Journal of Global Optimization **16:** 257–270, 2000.

A Stochastic Algorithm for Constrained Global Optimization

M.C. RECCHIONI and A. SCOCCIA
Istituto di Matematica e Statistica, Facoltà di Economia, Università degli Studi di Ancona, Piazzale Martelli, 60100, Ancona Italy
(e-mail: Recchioni@posta.econ.unian.it; Scoccia@posta.econ.unian.it)

(Received 26 October 1999; accepted in revised form 31 December 1999)

Abstract. We present a stochastic algorithm to solve numerically the problem of finding the global minimizers of a real valued function subject to lower and upper bounds. This algorithm looks for the global minimizers following the paths of a suitable system of stochastic differential equations. Numerical experience on several test problems known in literature is shown.

Key words: Nonlinear programming; Parabolic PDEs; Path following methods

1. Introduction

Let R^n be the n-dimensional real Euclidean space, $\underline{x} = (x_1, x_2, \ldots, x_n)^T \in R^n$ and let k_1, k_2 be non-negative integers such that $k_1 \leqslant k_2 \leqslant n$, we define the following region

$$\Lambda = \left\{ \underline{x} \in R^n \;\middle|\; \begin{array}{ll} m_i \leqslant x_i & i \in \mathrm{I}_1 \\ x_i \leqslant M_i & i \in \mathrm{I}_2 \\ m_i \leqslant x_i \leqslant M_i & i \in \mathrm{I}_3 \end{array} \right\} \tag{1.1}$$

where $\mathrm{I}_1 = \{1, \ldots, k_1\}$, $\mathrm{I}_2 = \{k_1 + 1, \ldots, k_2\}$, $\mathrm{I}_3 = \{k_2 + 1, \ldots, n\}$ and m_i, $i \in \mathrm{I}_1, \mathrm{I}_3$, M_i, $i \in \mathrm{I}_2, \mathrm{I}_3$, are given real constants with $m_i < M_i$, $i \in \mathrm{I}_3$. Let $\overset{0}{\Lambda}$ be the interior of Λ, we consider the problem

$$\min_{\underline{x} \in \Lambda} f(\underline{x}) \tag{1.2}$$

where $f : R^n \to R$ is a real valued function, $f \in C^2(\overset{0}{\Lambda})$.

Let $D(\underline{x})$ be the diagonal matrix $D(\underline{x}) = \mathrm{diag}(d_1(\underline{x}), d_2(\underline{x}), \ldots, d_n(\underline{x}))$ where

$$d_i(\underline{x}) = \begin{cases} x_i - m_i & i \in \mathrm{I}_1 \\ M_i - x_i & i \in \mathrm{I}_2 \\ \dfrac{(x_i - m_i)(M_i - x_i)}{M_i - m_i} & i \in \mathrm{I}_3 \end{cases} \tag{1.3}$$

and let $D^2(\underline{x}) = \mathrm{diag}(d_1^2(\underline{x}), d_2^2(\underline{x}), \ldots, d_n^2(\underline{x}))$.

We assume that there exists a finite solution of the problem (1.2) and we solve this problem following the paths of the stochastic differential equation given by

$$d\underline{x}(t) = -D^2(\underline{x})\nabla_{\underline{x}} f(\underline{x})\, dt + \varepsilon(t)D(\underline{x})\, d\underline{W}(t) \tag{1.4}$$

$$\underline{x}(0) \in \overset{0}{\Lambda} \tag{1.5}$$

where $\nabla_{\underline{x}} f(\underline{x})$ is the gradient of f with respect to $\underline{x}$, $\underline{W}(t)$ is an n-dimensional standard Wiener process and $\varepsilon(t)$ is a given positive function of t such that

$$\lim_{t \to +\infty} \varepsilon(t) = 0\,. \tag{1.6}$$

We note that a method which follows a trajectory of (1.4) when $\varepsilon(t) = 0$, $t > 0$, is local, while the stochastic term introduced in (1.4) when $\varepsilon(t) > 0$, $t > 0$, permits the trajectories to escape from the 'attracting set' of the local minimizers of f on Λ.

The stochastic differential equation (1.4) is inspired to the following stochastic differential equation

$$d\underline{x}(t) = -\nabla_{\underline{x}} f(\underline{x})\, dt + \varepsilon(t)\, d\underline{W}(t) \tag{1.7}$$

$$\underline{x}(0) = \underline{x}_0 \in R^n\,. \tag{1.8}$$

When $\varepsilon(t) = \varepsilon_0 \ \ \forall t > 0$, Equation (1.7) is known as the Smoluchowski–Kramer equation [1, Chap. 8]. Let $p_{\varepsilon_0}(0, \underline{x}_0, t, \underline{x})$ be the transition probability density of the solution process $\underline{x}(t)$ of (1.7), (1.8) with $\varepsilon(t) = \varepsilon_0$, it is known [2, 3, 4] that when f is smooth enough $p_{\varepsilon_0}(0, \underline{x}_0, t, \underline{x}) \to p^{\infty}_{\varepsilon_0}(\underline{x})$ in law as $t \to +\infty$, where $p^{\infty}_{\varepsilon_0}(\underline{x})$ is the stationary solution of the following Fokker Planck equation

$$\frac{\partial p_{\varepsilon_0}}{\partial t} = \sum_{i=1}^{n} \left[\frac{\partial}{\partial x_i} \left(p_{\varepsilon_0} \frac{\partial f}{\partial x_i} \right) + \frac{\varepsilon_0^2}{2} \frac{\partial^2 p_{\varepsilon_0}}{\partial x_i^2} \right] \qquad \underline{x} \in R^n \tag{1.9}$$

with

$$\lim_{t \to 0} p_{\varepsilon_0}(0, \underline{x}_0, t, \underline{x}) = \delta(\underline{x} - \underline{x}_0)\,, \tag{1.10}$$

which satisfies

$$\int_{R^n} d\underline{x}\ p_{\varepsilon_0}(\underline{x}) = 1\,. \tag{1.11}$$

