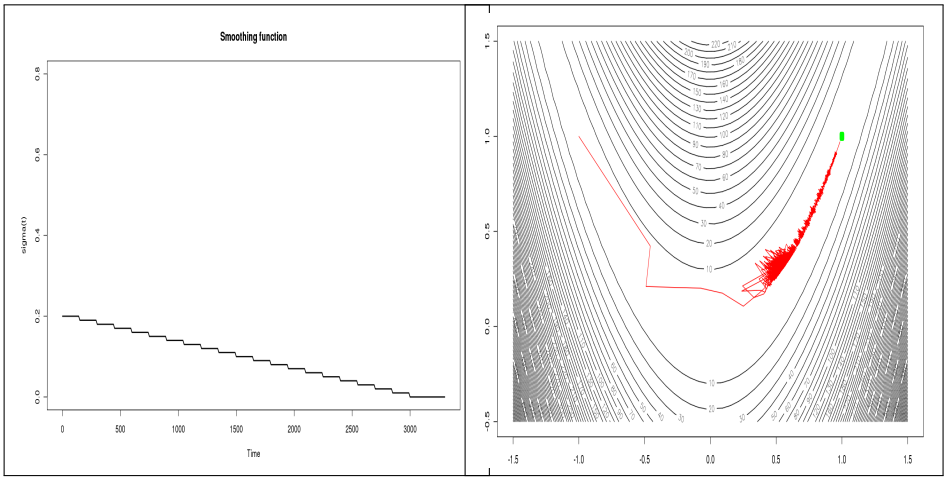



Figure 7: Rosenbrock function

We use the stochastic algorithm with the following sequences: $\gamma_t = \dfrac{1}{10^3 + t^{0.6}}$ and $\delta_t = \dfrac{10^{-2}}{t^{0.4}}$. The obtained minimum value is $(\theta_1, \theta_2)_{min} = (0.9856077, 0.9713646)$ and the Rosenbrock function evaluated at this point is $H_{min} = 2.07 \times 10^{-4}$.
Figure 8 shows the smoothing function used in the algorithm (8a) and the graph of convergence (8b).




(a) (b)

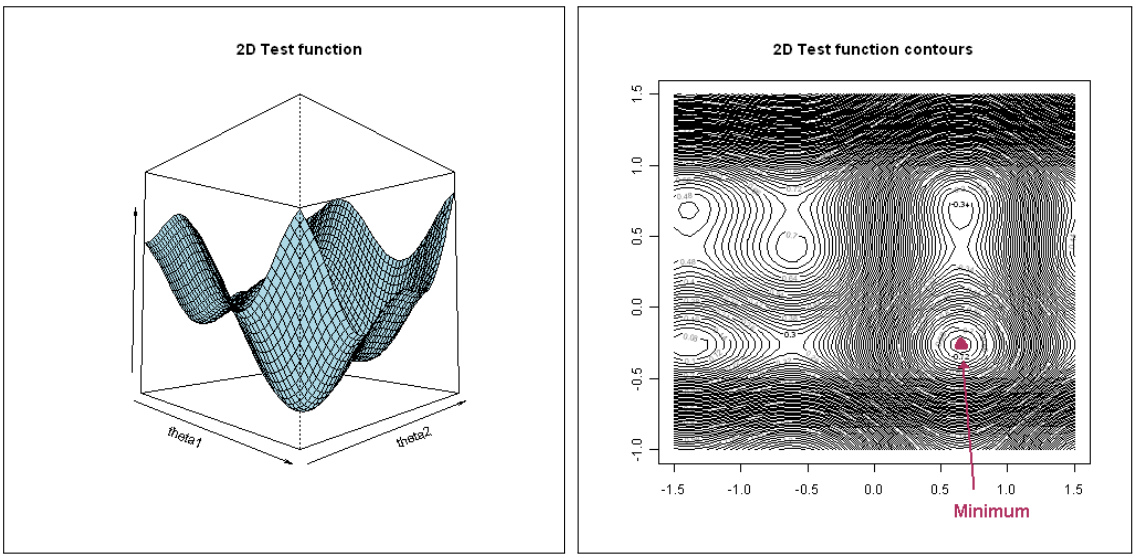
Figure 8: (a) Smoothing function. (b) stochastic algorithm applied to the Rosenbrock function.

