A Stochastic Algorithm for Global Optimization with application to M-estimators computation

Nabil Rachdi, Jean-Claude Fort

A Stochastic Algorithm for M-estimators computation

Nabil Rachdi[1]
nabil.rachdi@eads.net
Jean-Claude Fort[2]
jean-claude.fort@parisdescartes.fr

Abstract

Most of statistical procedures consist on estimating parameters by minimizing (or maximizing) some criterion. A minimizing parameter is also called in the literature *M-estimator*, [2]. Depending on the statistical problem and the available information, the criterion may be more or less complicated: non convex, no gradient, non smooth etc... Thus, it can be difficult in practice to compute a *M-estimator*. We propose a new algorithm to compute the parameters, mixing stochastic algorithms and smoothness technics. We will call it S^2Dyn for *Stochastic & Smooth Dynamic* algorithm.

1 Introduction

Let $H : \Theta \longrightarrow \mathbb{R}$ be some mapping, or criterion function, where Θ is a convex and compact set of $\mathbb{R}^k$. Moreover, we assume that H has a unique global minimum $\boldsymbol{\theta}$ over Θ 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 simply (see figure 1(a))

[1]Institut de Mathématiques de Toulouse - EADS Innovation Works, 12 rue Pasteur, 92152 Suresnes
[2]Université Paris Descartes, 45 rue des saints pères, 75006 Paris

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

where C is some 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 consider the estimators resulting in the statistical procedures introduce in the work of Rachdi et al. [4] :

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

where $Y_1, ..., Y_n$ are i.i.d random variables drawn from a distribution Q, $\mathcal{F}$ 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

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

In this paper we 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 the Silverman's rule-of-thumb :

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

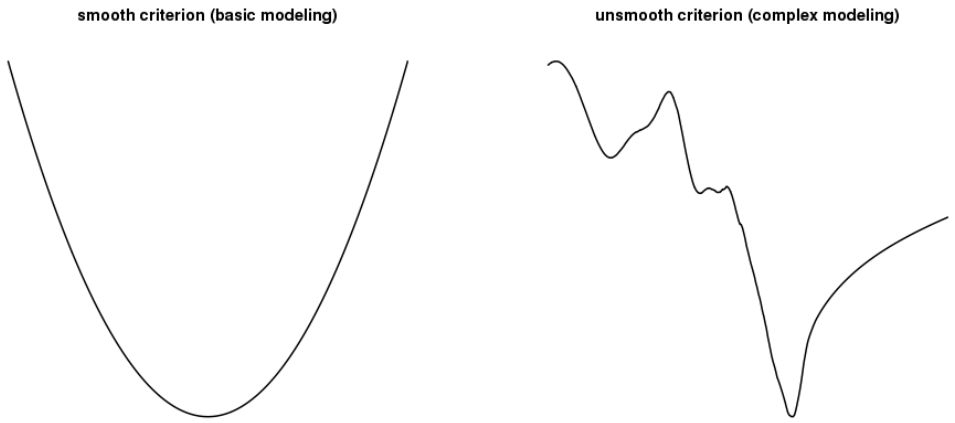
where $\hat{\sigma}_{\boldsymbol{\theta}}$ is the empirical standard deviation of the sample $h(\mathbf{X}_j, \boldsymbol{\theta})$, $j = 1, ..., m$.
Finally H becomes

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

where C is some constant independent of $\boldsymbol{\theta}$.
Recall that h is a computer code, viewed as a *black-box*[3], simulating a physical phenomenon. Typically, h gives solutions of differential equations etc... That implies two important things in order to investigate an algorithm for computing

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

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



(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.

First, the function can be highly non-convex (with many local minima), second we don't have the analytical expression of the gradient. The figure (1b) shows an example of function H resulting of this modeling.

2 Smoothness and stochastic algorithm

In this section, we attempt to overcome irregularities or unsmoothness of the function H, by making a convolution by some appropriate function, and we will see how naturally appears a stochastic algorithm.
The smoothness method is taken from [3] where the main idea is that instead of minimizing directly H, it is to minimize a modified function, smoother than H while controlling the degree of smoothness. However, when the modified function is a convolution of the criterion H with some function, one has to compute multi-dimensional integrals. That makes, in general, not possible the method in high dimension. Moreover, in many applications H is not analytically known, only computable. In this paper we propose optimization procedures based on stochastic algorithms to compute the minimizer, which in any case overcome the computation of the multi-dimensional integrals.