It is easy to see that if $\int_{R^n} d\underline{x}\, \exp(-2f(\underline{x})/\varepsilon_0^2) < +\infty$ then

$$p^{\infty}_{\varepsilon_0}(\underline{x}) = N_{\varepsilon_0} \exp\left(-\frac{2f(\underline{x})}{\varepsilon_0^2} \right) \tag{1.12}$$

where $N_{\varepsilon_0} = [\int_{R^n} d\underline{x}\, \exp(-2f(\underline{x})/\varepsilon_0^2)]^{-1}$.

The limit density (1.12) is independent of $\underline{x}_0$ and when f has only a finite number of isolated global minimizers this limit density is peaked around the global minimizers of f and the peaks are narrower when ε_0 is smaller. On this property of the limit density (1.12) is based the method proposed in [1, 2] in the case of

unconstrained optimization. Such method looks for the global minimizers of f following the paths of (1.7) with asymptotic condition (1.6). Mathematical foundation of it is given in [3] and a survey on stochastic techniques for global optimization is given in [9].

Let us go back to our problem (1.2). In this paper, first of all, we prove that the solution process of (1.4) belongs to the feasible region Λ if the initial point is an interior point of Λ (see Lemma 2.1). Then, we investigate if such process has theoretical properties analogous to those of the solution process of (1.7). We make this analysis looking for the limit density of the process (1.4), (1.5) with $\varepsilon(t) = \varepsilon_0$ (see Theorem 2.3) and we show that the behavior as $\varepsilon_0 \to 0$ of such limit density is analogous to that of the limit density (1.12) for an unconstrained optimization problem. So we propose a method which attempts to approach a solution of the problem (1.2) following a trajectory of (1.4), (1.5) with a suitable function ε which satisfies condition (1.6). In practice, let $\{t_j\}$ and $\{\varepsilon_j\}$ $j = 0, 1, 2, \ldots$ be sequences of real numbers such that

$$\begin{aligned} &0 \leqslant t_j < t_{j+1} \qquad \forall j = 0, 1, \ldots, \qquad \lim_{j \to +\infty} t_j = +\infty \\ &\varepsilon_j \geqslant \varepsilon_{j+1} \qquad \forall j = 0, 1, \ldots, \qquad \lim_{j \to +\infty} \varepsilon_j = 0 \end{aligned} \tag{1.13}$$

we choose the following piecewise constant function

$$\varepsilon(t) = \varepsilon_j \qquad t_j \leqslant t < t_{j+1} \qquad \forall j = 0, 1, \ldots, \tag{1.14}$$

and we look for a solution of the problem (1.2) by observing the asymptotic value as $t \to +\infty$ of a numerically computed sample trajectory of (1.4) with the function ε given by (1.14) and initial condition (1.5). The motivation of the method is the following: if the function ε is fixed at each value ε_j for a period (t_j, t_{j+1}) large enough we can expect that the transition density $p_{\varepsilon_j}(t_j, \underline{x}_j, t, \underline{x})$, where $\underline{x}_j = \underline{x}(t_j)$, approaches the limit density $p^{\infty}_{\varepsilon_j}(\underline{x})$ which is independent of $\underline{x}_j$ and which is more concentrated around the global minimizers of f when ε_j decreases.

In Section 2 we present the method and some properties of the solution process of (1.4), (1.5). In Section 3 we show a numerical integration of (1.4) and we give a brief description of the algorithm. In Section 4 we propose a modification of the algorithm suggested by the theoretical result shown in Section 2. Finally, in Section 5 we present some numerical results obtained applying the original algorithm, the modified one and the simulated annealing on test problems known in literature.

2. The stochastic differential equation and the constrained global optimization

Let us consider the stochastic differential equations

$$d\underline{x}(t) = -D^2(\underline{x}) \nabla_{\underline{x}} f(\underline{x})\, dt + \varepsilon(t) D(\underline{x})\, d\underline{W}(t) \tag{2.1}$$

$$\underline{x}(0) \in \overset{0}{\Lambda}, \tag{2.2}$$

where $D(\underline{x})$ is the matrix introduced in Section 1, f is the objective function of the problem (1.2), $\underline{W}(t)$ is an n-dimensional standard Wiener process and

$$\varepsilon(t) = \varepsilon_j \qquad t_j \leqslant t \leqslant t_{j+1} \qquad \forall j = 0, 1, \ldots, \tag{2.3}$$

where $\{t_j\}$ and $\{\varepsilon_j\}$ $j = 0, 1, 2, \ldots$ are sequences of real numbers such that

$$0 \leqslant t_j < t_{j+1} \qquad \varepsilon_j \geqslant \varepsilon_{j+1} \qquad \forall j = 0, 1, \ldots, \qquad \lim_{j \to +\infty} t_j = +\infty \quad \lim_{j \to +\infty} \varepsilon_j = 0 . \tag{2.4}$$

LEMMA 2.1. *Let $f \in C^2(\overset{0}{\Lambda})$ and let $\underline{x}(t)$ be a solution process of (2.1), (2.2) and $\underline{x}^* \in \partial\Lambda$, if $\forall i$ such that $d_i(\underline{x}^*) = 0$ we have*

$$x_i^* = m_i \Rightarrow \lim_{\underline{x} \to \underline{x}^*} \frac{\partial f}{\partial x_i}(\underline{x}) < +\infty \tag{2.5}$$

$$x_i^* = M_i \Rightarrow \lim_{\underline{x} \to \underline{x}^*} \frac{\partial f}{\partial x_i}(\underline{x}) > -\infty \tag{2.6}$$

then $\underline{x}(t) \in \Lambda$ for any value of t.