**Multi-optima function.**

Let us consider the function

$$H(\theta_1, \theta_2) = \frac{\sin(3\theta_1)}{\theta_1 - 3} + \frac{\sin(5\theta_2)}{\theta_2 + 5} + 0.6\,\theta_2^2 - 0.3\,\theta_1\,,$$

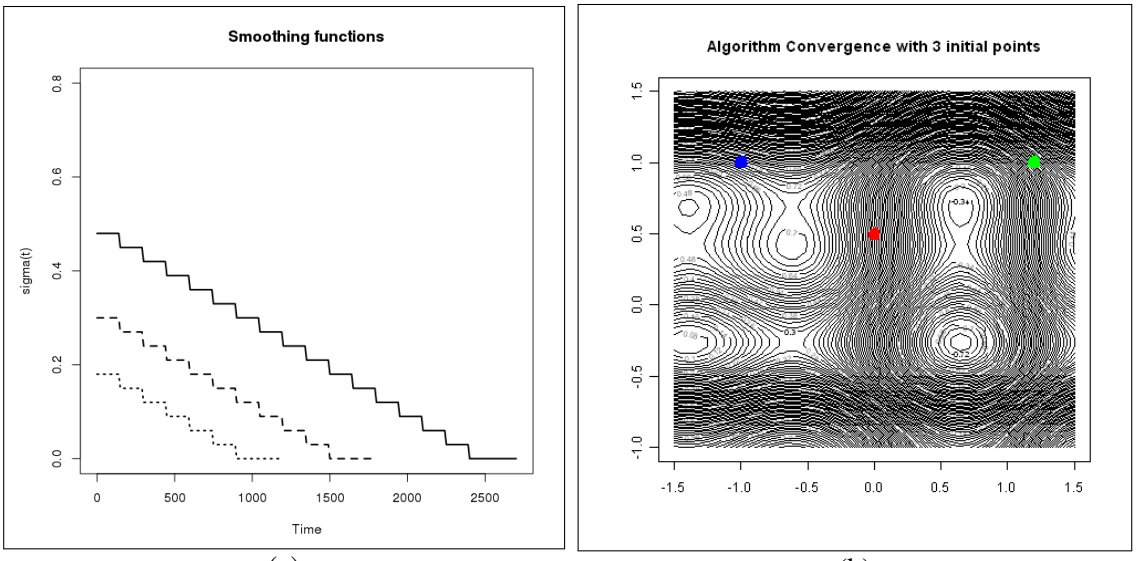
for $\theta_1 \in [-1.5, 1.5]$ and $\theta_2 \in [-1, 1.5]$. (See Figure 9).




(a) (b)

Figure 9: (a) Plot of the multi-optima function. (b) Contour visualization.

