A conditional Gaussian martingale algorithm for global optimization

Manuel L. Esquível *

Departamento de Matemática and CMA, FCT/UNL, Quinta da Torre, 2829-516 Caparica, Portugal, mle@fct.unl.pt,
WWW home page: http://ferrari.dmat.fct.unl.pt/personal/mle/

Abstract. A new stochastic algorithm for determination of a global minimum of a real valued continuous function defined on K, a compact set of $\mathbb{R}^n$, having an unique global minimizer in K is introduced and studied, a context discussion is presented and implementations are used to compare the performance of the algorithm with other algorithms. The algorithm may be thought to belong to the *random search* class but although we use Gaussian distributions, the mean is changed at each step to be the intermediate minimum found at the preceding step and the standard deviations, on the diagonal of the covariance matrix, are halved from one step to the next. The convergence proof is simple relying on the fact that the sequence of intermediate random minima is an uniformly integrable conditional Gaussian martingale.

1 Introduction

Quite some attention has been recently devoted to stochastic algorithms, as more than 300 bibliographic entries in the reference textbook [9] testifies. Highly schematized global optimization methods using randomized search strategies are object of a thorough synthetic theoretical study in [12] which also presents applications of these methods to engineering problems. Negative results as those in [10] show that overconfidence on the effectiveness of stochastic methods is not desirable but, nevertheless, it is natural to speculate that an adequate randomized algorithm may perform better than a deterministic one in global optimization, at least in most of the situations. Theoretical results such as those in [2], [11] and [7], indicate that stochastic algorithms may be thought to be as reliable as deterministic ones and efforts in order to find better performing algorithms continue to be pursued as in [1] and [6]. The main feature of the new algorithm presented here, allows to recover some interesting properties of other stochastic algorithms such as the clustering and adaptiveness properties simultaneously with the property of continuing to search the whole domain at each step, which is a characteristic feature of simulated annealing.

* This work was done with partial support of FCT (Fundação para a Ciência e Tecnologia) program POCTI (Portugal/FEDER-EU). I hereby express my gratitude to Ana Luísa Custódio for enlightening discussions and suggestions and to the Mathematics Department for allowing an intensive use of one of its computer laboratories.

2 The Solis and Wets approach to random search

We recall next the powerful meta-approach of Solis and Wets (see [8]) in order to generalize its formulation to the case of adaptive random search and almost sure convergence. The original formulation of the authors solves the problem for non adaptive random search and convergence in probability, as noted in the remark 1 ahead. According to [2, p. 22], with the exception of [8] there were no synthesis studies of random search algorithms prior to 1986. Consider $f : K \subseteq \mathbb{R}^n \longrightarrow \mathbb{R}$, K a Borel set, where we suppose that for $x \in K^c$ we have $f(x) = +\infty$ and, $(\Omega, \mathcal{A}, \mathbb{P})$ a complete probability space. The following general selection scheme is the nuclear part of the random algorithm. Let $\psi : K \times \mathbb{R}^n \longrightarrow K$ be such that the following hypothesis $[H1]$ is verified.

$$\begin{cases} \forall x, t & f(\psi(t,x)) \leq f(t) \\ \forall x \in K & f(\psi(t,x)) \leq f(x) \ . \end{cases} \tag{2.1}$$

The general conceptual algorithm of Solis and Wets is as follows.

S. 0: Take $t_0 \in K$ and set $j = 0$.
S. 1: Generate a point x_j from $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n), \mathbb{P}_j)$.
S. 2: Set $t_{j+1} = \psi(t_j, x_j)$ choose $\mathbb{P}_{j+1}$, set $j = j + 1$ and return to step 1 (S.1).

Observe that in adaptive methods, x_j is a point with distribution $\mathbb{P}_j$ which depends on $t_{j-1}, t_{j-2}, \ldots, t_0$, thus being a conditional distribution. Let now λ denote the Lebesgue measure over $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$ and α be the essential infimum of f over K, that is: $\alpha := \inf\{t \in \mathbb{R} : \lambda(\{x \in K : f(x) < t\}) > 0\}$. Consideration of the essential infimum is mandatory to deal correctly with non continuous or unbounded functions such as $\mathbb{1}_{[0,1]\setminus\{1/2\}}$ defined in $[0, 1]$, or $\ln(|x|)\mathbb{1}_{\mathbb{R}\setminus\{0\}} + (-\infty)\mathbb{1}_{\{0\}}$. Let $E_{\alpha+\epsilon,M}$ denote the level set of f having level $\alpha + \epsilon$ defined by:

$$E_{\alpha+\epsilon,M} := \begin{cases} \{x \in K : f(x) < \alpha + \epsilon\} & \text{if } \alpha \in \mathbb{R} \\ \{x \in K : f(x) < M\} & \text{if } \alpha = -\infty \ . \end{cases} \tag{2.2}$$

A Solis and Wets's type convergence theorem may now be formulated and proved.