Proof. If $\underline{x}^* \in \partial\Lambda$ with $x_i^* = m_i$ then $d_i(\underline{x}^*) = 0$ and because of (2.5) we have

$$\lim_{\underline{x} \to \underline{x}^*} - d_i^2(\underline{x}) \frac{\partial f}{\partial x_i} \geqslant 0 \tag{2.7}$$

Analogously, if $\underline{x}^* \in \partial\Lambda$ with $x_i^* = M_i$ then we have $d_i(\underline{x}^*) = 0$ and because of (2.6) we have

$$\lim_{\underline{x} \to \underline{x}^*} - d_i^2(\underline{x}) \frac{\partial f}{\partial x_i} \leqslant 0 \tag{2.8}$$

This means that the trajectory $\underline{x}(t)$ cannot cross the boundary of Λ. □

When $\varepsilon(t) = 0$, $\forall t > 0$ the stochastic differential equation (2.1) reduces to the following ordinary differential equation

$$d\underline{x}(t) = -D^2(\underline{x}) \nabla_{\underline{x}} f(\underline{x})\, dt . \tag{2.9}$$

LEMMA 2.2. *The trajectories solution of the ordinary differential equation (2.9) follow the steepest descent direction with respect to the Riemannian metric $G(\underline{x}) = D(\underline{x})^{-2}$ defined for $\underline{x} \in \overset{0}{\Lambda}$.*

Proof. The steepest descent direction $\underline{v} \in R^n$ of a function f defined on an open set with respect to the Riemannian metric $d\underline{x}^2 = \sum_{i=1}^n \frac{dx_i\, dx_i}{d_i^2(\underline{x})}$ at $\underline{x} \in R^n$ is the direction that minimizes the linear functional $\nabla_{\underline{x}} f(\underline{x})^T \underline{v}$, that is

$$\min_{\underline{v} \in R^n} \nabla_{\underline{x}} f(\underline{x})^T \underline{v} \tag{2.10}$$

on the ellipsoid

$$\underline{v}^T D^{-2}(\underline{x})\underline{v} = r^2 . \tag{2.11}$$

The Lagrange conditions for this problem are:

$$\begin{cases} \nabla_{\underline{x}} f(\underline{x}) = 2\mu D^{-2}(\underline{x})\underline{v} \\ \underline{v}^T D^{-2}(\underline{x})\underline{v} = r^2 \end{cases} \tag{2.12}$$

so we find

$$\underline{v} = \frac{1}{2\mu} D^2(\underline{x})\nabla_{\underline{x}} f(\underline{x}) . \tag{2.13}$$

It is easy to see that the vector $\underline{w} \in R^n$, independent of r, given by

$$\underline{w} = -D^2(\underline{x})\nabla_{\underline{x}} f(x) \tag{2.14}$$

points in the minimizing direction for the linear functional considered. □

Since a method for solving problem (1.2) based on the ordinary differential equation (2.9) is local (it depends on the behavior of f along the trajectory), we propose a method that attempts to get a solution of problem (1.2) looking at the asymptotic value of a sample trajectory obtained by the numerical integration of (2.1), (2.2).

Let $\underline{x}(t)$ be a solution process of (2.1) for $t > t_j$ with condition

$$\underline{x}(t_j) = \underline{x}_j , \tag{2.15}$$

for any Borel set $\mathbf{A} \subseteq R^n$, we define

$$\mathrm{P}_{\varepsilon_j}(t_j, \underline{x}_j, t, \mathbf{A}) = P\{\underline{x}(t) \in \mathbf{A}\} \tag{2.16}$$

where $P\{\cdot\}$ is the probability of $\{\cdot\}$ and $\mathrm{P}_{\varepsilon_j}(t_j, \underline{x}_j, t, \underline{x})$ is the transition probability of $\underline{x}(t)$. In the following we assume that $\forall j = 0, 1, 2, \ldots$ there exists a transition probability density $p_{\varepsilon_j}(t_j, \underline{x}_j, t, \underline{x})$ such that

$$\mathrm{P}_{\varepsilon_j}(t_j, \underline{x}_j, t, \mathbf{A}) = \int_{\mathbf{A}} d\underline{x}\ p_{\varepsilon_j}(t_j, \underline{x}_j, t, \underline{x}) \tag{2.17}$$

which is a solution of the following Fokker Planck equation

$$\frac{\partial p_{\varepsilon_j}}{\partial t} = \sum_{i=1}^{n} \left[\frac{\partial}{\partial x_i} \left(d_i^2(\underline{x}) p_{\varepsilon_j} \frac{\partial f}{\partial x_i} \right) + \frac{\varepsilon_j^2}{2} \frac{\partial^2}{\partial x_i^2} (d_i^2(\underline{x}) p_{\varepsilon_j}) \right] \qquad \underline{x} \in \overset{0}{\Lambda} \tag{2.18}$$

with

$$\lim_{t \to t_j} p_{\varepsilon_j}(t_j, \underline{x}_j, t, \underline{x}) = \delta(\underline{x} - \underline{x}_j) . \tag{2.19}$$