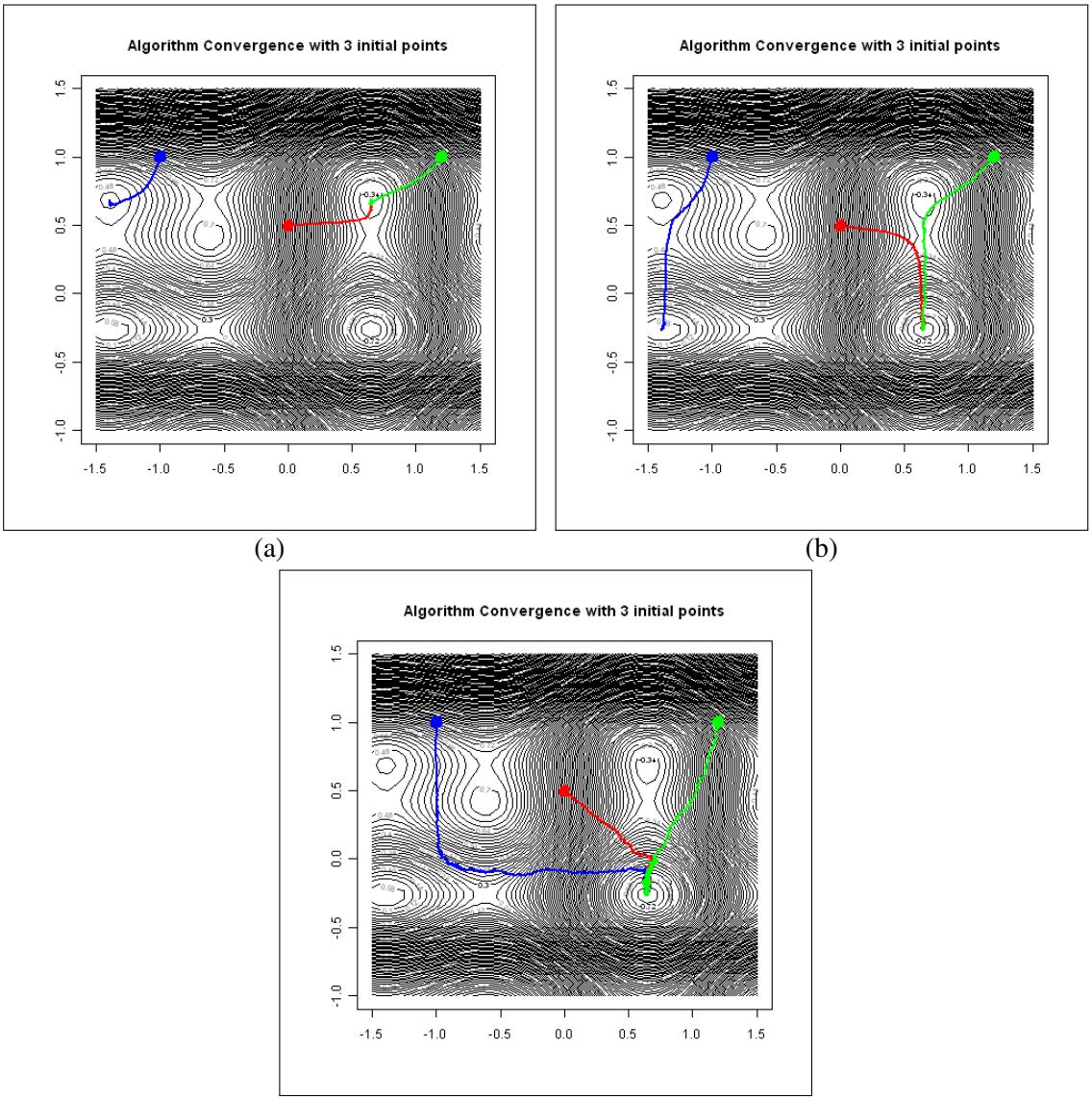
In order to highlight the "global" aspect of our method, we have chosen arbitrarily 3 initial points from which we have run the stochastic algorithm (see Figure 10(b)). The sequences $(\gamma_t)_{t\geq 0}$ and $(\delta_t)_{t\geq 0}$ are the same as in the previous example (Rosenbrock function), i.e $\gamma_t = \dfrac{1}{10^3 + t^{0.6}}$ and $\delta_t = \dfrac{10^{-2}}{t^{0.4}}$. Again we considered 3 smoothing functions given in Figure 10(a) .




(a) (b)

Figure 10: (a) Plot of the smoothing functions used. (b) Initial points chosen for the optimisations.