Theorem 1 *Suppose that f is bounded from below. Let the sequence of random variables $(T_j)_{j\geq 0}$ be defined inductively by using the sequence $(X_j)_{j\geq 0}$ which, in turn, depends on the family of probability laws $(\mathbb{P}_j)_{j\geq 0}$ given by the algorithm above and verifying: $T_0 = X_0$ such that $X_0 \frown \mathbb{P}_0$, $X_j \frown \mathbb{P}_j$ (to mean that X_j has $\mathbb{P}_j$ as law) and $T_{j+1} = \psi(T_j, X_j)$. If we have that the following hypothesis $[H2(\epsilon)]$ is verified: for some $\epsilon \geq 0$ and $M \in \mathbb{R}$*

$$\lim_{k\to+\infty} \inf_{0\leq j\leq k-1} \mathbb{P}_j[E^c_{\alpha+\epsilon,M}] = \lim_{k\to+\infty} \inf_{0\leq j\leq k-1} \mathbb{P}[X_j \in E^c_{\alpha+\epsilon,M}] = 0 \ , \tag{2.3}$$

then,

$$\lim_{k\to+\infty} \mathbb{P}[T_j \in E_{\alpha+\epsilon,M}] = 1 \ . \tag{2.4}$$

If, for all $\epsilon > 0$ $[H2(\epsilon)]$ is true, $(f(T_j))_{j\geq 0}$ converges almost surely to a random variable Y_* such that:

$$\mathbb{P}[Y_* \leq \alpha] = 1 \ . \tag{2.5}$$

Proof. Observe first that by hypothesis $[H1]$ in formula 2.1 we have that if $T_k \in E_{\alpha+\epsilon,M}$ or $X_k \in E_{\alpha+\epsilon,M}$ then for all $n \geq 0$, $T_{k+n} \in E_{\alpha+\epsilon,M}$. As a consequence,

$$\{T_k \in E^c_{\alpha+\epsilon,M}\} \subseteq \{T_1, T_2, \ldots, T_{k-1} \in E^c_{\alpha+\epsilon,M}\} \cap \{X_1, X_2, \ldots, X_{k-1} \in E^c_{\alpha+\epsilon,M}\} \; .$$

So, for all $j \in \{0, 1, \ldots, k-1\}$:

$$\mathbb{P}[T_k \in E^c_{\alpha+\epsilon,M}] \leq \mathbb{P}\left[\bigcap_{0 \leq l \leq k-1} \{T_l \in E^c_{\alpha+\epsilon,M}\} \cap \{X_l \in E^c_{\alpha+\epsilon,M}\}\right] \leq$$
$$\leq \mathbb{P}[X_j \in E^c_{\alpha+\epsilon,M}] = \mathbb{P}_j[E^c_{\alpha+\epsilon,M}] \; ,$$

which implies $\mathbb{P}[T_k \in E^c_{\alpha+\epsilon,M}] \leq \inf_{0 \leq j \leq k-1} \mathbb{P}_j[E^c_{\alpha+\epsilon,M}]$. We may now conclude that:

$$1 \geq \mathbb{P}[T_k \in E_{\alpha+\epsilon,M}] = 1 - \mathbb{P}[T_k \in E^c_{\alpha+\epsilon,M}] \geq 1 - \inf_{0 \leq j \leq k-1} \mathbb{P}_j[E^c_{\alpha+\epsilon,M}] \; ,$$

which, as a consequence of formula 2.3 implies the conclusion in formula 2.4. Define now the filtration $\mathbb{G} = (\mathcal{G}_j)_{j \geq 0}$ by $\mathcal{G}_j = \sigma(T_0, T_1, \ldots, T_j)$. It is then clear, from hypothesis $[H1]$ in formula 2.1, that:

$$\mathbb{E}[f(T_{j+1}) \mid \mathcal{G}_j] = \mathbb{E}[f(\psi(T_j, X_j) \mid \mathcal{G}_j] \leq \mathbb{E}[f(T_j) \mid \mathcal{G}_j] = f(T_j) \; ,$$

thus showing that $(f(T_j)_{j \geq 0}$ is a supermartingale bounded from below which we know to be almost surely convergent to some random variable which we will denote by Y_*. This conclusion together with formula 2.4 already proved shows that formula 2.5 yields, as a consequence of lemma 1.