Moreover we assume that the transition probability density $p_{\varepsilon_j}(t_j, \underline{x}_j, t, \underline{x})$ approaches as $t \to +\infty$ a function $p^{\infty}_{\varepsilon_j}(\underline{x})$ which is the probability density of a random

variable $\underline{x}_\infty$ such that $\underline{x}(t) \to \underline{x}_\infty$ as $t \to +\infty$ in law, and we assume that $p^\infty_{\varepsilon_j}(\underline{x})$ is the unique stationary solution of (2.18) with $\int_\Lambda d\underline{x}\ p_{\varepsilon_j}(\underline{x}) = 1$. With such assumption, we are able to find such stationary probability density $p^\infty_{\varepsilon_j}(\underline{x})$, so we can study the behavior of $p^\infty_{\varepsilon_j}(\underline{x})$ as $\varepsilon_j \to 0$ in order to give a heuristic motivation to the method we propose.

THEOREM 2.3. *Let us consider the Cauchy problems (2.1), (2.2) with* $\varepsilon(t) = \varepsilon_j$, $\forall t > t_j$ *and let* $f \in C^2(\overset{0}{\Lambda})$ *be the objective function of the problem (1.2) satisfying*

$$\int_\Lambda \frac{exp(-\alpha f(\underline{x}))}{\Pi_{i=1}^n d_i^2(\underline{x})}\, d\underline{x} < +\infty \forall \alpha > 0\,, \tag{2.20}$$

where $d_i(\underline{x})$, $i = 1, \ldots, n$ *are given in (1.3). If there exists a limit density* $p^\infty_{\varepsilon_j}(\underline{x})$ *of (2.1), (2.2) and if* $p^\infty_{\varepsilon_j}(\underline{x})$ *is the unique stationary solution of the Fokker Planck equation, that is*

$$\frac{\partial p^\infty_{\varepsilon_j}}{\partial t} = \sum_{i=1}^n \left[\frac{\partial}{\partial x_i} \left(p^\infty_{\varepsilon_j} d_i^2(\underline{x}) \frac{\partial f}{\partial x_i} \right) + \frac{\varepsilon_j^2}{2} \frac{\partial^2}{\partial x_i^2} (d_i^2(\underline{x}) p^\infty_{\varepsilon_j}) \right] = 0 \quad \underline{x} \in \overset{0}{\Lambda} \tag{2.21}$$

with the condition

$$\int_\Lambda d\underline{x}\ p^\infty_{\varepsilon_j}(\underline{x}) = 1\,, \tag{2.22}$$

then $p^\infty_{\varepsilon_j}$ *is given by*

$$p^\infty_{\varepsilon_j}(\underline{x}) = N_{\varepsilon_j} \left(\prod_{i=1}^n d_i^2(\underline{x}) \right)^{-1} \exp\left(-2 \frac{f(\underline{x})}{\varepsilon_j^2} \right) \qquad \underline{x} \in \overset{0}{\Lambda}\,, \tag{2.23}$$

where N_{ε_j} *is the normalization constant.*

Proof. First, we observe that (2.20) implies that f satisfies the assumptions of Lemma 2.1, so that $\underline{x}(t) \in \Lambda$ for any value of t. Then, the thesis follows by substituting formula (2.23) into (2.21). □

REMARK 2.4. Using the change of variables

$$y_i = \begin{cases} \log(x_i - m_i) & i \in \mathrm{I}_1 \\ -\log(M_i - x_i) & i \in \mathrm{I}_2 \\ \log(x_i - m_i) - \log(M_i - x_i) & i \in \mathrm{I}_3 \end{cases} \tag{2.24}$$

and Ito's Lemma, Equations (2.1), (2.2) with $\varepsilon(t) = \varepsilon_j$, $\forall t > t_j$, as in Theorem 2.3 become

$$d\underline{y}(t) = -\nabla_{\underline{y}} \tilde{f}(\underline{y})\, dt + \varepsilon(t)\, d\underline{W}(t) \tag{2.25}$$

$$\underline{y}(0) = \underline{y}(\underline{x}(0)) \in R^n \tag{2.26}$$

where

$$\tilde{f}(\underline{y}) = f(\underline{x}(\underline{y})) + \frac{\varepsilon(t)^2}{2} \sum_{i=1}^{n} \log d_i(\underline{x}(\underline{y})) \,. \tag{2.27}$$

Equation (2.25) is the stochastic differential equation (1.7) for an unconstrained optimization problem with the objective function $\tilde{f}(\underline{y})$. Writing the function $\tilde{f}(\underline{y})$ in the following way

$$\tilde{f}(\underline{y}) = -\frac{\varepsilon(t)^2}{2} \log\left(\frac{\exp\left(\dfrac{-2}{\varepsilon(t)^2} f(\underline{x}(\underline{y})) \right)}{\Pi_{i=1}^{n} d_i(\underline{x}(\underline{y}))} \right) \tag{2.28}$$

we can show that under the assumptions of Theorem 2.3, the limit density of (2.25), (2.26) as $\varepsilon_j \to 0$ is peaked up on the global minimizers of $\tilde{f}(\underline{y})$ and such minimizers as $\varepsilon_j \to 0$ approach with the global minimizers of $f(\underline{x}(\underline{y}))$. This means that the behavior of the limit density (2.23) as $\varepsilon_j \to 0$ is analogous to the behavior as the limit density for an unconstrained optimization problem. So that it is plausible to look for the global minimizer of f on Λ choosing the function $\varepsilon(t)$ as in (2.3) and integrating Equation (2.1) with (2.2) in order to observe the asymptotic value of the trajectory solution of (2.1), (2.2). This means that we integrate Equation (1.4) with $\varepsilon(t) = \varepsilon_j$ for large time interval $[t_j, t_{j+1})$ to permit the system to 'reach the equilibrium' and then we take $\varepsilon(t) = \varepsilon_{j+1} < \varepsilon_j \; \forall t \in [t_{j+1}, t_{j+2})$. Indeed this is only a heuristic justification to our method, a rigorous foundation deserves further investigation.