Let g_Σ be the probability density function of a centered random variable of $\mathbb{R}^k$, and Σ be the diagonal matrix $(k \times k)$ $\texttt{diag}(\sigma_1^2, ..., \sigma_k^2)$. For simplicity, suppose that $\sigma^2 = \sigma_l^2$, $l = 1, ..., k$, where $\sigma > 0$, and denote by $g_\Sigma = g_{\sigma^2}$. Denote by λ_Θ the Lebesgue measure on Θ and denote by $L_1(\lambda_\Theta)$ the space of all integrable functions under the measure λ_Θ.

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

which are the convolution of H and g_{σ^2}.

Lemma 2.1. *Let $\mathcal{C}(\Theta)$ be the space of all continuous functions on Θ.*
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 $H \in L_1(\lambda_\Theta)$, 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).
Let consider the following function as an academic example

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

for some constants $a > 0$ 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 2.1. Notice that we have $H_\sigma(\boldsymbol{\theta}) \underset{\sigma \to 0}{\longrightarrow} H(\boldsymbol{\theta})$ but not $H_\sigma(\boldsymbol{\theta}) \underset{\sigma \to +\infty}{\longrightarrow} 0$ ($H \notin L_1(\lambda_\Theta)$). However, for large $\sigma > 0$, $H(\boldsymbol{\theta}) \approx \boldsymbol{\theta}^2 + \sigma^2$ that is very smooth.

The figure (2) shows the behaviour of the function H_σ (with $a = 1$ and $b = 6$) with the parameter σ.
Now the challenge is to compute the minimizer of H_σ which would be, a priori, more tractable than those of H. However, to compute $H_\sigma(\boldsymbol{\theta})$ requires the knowledge of H that is suppose not known analytically. Also, the computation of $H_\sigma(\boldsymbol{\theta})$ needs to integrating on $\Theta \in \mathbb{R}^k$ that can be difficult in general, especially if k is large. The following remark is the motivation of this work.
Notice that

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

where $W^\sigma \sim g_{\sigma^2}$ means that W^σ is a random variable distributed from the density g_{σ^2}. For notational simplicity, we will denote abusively

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

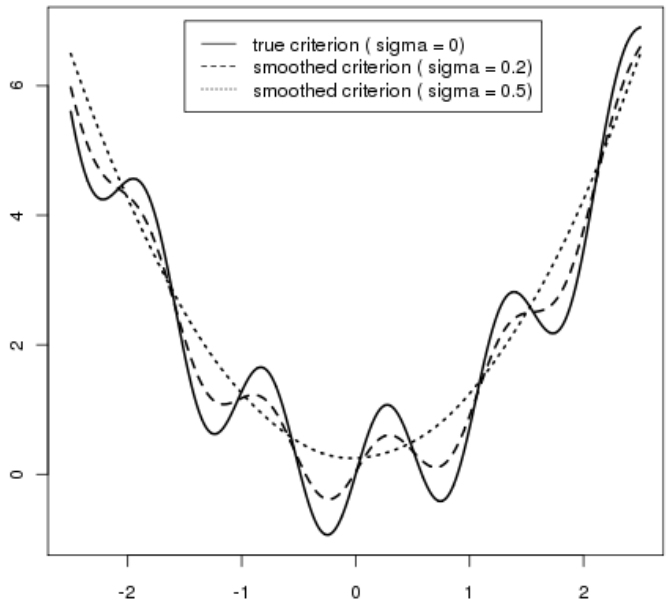


Figure 2: Illustration of the transformation (7) for different values of σ, with $a = 1$ and $b = 6$.

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

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

where $(e^l)_{l=1,\ldots,k}$ is the canonical basis of $\mathbb{R}^k$.