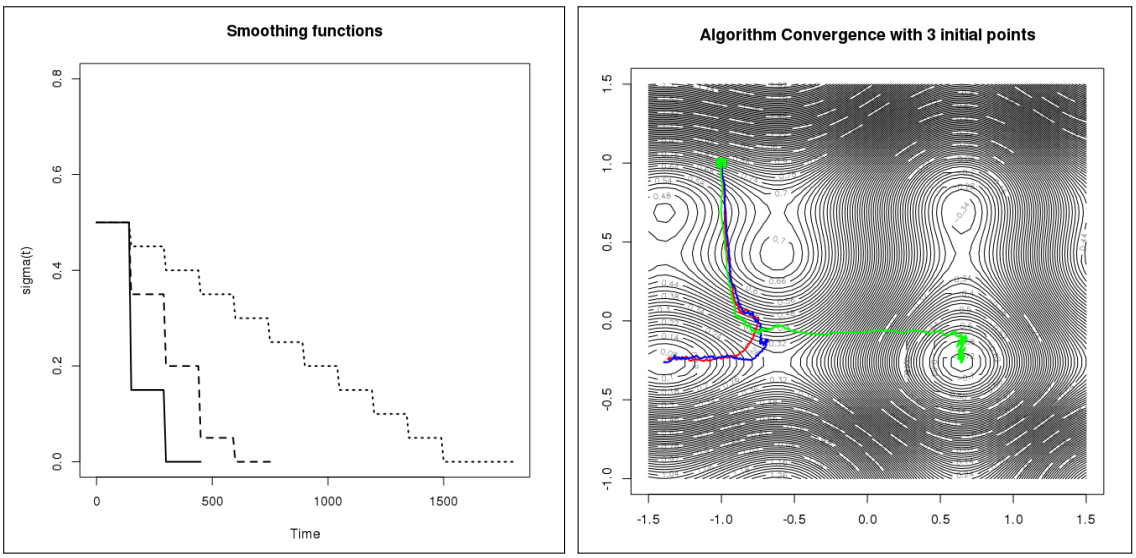
Figure 11 shows the results obtained by running the stochastic algorithm at the 3 initial points and by considering 3 different smoothing functions. We clearly see the crucial role of the smoothing function. In particular, we may notice that the smoothing functions in Figure 10(a) have roughly the same “shape” and only differ from their initial value $\sigma(0)$. Hence, one can say that the convergence results given in Figure 11 depend on the initial value $\sigma(0)$, for some given shape of the smoothing function.
Now, let us investigate for instance what happens when fixing $\sigma(0)$ and varying the shape. Figure 12 shows a simulation by considering three smoothing functions which have the same value at $t = 0$, $\sigma(0) = 0.5$, and with different shapes.




(a) (b)

(c)

Figure 11: (a) Algorithm convergence using the smoothing function in “dotted” line. (b) Algorithm convergence using the smoothing function in “dashed” line. (c) Algorithm convergence using the smoothing function in “solid” line.




(a) (b)

Figure 12: (a) Smoothing functions. (b) Algorithm convergence.

## 5 About the convergence of the algorithm.

The stochastic algorithm we propose writes :

$$\forall l, 1 \leq l \leq k, \quad \theta_{t+1}^l = \theta_t^l - \gamma_{t+1} \frac{H(\boldsymbol{\theta}_t + \delta_{t+1}\mathbf{e}^l, W_{t+1}^{\sigma_{t+1}}) - H(\boldsymbol{\theta}_t - \delta_{t+1}\mathbf{e}^l, W_{t+1}^{\sigma_{t+1}})}{2\delta_{t+1}} \tag{10}$$

We will use the framework of the Kushner & Clark theorem (see [5]) to analyze its behavior. We set:

$$\frac{H(\boldsymbol{\theta}_t + \delta_{t+1}\mathbf{e}^l, W_{t+1}^{\sigma_{t+1}}) - H(\boldsymbol{\theta}_t - \delta_{t+1}\mathbf{e}^l, W_{t+1}^{\sigma_{t+1}})}{2\delta_{t+1}} = \nabla H^l(\boldsymbol{\theta}_t) + \eta_{t+1}^l \,.$$

The notation $\nabla H^l$ means that we consider the $l$-th component of $\nabla H$. We have :

$$\nabla H^l(\boldsymbol{\theta}_t) + \eta_{t+1}^l = \nabla H^l(\boldsymbol{\theta}_t, W_{t+1}^{\sigma_{t+1}}) + \frac{\delta_{t+1}}{4}(\nabla^2 H^{l,l}(\boldsymbol{\xi}_{t,1}, W_{t+1}^{\sigma_{t+1}}) - \nabla^2 H^{l,l}(\boldsymbol{\xi}_{t,2}, W_{t+1}^{\sigma_{t+1}})),$$

where $\boldsymbol{\xi}_{t,1}$ and $\boldsymbol{\xi}_{t,2}$ belongs to $(\boldsymbol{\theta}_t - \delta_{t+1}\mathbf{e}^l, \boldsymbol{\theta}_t + \delta_{t+1}\mathbf{e}^l)$, and the notation $\nabla^2 H^{i,j}$ stands for the element of the Hessian matrix $\nabla^2 H$ at the $i$-th row and $j$-th column.