3. Description of the algorithm

In this section we propose an interior point algorithm for solving problem (1.2). The ideas on which this algorithm is based on are analogous to those presented in [1, 2]. In fact the algorithm approximates a solution of problem (1.2) looking at the asymptotic value as $t \to +\infty$ of a numerically computed sample trajectory of (2.1), (2.2) with the function (2.3), so that the basic step of this algorithm is the numerical discretization of (2.1). We discretize the Cauchy problem (2.1) using the Eulero–Cauchy method.

Let $h_k > 0$ the step-length of the kth iteration, $t_k = \Sigma_{i=0}^{k-1} h_i$, $t_0 = 0$, $k = 0, 1, \ldots$ and let $\underline{u}_k$ be a random vector sampled from an n-dimensional standard Gaussian distribution, then the numerical approximation $\underline{x}^{k+1}$ of $\underline{x}(t_{k+1})$ is obtained by the following finite difference scheme:

$$\underline{x}^{k+1} - \underline{x}^{k} = -h_k D^2(\underline{x}^{k}) \nabla_{\underline{x}} f(\underline{x}^{k}) + \varepsilon_j \sqrt{h_k} D(\underline{x}^{k}) \underline{u}_k \tag{3.1}$$

$$\underline{x}^{0} \in \overset{0}{\Lambda} \,, \qquad j = 0, 1, 2 \ldots \quad k = 0, 1, 2, \ldots \tag{3.2}$$

where $\sqrt{h_k}\,\underline{u}_k = \underline{W}(t_{k+1}) - \underline{W}(t_k)$, as a consequence of the properties of the Wiener process.

The algorithm follows simultaneously a given number of trajectories $\{\underline{x}^k\}$ $k = 0, 1, 2, \ldots$, for a period of observation whose duration is a given number of time integration steps. During this period of observation the index j is constant, so that ε_j is a parameter, while the step-length h_k is adjusted by the algorithm in such a manner that $\underline{x}^{k+1} \in \overset{0}{\Lambda}$, $\forall k = 0, 1, 2, \ldots$.

At the end of every period the trajectories are compared and they are ordered according with the value that the objective function assumes on them. Then a trajectory is discarded and replaced with a new trajectory starting from the best point reached until the current iteration. Successively the value of ε_j is decreased according with a reduction coefficient r_ε (i.e. $\varepsilon_{j+1} = r_\varepsilon \varepsilon_j$) and a new period of observation is computed.

We note that the perturbation coefficient ε_j must go to zero very slowly in order to permit the trajectories to escape from the 'attracting set' of the local minimizers. This means that a large number of steps is required anyway and for this reason we choose the Euler–Cauchy method for cheap basic time steps.

If $\varepsilon_j \leqslant \varepsilon_{\min}$, where $\varepsilon_{\min}$ is a given positive constant, all the trajectories are stopped and just a new trajectory is computed starting from the best point reached and using (3.1) with $\varepsilon_j = 0$. Finally the algorithm stops when this new trajectory satisfies the following criterion

$$|f(\underline{x}^{k+1}) - f(\underline{x}^k)| < fappr \tag{3.3}$$

where *fappr* is a fixed positive constant.

4. A modification of the algorithm

In Section 2 we gave a heuristic justification to our method based of the behavior of the limit density (2.23) of Theorem 2.3 as $\varepsilon_j \to 0$. Let we observe that in Theorem 2.3 the assumption (2.20) implies that $f \to +\infty$ $\forall \underline{x} \in \partial\Lambda$, in fact only in this case the limit density (2.23) is normalizable. In Section 5 we show that the algorithm works also when the test problems considered do not satisfy (2.20). However, in this section, we propose a modification of the method presented in Sections 2 and 3 that does not require the assumption (2.20) on f.

The idea consists of spreading out the feasible region Λ in the following way

$$\tilde{\Lambda} = \left\{ \underline{x} \in R^n \;\middle|\; \begin{array}{ll} m_i - \beta \leqslant x_i & i \in \mathrm{I}_1 \\ x_i \leqslant M_i + \beta & i \in \mathrm{I}_2 \\ m_i - \beta \leqslant x_i \leqslant M_i + \beta & i \in \mathrm{I}_3 \end{array} \right\} \tag{4.1}$$

where β is a positive constant, and considering the new function:

$$\tilde{f}(\underline{x}) = \begin{cases} f(\underline{x}) & \underline{x} \in \Lambda \\ f(\underline{x}) + \sum_{i=1}^{n} g_i(\underline{x}) & \underline{x} \in \tilde{\Lambda} \backslash \Lambda \end{cases} \tag{4.2}$$