Theorem 2.1. ***Kiefer-Wolfowitz Algorithm convergence***
Let 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

- $\sum_{t\geq 0} \gamma_t = +\infty$

- $\displaystyle\sum_{t\geq 0}\left(\frac{\gamma_t}{\delta_t}\right)^2 < +\infty\,.$

Then, for all $\sigma > 0$

$$\boldsymbol{\theta}_t^{\sigma} \underset{t\to+\infty}{\longrightarrow} \widehat{\boldsymbol{\theta}^{\sigma}} := \underset{\boldsymbol{\theta}\in\Theta}{\text{Argmin}}\; H_{\sigma}(\boldsymbol{\theta}) \quad \textit{almost surely} \quad .$$

3 Construction of the S^2Dyn algorithm

3.1 Sequential approach

The method consists on using sequentially the algorithm (KW) in (9).

Algorithm 1 Sequential method

Require: $\{\sigma_0 > \sigma_1 > ... > \sigma_{q-1} > \sigma_q = 0\}$, γ, δ, $\boldsymbol{\theta}_0^{\sigma_0}$, T_{seq}

Construct the sequence $T_{l+1} = T_l + T_{seq}$, $l \in \{0, ..., q-1\}$, $T_0 = 0$

for i from 0 to q **do**

generate independent $W_1^{\sigma_i}, ..., W_{T_{seq}}^{\sigma_i}$ with $W_t^{\sigma_i} \sim g_{\sigma_i}$

for t from 0 to $T_{seq} - 1$ **do**

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

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

end for

$\boldsymbol{\theta}_0^{\sigma_{i+1}} = \boldsymbol{\theta}_{T_{seq}}^{\sigma_i}$

end for

return $\boldsymbol{\theta}_{T_{seq}}^{\sigma_q}$

Remark 3.1. Notice that the number of iterations T_{seq} could be considered depending on the smoothing parameter σ, for instance T_{seq}^{σ} decreases with σ would mean that we allow more iterations for non smooth functions.

This sequential point of view can be extended to a *dynamic* one where the parameter σ depends on the time t, satisfying $\sigma_t \underset{t\to+\infty}{\longrightarrow} 0\,.$

3.2 Stochastic & Smooth Dynamic algorithm

Now we consider the (KW) algorithm (9) with parameter σ depending on t. Let denote by $\sigma := (\sigma_t)_{t\geq 0}$, $\gamma := (\gamma_t)_{t\geq 0}$, and $\delta := (\delta_t)_{t\geq 0}$ sequences of real numbers decreasing to zero. We propose the following S^2Dyn algorithm.

Algorithm 2 S^2Dyn algorithm

Require: $\sigma : t \mapsto \sigma_t$, γ, δ, $\boldsymbol{\theta}_0$, T_{dyn}

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

for t from 0 to $T_{dyn} - 1$ **do**

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

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

end for

return $\boldsymbol{\theta}_{T_{dyn}}$

Remark 3.2. The term of *Dynamic* would mean that along the time the function H_{σ_t} changes, converging toward the "true" function H.

Notice that the Sequential method (Algorithm 1) is a particular case of the Dynamic one (Algorithm 2). Indeed, let consider the sequences $\{\sigma_0 > \sigma_1 > ... > \sigma_{q-1} > \sigma_q = 0\}$ and $(T_l)_{0 \leq l \leq q-1}$ such that $T_0 = 0$ and $T_{l+1} = T_l + T_{seq}$. Hence, the sequential algorithm is equivalent to the dynamic algorithm by taking

$$\sigma_t = \sum_{l=0}^{q-1} \mathbb{1}_{[T_l, T_{l+1}[}(t)\,\sigma_l\,.$$

Moreover, we have that $T_{dyn} = q\,T_{seq}$.
The function $\sigma : t \mapsto \sigma_t$ will be called *smoothing function* and we will see in the next section that its behaviour is crucial for the convergence our algorithm.

Remark 3.3. The stochastic process $\{\boldsymbol{\theta}_t, t \geq 0\}$ provided by the Algorithm 2 is a Markov Chain.

4 Simulation study

In this section, we test our algorithm on the 1D function (6) (with $a = 1$ and $b = 6$), and on the 2D Rosenbrock function.