Remark 1 *Hypothesis* $[H2]$ *given by formula 2.3 mean that the more the algorithm progresses in its steps, the more mass of the distributions* $\mathbb{P}_j$ *should be concentrated in the set* $E_{\alpha+\epsilon,M}$ *where the interesting points are. Our formulation of hypothesis* $[H2]$ *differs from the one presented in [8] which reads:*

$$\forall A \in \mathcal{B}(\mathbb{R}^n) \quad \lambda(A) = 0 \Rightarrow \prod_{j=0}^{+\infty} (1 - \mathbb{P}_j[A]) = 0 \; . \tag{2.6}$$

Formula 2.3 implies that, for $\epsilon > 0$*, we have* $\prod_{j=0}^{+\infty} (1 - \mathbb{P}_j[E_{\alpha+\epsilon,M}]) = 0$ *and so, our hypothesis is stronger then the one in [8]. The hypothesis given by formula 2.6 is more appealing as it does not use a condition on the set* $E_{\alpha+\epsilon,M}$ *which, in general, is not explicitly known and, in almost every case, will be difficult to use computationally. On the other hand, hypothesis given by formula 2.6 does not allow the conclusion of the Convergence Theorem (Global Search) in [8, p. 20] to hold in full generality. The theorem is true, with the proof presented there, if the sequence* $(X_j)_{j \geq 0}$ *is a sequence of independent random variables. The authors do not mention this* caveat *and the phrase* ... Nearly all random search methods are adaptive by which we mean that μ_k[1] depends on the quantities ... generated

[1] In our notation, the $\mathbb{P}_j$.

by the preceding iterations ... *may induce the reader in the opposite belief. In fact, the inequality on the right in the third line in the proof of the theorem (see [8, p. 21]) is, in the general case of dependent draws of the* $(X_j)_{j\geq 0}$*, the reversed one as a consequence of the elementary fact that if* $A \subset B$ *and* $0 < \mathbb{P}[B] < 1$ *then:* $\mathbb{P}[A \cap B] = \mathbb{P}[A] = \mathbb{P}[B] \cdot \mathbb{P}[A]/\mathbb{P}[B] \geq \mathbb{P}[A] \cdot \mathbb{P}[B]$.

If f is continuous over K, a compact subset of $\mathbb{R}^n$, then f attains its minimum and it is verified that $\alpha = \min_{x \in K} f(x)$. The conceptual algorithm above furnishes a way of determining this minimum. This is a simple consequence of the following result, for which the proof is easily seen.

Corolary 1 *Under the same hypothesis, if in addition for all* $x \in K$ *we have* $f(x) \geq \alpha$*, then:* $\mathbb{P}[Y_* = \alpha] = 1$.

For the reader's commodity, we state and prove the lemma used above.

Lemma 1 *Let* $(Z_j)_{j\geq 0}$ *be a sequence of random variables such that almost surely we have* $\lim_{j\to+\infty} Z_j = Z$ *and for some* $\delta > 0$ *we have* $\lim_{j\to+\infty} \mathbb{P}[Z_j < \delta] = 1$. *Then,* $\mathbb{P}[Z \leq \delta] = 1$.