where

$$g_i(\underline{x}) = \begin{cases} \dfrac{(m_i - x_i)^p}{(x_i - m_i + \beta)^q} & m_i - \beta < x_i < m_i\,, \quad i \in \mathrm{I}_1 \vee \mathrm{I}_3 \\ \dfrac{(x_i - M_i)^p}{(M_i - x_i + \beta)^q} & M_i < x_i < M_i + \beta\,, \quad i \in \mathrm{I}_2 \vee \mathrm{I}_3 \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

with $p, q \in R$ such that

$$2 < p < q - 1\,. \tag{4.4}$$

In such a manner we have a function $\tilde{f}$ defined on $\tilde{\Lambda}$ that satisfies the assumptions of Theorem 2.3, in particular $\tilde{f} \to +\infty\ \forall \underline{x} \in \partial\tilde{\Lambda}$. In this case the diagonal matrix $D(\underline{x})$ becomes $D_\beta(\underline{x}) = \mathrm{diag}(d_{\beta_1}(\underline{x}),\ d_{\beta_2}(\underline{x}), \ldots, d_{\beta_n}(\underline{x}))$ with

$$d_{\beta_i}(\underline{x}) = \begin{cases} x_i - m_i + \beta & i \in \mathrm{I}_1 \\ M_i - x_i + \beta & i \in \mathrm{I}_2 \\ \dfrac{(x_i - m_i + \beta)(M_i - x_i + \beta)}{M_i - m_i + 2\beta} & i \in \mathrm{I}_3 \end{cases} \tag{4.5}$$

and the stochastic differential equation (1.4), (1.5) become

$$d\underline{x}(t) = (-D^2_\beta(\underline{x})\nabla_{\underline{x}} f(x) + \underline{b}_\beta(\underline{x}))\, dt + \varepsilon(t) D_\beta(\bar{x})\, d\underline{W}(t) \tag{4.6}$$

$$\underline{x}(0) \in \overset{0}{\tilde{\Lambda}} \tag{4.7}$$

where $\overset{0}{\tilde{\Lambda}}$ is the interior of $\tilde{\Lambda}$ and $\underline{b}_\beta(\underline{x}) \in R^n$ is the vector whose components are

$$b_{\beta_i}(\underline{x}) = d^2_{\beta_i}(\underline{x}) \frac{\partial g_i(\underline{x})}{\partial x_i} \qquad i = 1, \ldots, n\,. \tag{4.8}$$

When $\beta \to 0$ we have $\tilde{\Lambda} \to \Lambda$ and $\tilde{f} \to f$, so we propose to solve the problem (1.2) following the path of the stochastic differential equation (4.6), (4.7) with the piecewise constant function ε given by (2.3) and with $\beta \to 0$ as $t \to +\infty$.

We can see with an easy computation that (4.4) and (4.8) imply that $|b_{\beta_i}(\underline{x})| \to +\infty$ *as* $\underline{x} \to \partial\tilde{\Lambda}$ strong enough to push the trajectory back to the feasible region Λ as $\beta \to 0$, both from the theoretical and from the computational point of view.

5. Numerical results

In this section we show some numerical results obtained applying the new algorithms proposed in this paper on some test problems known in literature. In the following, we refer to the algorithm described in Section 3 as A1-algorithm and to the algorithm in Section 4 as A2-algorithm.

Every test problem consists of an objective function f, a feasible region Λ and an

initial feasible point $\underline{x}_0 \in R^n$. The starting point $\underline{x}_0$ has been chosen in the 'attracting set' of a local minimizer.

The algorithm has been coded in Fortran 77 language and tested on a P.C. Pentium 300 MHz in double precision arithmetic.

TEST 1. Sinusoidal test function:

$$\min_{\underline{x} \in \Lambda} -(2.5 \operatorname{sen} x_1 \operatorname{sen} x_2 + \operatorname{sen} 5x_1 \operatorname{sen} 5x_2) \tag{5.1}$$

with

$$\Lambda = \{\underline{x} \in R^2 \,|\, 0 \leqslant x_i \leqslant \pi\,, \quad i = 1, 2\} \tag{5.2}$$

The starting point is $\underline{x}_0 = (3.14, 3.14)^T$ and the global minimizers di f in Λ is $\underline{x}^* = (\pi/2, \pi/2)^T$.

Test problems 2–5 have the following objective function (Levy and Montalto [6]):

$$\min_{\underline{x} \in \Lambda} \frac{\pi}{n} \left\{ 10 \sin^2(\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 [1 + 10 \sin^2(\pi x_{i+1})] + (x_n - 1)^2 \right\} \tag{5.3}$$

where Λ is given by

$$\Lambda = \{\underline{x} \in R^n \,|\, -10 \leqslant x_i \leqslant 10\,, \quad i = 1, 2, \ldots, n\}\,. \tag{5.4}$$

The objective function in (5.3) has about 10^n local minima in Λ and only a global minimizer at $\underline{x}^* = (1, 1, \ldots, 1)^T \in R^n$.

TEST 2. Solving (5.3), (5.4) with $n = 5$ and $\underline{x}_0 = \underline{0} \in R^5$.

TEST 3. Solving (5.3), (5.4) with $n = 8$ and $\underline{x}_0 = \underline{0} \in R^8$.

TEST 4. Solving (5.3), (5.4) with $n = 10$ and $\underline{x}_0 = \underline{0} \in R^{10}$.

TEST 5. Solving (5.3), (5.4) with $n = 15$ and $\underline{x}_0 = \underline{0} \in R^{15}$.