Let us assume that for all $\boldsymbol{\theta} \in \Theta$:

$$\mathbb{E}\nabla H^l(\boldsymbol{\theta}, W_{t+1}^{\sigma_{t+1}}) = \nabla H_{\sigma_{t+1}}^l(\boldsymbol{\theta}) \,, \quad l = 1, \ldots, k.$$

So that setting $\nabla H^l(\boldsymbol{\theta}_t, W_{t+1}^{\sigma_{t+1}}) - \nabla H_{\sigma_{t+1}}^l(\boldsymbol{\theta}_t) = \varepsilon_{t+1}^l$, we have $\mathbb{E}(\varepsilon_{t+1}^l | \mathcal{F}_t) = 0$ where $\mathcal{F}_t$ is the $\sigma$-field generated by $(W_1^{\sigma_1}, \ldots, W_t^{\sigma_t})$.

We have obtained the decomposition $\eta_{t+1}^l = \varepsilon_{t+1}^l + r_{t+1}^l$, with :

$$r_{t+1}^l = \nabla H_{\sigma_{t+1}}^l(\boldsymbol{\theta}_t) - \nabla H^l(\boldsymbol{\theta}_t) + \frac{\delta_{t+1}}{4}(\nabla^2 H^{l,l}(\boldsymbol{\xi}_{t,1}, W_{t+1}^{\sigma_{t+1}}) - \nabla^2 H^{l,l}(\boldsymbol{\xi}_{t,2}, W_{t+1}^{\sigma_{t+1}})).$$

Thus, denoting $\boldsymbol{\epsilon}_{t+1} = (\varepsilon_{t+1}^1, \ldots, \varepsilon_{t+1}^k)^T$, and $r_{t+1} = (r_{t+1}^1, \ldots, r_{t+1}^k)^T$ the algorithm now writes:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \gamma_{t+1}(\nabla H(\boldsymbol{\theta}) + \boldsymbol{\epsilon}_{t+1} + \mathbf{r}_{t+1}). \tag{11}$$

We use the classical assumptions required on $H$ and the positive steps $\gamma$ and $\delta$ for the convergence of a stochastic gradient algorithm, whose our algorithm is a perturbation.

**Assumption $\mathcal{H}$**

$\mathcal{H}_1$. $\sum_1^\infty \gamma_t = \infty$; $\sum_1^\infty (\frac{\gamma_t}{\delta_t})^2 < \infty$; $\lim_{t\to\infty} \frac{\gamma_t}{\delta_t^2} = 0$;

$\mathcal{H}_2$. $H$ is of class $C^2$; $H \geq 0$; $\lim_{\|\boldsymbol{\theta}\|\to\infty} H(\boldsymbol{\theta}) = \infty$; $\nabla H$ and $\nabla^2 H$ are lipschitz with constant $L$; it exists $C$ such that $\|\nabla H(\boldsymbol{\theta})\|^2 \leq C(1 + H(\boldsymbol{\theta}))$.

It remains to control the two perturbations $\boldsymbol{\epsilon}$ and $\mathbf{r}$. One can easily check that it exists two positive constants $A$, $B$ such that

$$\mathbb{E}(\|\varepsilon_{t+1}\|^2|\mathcal{F}_t) \leq A\,\sigma_{t+1}^2 \quad \text{and} \quad \|r_{t+1}\| \leq B\,(\sigma_{t+1} + \delta_{t+1}^2)\,.$$

As $\sigma_t$ and $\delta_t$ tend to 0, $r_{t+1}$ tends to 0.

Thus we are in position to apply a theorem of convergence of stochastic gradient algorithm (see [1], [3]).