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 do not dispose of the gradient of H and that we can not compute H_σ

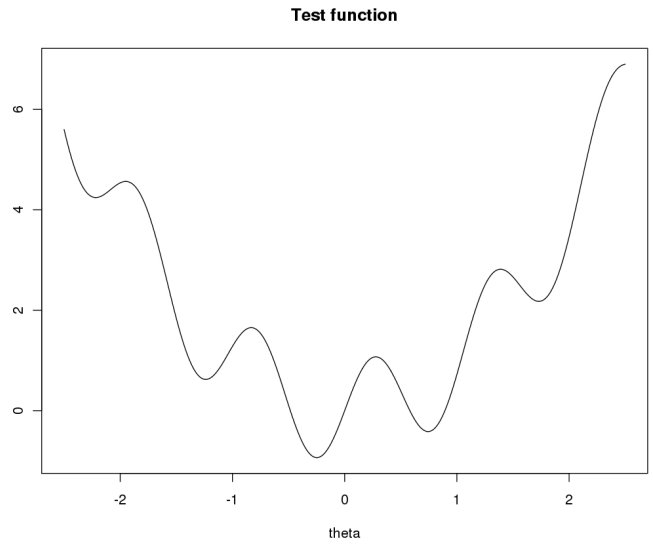
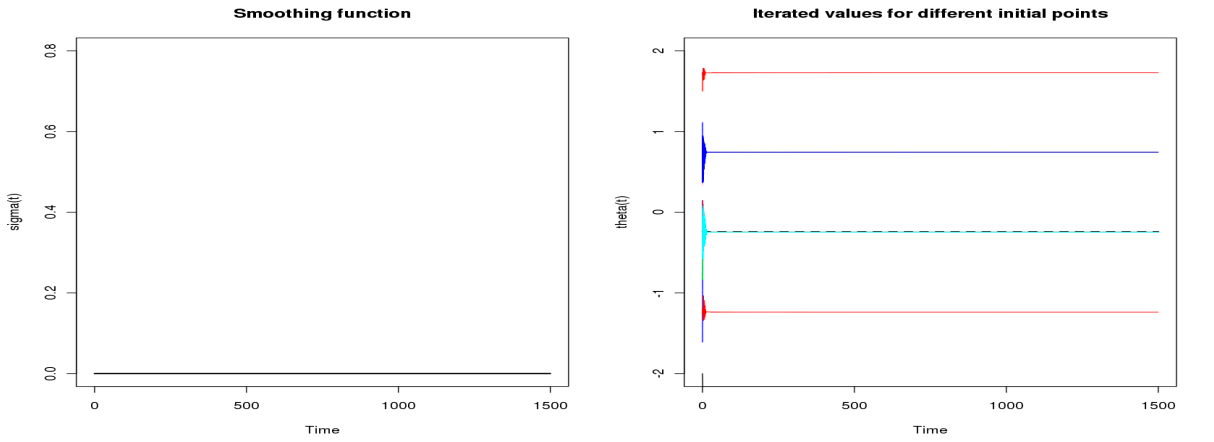


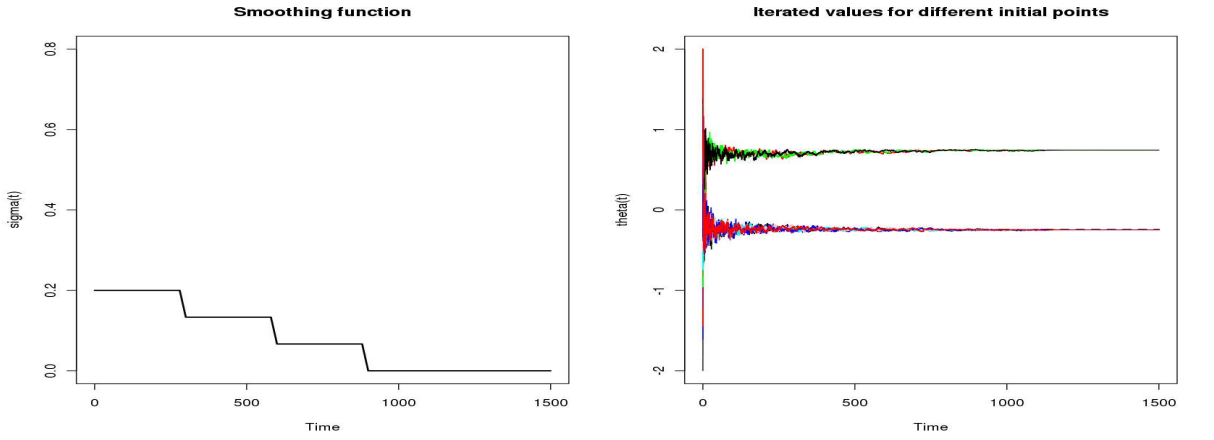
Figure 3: test function $H(\boldsymbol{\theta}) = \theta^2 + \sin(6\,\boldsymbol{\theta})$.