TEST 6. This test problem, proposed in [7], is given by

$$\min_{\underline{x} \in \Lambda} -\left(\sum_{i=1}^{8} x_i^2 \right) \left(\sum_{i=1}^{8} x_i^4 \right) + \left(\sum_{i=1}^{8} x_i^3 \right)^2 \tag{5.5}$$

with

$$\Lambda = \{\underline{x} \in R^8 \,|\, 0 \leqslant x_i \leqslant 1\,, \quad i = 1, 2, \ldots, 8\} \tag{5.6}$$

and $\underline{x}_0 = (1/2, 1/2, \ldots, 1/2)^T \in R^8$, that is a Kuhn Tucker point. The global minimizers of f on Λ are obtained setting four components equal to 0.5 and the others equal to 1.

TEST 7 (Six-hump Camel-back function [6]). Let $\underline{x} \in R^2$, we have

$$\min_{\underline{x} \in \Lambda} \left(4 - 2.1x_1^2 + \frac{x_1^4}{3} \right) x_1^2 + x_1 x_2 + (4x_2^2 - 4)x_2^2 \tag{5.7}$$

where Λ is given by

$$\Lambda = \{\underline{x} \in R^2 \,|\, -3 \leqslant x_1 \leqslant 3,\ -2 \leqslant x_2 \leqslant 2\}\,. \tag{5.8}$$

The starting point is $\underline{x}_0 = (0, 0)^T$ where the gradient of f is equal to zero. The function f has four local minimizers in Λ and two global minimizers $\underline{x}^* = \pm(-0.089842, 0.71266)^T$.

We note that the objective functions of the considered test problems do not satisfy the assumption of Theorem 2.3, but numerical experience shows that also in this case A1-algorithm works.

We studied the behavior of the two algorithms with respect to the input parameters. In fact an unsuitable choice of such parameters effects the success of the algorithms. To show the performance of the two algorithms we run the algorithms choosing the parameters involved to be the 'best' input parameters for the test problems considered. These 'best' parameters have been selected by several numerical experiments and they are shown in Table 5.1. With these parameters we find a global minimizer of the test problems considered almost at each running and, at the same time, we take as low as possible the computational cost of the algorithms.

In all the experiments for each value of the parameter $\varepsilon_j > \varepsilon_{\min}$ we followed a given number, NT, of trajectories for 100 iterations, then we ordered the trajectories according with the value that the objective function assumes on them and we discarded the trajectory numbered NT/2. In all the experiments we chose $\varepsilon_{\min} = 1/E - 2$. In Table 5.1 we report the other input parameters of the algorithms used for each test problem. For the A2-algorithm we had to choose three further parameters: p, q and β (see Section 4). For all the problems we used the parameters $p = 2.5$ and $q = 4$. In Table 5.1 we show the initial value of the parameter β that has been reduced at the end of each period of observation in such a manner to keep feasible each trajectory.

For each test problem we run A1 and A2 algorithms for 100 times. In Table 5.1 we show the number of successes of the two algorithms, that is how many times

Table 5.1. A1 and A2 algorithms

Test problems	ε_0	r_ε	HMAX	β	NT	Number of gradient evaluation	Successes of A1 algorithm	Successes of A2 algorithm
Test 1	100	0.7	1E-1	0.5	3	2600	99	100
Test 2	10	0.8	1E-4	1	4	3100	94	95
Test 3	100	0.8	1E-3	1	4	4200	98	98
Test 4	100	0.8	1E-3	1	6	4200	96	95
Test 5	100	0.8	1E-3	1	8	4200	92	90
Test 6	10	0.9	1E-1	0.5	10	6600	100	100
Test 7	1	0.7	1E-3	0.5	3	1300	100	100

ε_0: starting value of ε.
r_ε: reduction coefficient of ε.
HMAX: biggest value of the step-length h_k.
NT: number of trajectories followed.

they reached a global minimizer. Moreover in Table 5.1 we show the number of gradient evaluations required for each trajectory before the last step of the algorithm is called.

From Table 5.1 we can see that the A1 and A2 algorithms have substantially the same behavior. This fact is trivial when the global minimizer is in the interior of the feasible region (see Tests 1–5 and 7) but it is not obvious when the global minimizer is on the boundary as in Test 6. In this case to obtain a good performance of A1 algorithm we have to choose the step-length h_k (see (3.1)) in such a manner that the trajectory can reach the boundary only for a suitable small value of ε.

We note that the two algorithms require a large number of steps, so that, when the problem has a large number of variables, it is convenient to implement it in parallel.

In the last part of this section we compare the behavior of A1-algorithm (on test problems 1–7) with the method in [1, 2] (that we call A3-algorithm) and with the Simulated Annealing Algorithm that is a well-known stochastic algorithm for global optimization. For numerical implementation of the Simulated Annealing we refer to the Web sites:

http://weber.u.washington.edu/~savic/SA.HTM ,

http://www.bit.uq.edu.au/Francis/sim_ann.html .

In particular, at each iteration we generate a random direction whose component are numbers uniformly distributed in $[-1, 1]$ and we generate a random point uniformly along such direction. Then we update the new point with acceptance probability.

We made a numerical experience to establish the best input parameters of each algorithm on each test problem. We use these parameters to compare the performance of these algorithms. The parameters are shown in Tables 5.1, 5.2, 5.3.