**Theorem 5.1.** *Assume assumption $\mathcal{H}$ holds. Then $a.s.$ $H(\boldsymbol{\theta}_t)$ converges and* $\sum_{t=1}^\infty \gamma_t \|\nabla H(\boldsymbol{\theta}_t)\|^2 < \infty$.

The main consequence is that $\boldsymbol{\theta}_t$ is $a.s.$ bounded. So we may apply the $O.D.E.$ method of Kushner & Clark [5]:

**Theorem 5.2.** *Under the previous assumption $\mathcal{H}$ and if $\nabla H$ has isolated zeros, then $\boldsymbol{\theta}_t$ $a.s.$ converges toward a $\boldsymbol{\theta}^*$ such that $\nabla H(\boldsymbol{\theta}^*) = 0$.*

Unfortunately, as $\sigma_t$ tends to 0 we cannot claim that $\boldsymbol{\theta}^*$ is a global minimum of $H$.
To understand what happens, we make an assumption on the functions $H_\sigma$, $\sigma > 0$.

**Assumption $\mathcal{A}$**
Assume that there exist two finite sequences $\sigma_0 > \sigma_1 > \ldots > \sigma_N = 0$ and $\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N$ such that : $\boldsymbol{\theta}_0$ is the unique and global minimum of $H_{\sigma_0}$, $\boldsymbol{\theta}_0$ belongs to the basin of attraction of $\boldsymbol{\theta}_1$ which is a local minimum of $H_{\sigma_1}$, more generally $\boldsymbol{\theta}_{j-1}$ belongs to the basin of attraction of $\boldsymbol{\theta}_j$ which is a local minimum of $H_{\sigma_j}$, and $\boldsymbol{\theta}_N$ is the global minimum of $H$.

This means that we may find a "cascade" of (local) minima of $H_{\sigma_j}$ that leads to the global minimum of $H$ via a succession of basins of attractions.

Then we may have the following heuristic reasoning: if we run the algorithm with $\sigma_0$ a long enough time, $\boldsymbol{\theta}_t$ approches $\boldsymbol{\theta}_0$ (uniqueness of the minimum of $H_{\sigma_0}$), thus it belongs to the basin of attraction of $\boldsymbol{\theta}_1$. Now we change $\sigma_0$ into $\sigma_1$. As $\gamma_t$ is small enough $\boldsymbol{\theta}_t$ stays in this basin, so by Kushner & Clark lemma it converges to $\boldsymbol{\theta}_1$, so it stays in a basin of attraction of $\boldsymbol{\theta}_2$. Now we change $\sigma_1$ into $\sigma_2$ and so on. We may expect that if we run the algorithm with a slowly decreasing sequence $\sigma_t$ taking values $\sigma_j, 0 \leq j \leq N$, $\boldsymbol{\theta}_t$ will converge to $\boldsymbol{\theta}_N$.

# 6 Conclusion

Here we have presented an attempt to find the global minimum of a "complex" function via a sequence of smoothings.

The key point is that to compute a smoothed version of the function to minimize is in general far too difficult.

So we proposed to avoid this difficulty by introducing the smoothing as the effect of a stochastic approximation scheme.

The simple examples we tested are convincing, nevertheless we are far from having a practical characterisation of functions at which this method may be applied. This will be the topic of a future work.

# References


[1] M. Duflo. *Algorithmes stochastiques*. Springer, 1996.

[2] M. Duflo. *Random iterative models*, volume 34. Springer Verlag, 1997.

[3] J.C. Fort and G. Pagès. Decreasing step stochastic algorithms: As behaviour of weighted empirical measures. *Monte Carlo Methods and Applications*, 8(3):237–270, 2009.

[4] P.J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pages 73–101, 1964.

[5] H.J. Kushner and D.S. Clark. *Stochastic approximation methods for constrained and unconstrained systems*, volume 6. Springer-Verlag New York, 1978.

[6] J.J. Moré and Z. Wu. Global continuation for distance geometry problems. *Preprint Mcs-p, SIAM J. Optimization*, 7(7):814–836, 1995.

[7] Nabil Rachdi, Jean-Claude Fort, and Thierry Klein. Risk bounds for new m-estimation problems. *ESAIM: Probability and Statistics*, 17:740–766, 2013.