(which is given by (7)).
Now let consider the S^2Dyn algorithm with the following configurations.
Let the "time" be $T_{dyn} = 1500$ with time step $\Delta t = 5.10^{-2}$. Then, consider the following sequences (or functions): $\gamma_t = \dfrac{10^{-1}}{t}$ and $\delta_t = \dfrac{10^{-1}}{t^{0.4}}$ (notice that these sequences satisfy conditions of Theorem (2.1)).

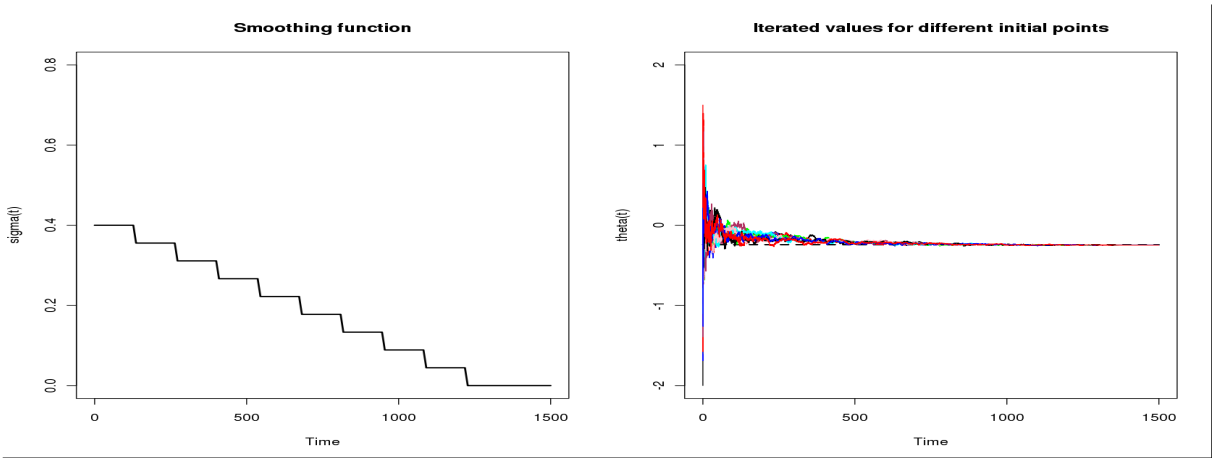
The figure (4) presents the evolution of $\boldsymbol{\theta}_t$ in t at different starting points, for three smoothing functions. In the figure (4a), we consider the trivial smoothing function $\sigma_t = 0$ for all $t \in [0, T_{dyn}]$, e.g there is no *dynamic* during the time, $H_{\sigma=0} = H$ and it amounts to local methods (we see that $\boldsymbol{\theta}_t$ converges to the nearest minimum).
In the figures (4b) and (4c) we consider two others smoothing functions, the first decreases rapidly (b), and the other one decreases slowly (c). It appears that for a suitable function σ (that not decreases too fast), the process $\{\boldsymbol{\theta}_t,\, t \in [0, T_{dyn}]\}$ converges (in some sense) to the minimum for any starting points (figure (4c) right).



(a)



(b)



(c)

Figure 4: Convergence of the S^2Dyn algorithm vs. behaviour of (decreasing rate of) the smoothness function σ.