With the parameters described, we run each algorithm for each test problem for 100 times with the following stop criterion:

$$(|f(\underline{x}^k) - f(\underline{x}^*)| < \textit{precision})\ \textit{OR}\ (\textit{number of iterations} > \textit{BigM}) \tag{5.9}$$

where $\underline{x}^*$ is the known optimal points of the problem considered and *BigM* is a very

Table 5.2. Parameters used for Simulated Annealing algorithm

Test problems	ε_0	r_ε	NP	Initial penality factor	r_p
Test 1	1E-5	0.9	100	100	1.1
Test 2	1	0.9	100	100	1.1
Test 3	1	0.95	100	100	1.1
Test 4	1	0.95	100	100	1.1
Test 5	1	0.98	100	100	1.1
Test 6	1	0.98	1000	1000	1.05
Test 7	1E-5	0.9	100	100	1.1

r_p: growth coefficient of penality factor.
NP: number of iterations between two consecutive updating of ε and penality factor.

Table 5.3. Parameters used for A3 algorithm

Test problems	ε_0	r_ε	NP	Initial penality factor	r_p	NT	h_k
Test 1	1	0.7	100	100	1.1	3	1E-3
Test 2	10	0.8	100	100	1.1	4	1E-3
Test 3	10	0.9	100	100	1.1	4	1E-3
Test 4	10	0.95	100	100	1.1	6	1E-3
Test 5	1	0.8	200	100	1.1	8	1E-2
Test 6	10	0.8	300	1000	1.05	10	1E-3
Test 7	1	0.7	100	100	1.1	3	1E-3

r_p: growth coefficient of penality factor.
NP: number of iterations between two consecutive updating of ε and penality factor.

large fixed constant. We chose *precision = 1E-5* for Test problems 1–5 and 7 and *precision = 1E-2* for Test problem 6. The different level of precision required for test problem 6 is due to the fact that this problem is not easy because the global and the local minimiers are 'nears' and located on the boundary of the feasible region. This fact makes poor the accuracy of the three algorithms, in particular the accuracy of the Simulated Annealing algorithm.

We note that the stop criterion (5.9) is different from the criterion described in Section 3, we use (5.9) since it is well suited to compare the performance of the algorithms.

The first aim of this study is to compare the number of successes in percent, that is how many times they reach a global minimizer on 100 running.

The second aim is to compare the average number of iterations required by the algorithms to satisfy (5.9) when a global optimum is found. The comparison is made keeping in mind that one iteration of the Simulated Annealing algorithm requires just one evaluation of function instead one iteration of the other two algorithms requires one evaluation of gradient for each trajectory considered.

The results of this comparison are shown in Table 5.4 where for each algorithm considered we report the number of successes (%), the average number of iterations

Table 5.4. Comparison between the algorithms considered

Test problems	A1-algorithm			Simulated Annealing		A3-algorithm		
	%	no. iter.	NT	%	no. iter.	%	no. iter.	NT
Test 1	99	2388	3	100	2753	100	2753	3
Test 2	95	3104	4	100	33437	88	2700	4
Test 3	98	4671	4	95	80775	97	6500	4
Test 4	96	4753	6	79	87053	70	7677	6
Test 5	92	4923	8	63	141212	65	14660	8
Test 6	100	5900	10	40	856655	95	6823	10
Test 7	100	931	3	100	1674	100	1515	3

(no. iter.) required to satisfy (5.9) when a global optimum is found and the number of trajectories followed for the A1 and A3 algorithms (NT).

We can see that in the case of Test problems 1–3 and 7 with low number of variables the behavior of the Simulated Annealing algorithm is better than the behavior of A1-algorithm. Instead, when the number of variables increases (Tests 4–6) and in particular when the value of the objective function f at the global minimizer is 'near' to the value of f at a local minimizer (Test 6) the performance of Simulated Annealing algorithm makes very poor.

We note that the behavior of A1 algorithm seems to be a little better than the behavior of A3 algorithm, moreover A1-algorithm has the advantage that it does not require penalization function and it guarantees (see Lemma 2.1) that $\underline{x}^k \in \overset{0}{\Lambda}$, $k = 1, 2, \ldots$.

References

1. Aluffi-Pentini, F., Parisi, V. and Zirilli, F. (1985), Global Optimization and Stochastic Differential Equations, *J. Optim. Theory Appl.* 47: 1–16.
2. Aluffi-Pentini, F., Parisi, V. and Zirilli, F. (1988), A Global Optimization Algorithm Using Stochastic Differential Equations, *ACM Trans. Math. Software* 14: 345–365.
3. Chiang, T.S., Hwang, C.R. and Shen, S.J. (1987), Diffusion for Global Optimization in R^n, *SIAM J. Control Optim.* 25: 737–753.
4. Hasminski, R.Z. (1980), *Stochastic Stability of Differential Equations*. Sijthoff & Noordhoff Intnl., Alpen aan den Rijn.
5. Hwang, C.R. and Shen, S.J. (1990), Large-Time Behavior of Perturbed Diffusion Markov Processes with Applications to the Second Eigenvalue Problem for Fokker Planck Operators and Simulated Annealing, *Acta Appl. Math.* 19: 253–295.
6. Levy, A.V. and Montalvo, A. (1985), The tunneling algorithm for the global minimization of functions, *SIAM J. Sci. Stat. Comput.* 6.
7. Schittkowski, K. (1987), *More Test Examples for Nonlinear Programming Codes*, Lecture Notes in Economics and Mathematical Systems 282, Springer-Verlag, Berlin, Germany.
8. Schuss, Z. (1980), *Theory and Applications of Stochastic Differential Equations*. John Wiley and Sons, New York.
9. Shoen, F. (1991), Stochastic Techniques for Global Optimization: a Survey of Recent Advances, *J. Global Optimization* 1: 207–228.