4.2 2D example

Now, let consider the Rosenbrock function

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

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

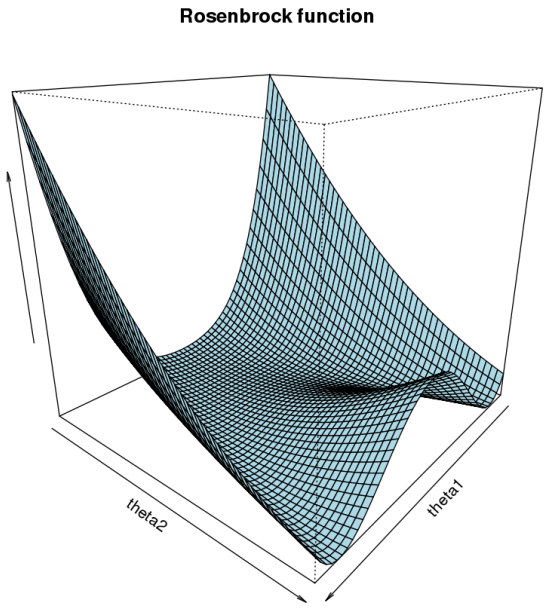
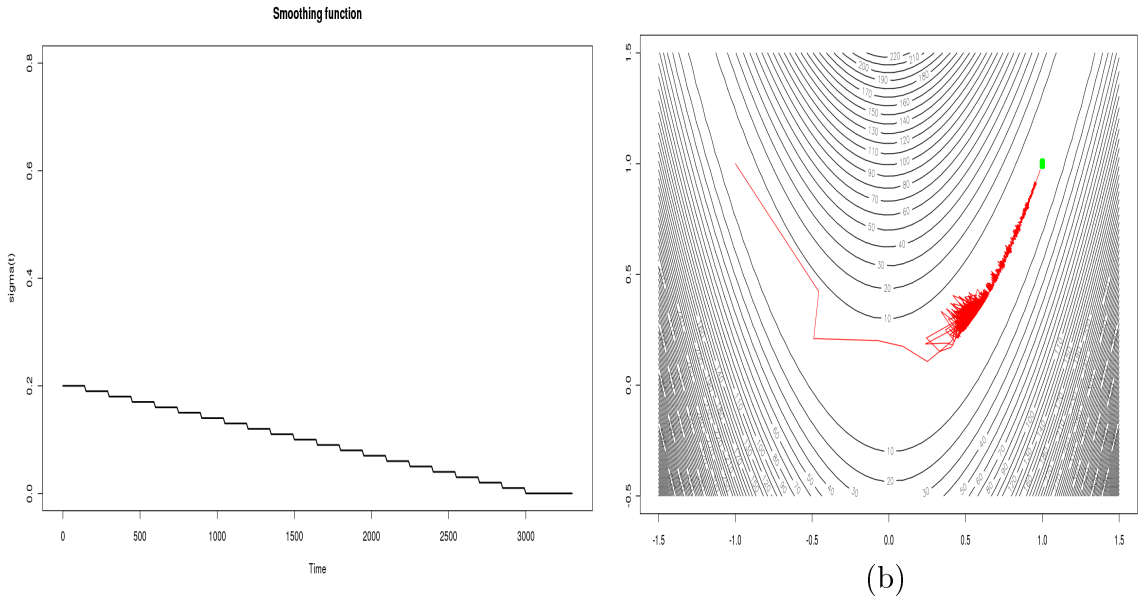


Figure 5: Rosenbrock function

We use the S^2Dyn algorithm with the following sequences: $\gamma_t = \dfrac{1}{10^3 + t^{0.6}}$ and $\delta_t = \dfrac{10^{-2}}{t^{0.4}}$. For $T_{dyn} = 3300$ and a time step $\Delta t = 5.10^{-2}$, the obtained minimum value is $(\theta_1, \theta_2)_{min} = (0.9856077, 0.9713646)$ and the Rosenbrock function evaluated at this point is $H_{min} = 2.07 \times 10^{-4}\,.$
The figure (6) shows the smoothing function used in the algorithm (6a) and the graph of convergence (6b).



(a) (b)

Figure 6: (a) Smoothing function. (b) S^2Dyn algorithm applied to Rosenbrock function.

References

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

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

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

[4] N. Rachdi, Jean-Claude Fort, and Thierry Klein. Oracle inequalities for new M-estimation and model selection problems . 2010.