Proof. It is a consequence of a simple observation following Fatou's lemma.

$$\mathbb{P}[Z > \delta] = \mathbb{P}[(\liminf_{j\to+\infty} Z_j) > \delta] = \mathbb{P}[\liminf_{j\to+\infty} \{Z_j > \delta\}] \leq \liminf_{j\to+\infty} \mathbb{P}[\{Z_j > \delta\}] \leq$$
$$\leq \limsup_{j\to+\infty} \mathbb{P}[\{Z_j \geq \delta\}] = \lim_{j\to+\infty} \mathbb{P}[\{Z_j \geq \delta\}] = 0 \, .$$

3 The conditional Gaussian martingale (CGM) algorithm

The algorithm presented here may be included, on a first approximation, in the class of random search methods as this class *consists of algorithms which generate a sequence of points in the feasible region following some prescribed probability distribution or sequence of probability distributions*, according to [3, p. 835]. The main idea of the method studied here is to change, at each new step, the location and dispersion parameters of the probability Gaussian distribution in order to concentrate the points, from which the new intermediate minimum will be selected, in the region where there is a greater chance of finding a global minimum not precluding, however, a new intermediate minimum to be found outside this region. We use a sequence of Gaussian distributions taking at each step the mean equal to the intermediate minimum found in the preceding step and the standard deviation diagonal elements of the covariance matrix equal to half the ones taken in the preceding step. We now briefly describe the algorithm and after we will present the almost sure convergence result. The goal is to find a global minimum of a real function f defined over a compact set $K \subset \mathbb{R}^n$ having diameter c. From now on, $\mathcal{U}(K)$ will denote the uniform distribution over K and $\mathcal{N}(m, \sigma)$ denotes the Gaussian distribution with mean m and covariance matrix with equal diagonal elements σ. With the presentation protocol of [5] the algorithm is as follows.

S. 0 Set $j = 0$;
S. 1 Generate $x_1^0, x_2^0, \ldots, x_N^0$ from the uniform distribution over the domain K.
S. 2 Choose $t_0 = x_{i_0}^0$ such that $f(x_{i_0}^0)$ is equal to $\min\{f(x_i^0) : 1 \leq i \leq N\}$. Increment j.
S. 3 Generate $x_1^j, x_2^j, \ldots, x_N^j$ from the normal distribution $\mathcal{N}(t_{j-1}, c/2^j)$ having mean t_{j-1} and diagonal covariance matrix elements $c/2^j$, c being the diameter of K.
S. 4 Choose $t_j = t_{i-1}$ if $f(t_{i-1})$ is strictly inferior to $\min\{f(x_i^j) : 1 \leq i \leq N\}$ and choose $t_j = x_{i_0}^j$ if $f(x_{i_0}^j)$ is less or equal to $\min\{f(x_i^j) : 1 \leq i \leq N\} \leq f(t_{i-1})$.
S. 5 Perform a stopping test and then: either stop, or increment j and return to step 3 (S. 3).

Observe that steps 1 and 2 are useful in order to choose a starting point for the algorithm in K. The repetition of steps 3, 4 and 5, provide a sort of clustering of the random test points around the intermediate minimizer found at the preceding step while continuing to search the whole space.

This algorithm's core is easily implemented in a programming language allowing symbolic computation (all implementations presented in this text are fully downloadable from the author's web page). For general purposes, we may extend f to the whole space by defining $f(x) = A$ (A large enough) for $x \notin K$.

```
Jota = 400; Uba = {}; Ruba = {};
For[j = 1, j ≤ Jota, j++,
 Tuba = {0}; Luba = {};
 minimo = Module[
   {ptminX = xm, ptminY = ym, ptmaxX = xM, ptmaxY = yM,
    cont = 850, cont1 = 5000, alea1, alea, T1, M1, Tes, eMes, NunOr},
   alea1 = Table[{Random[Real, {ptminX, ptmaxX}],
      Random[Real, {ptminY, ptmaxY}]}, {i, 1, cont1}];
   T1 = Table[{alea1[[i]], f[alea1[[i]][[1]], alea1[[i]][[2]]]}, {i, 1, cont1}];
   NunOr = Table[i, {i, 1, cont1}];
   M1 = Select[NunOr, Apply[f, Column[T1, 1][[#]]] <= Min[Column[T1, 2]] &];
   M1 = Min[M1];
   eMes = Flatten[Column[T1, 1][[M1]]]; (*Print[eMes]*);
   For[i = 1, And[i ≤ 52, Abs[Apply[f, {x0, y0}] - Apply[f, eMes]] > 10^(-10),
     Norm[{x0, y0} - eMes] > 10^(-10)], i++,
    alea = Table[Random[MultinormalDistribution[eMes, {{(ptmaxX - ptminX) / 2^(i - 2),
          0}, {0, (ptmaxY - ptminY) / 2^(i - 2)}}]], {j, 1, cont}];
    Tes = Table[{alea[[k]], f[alea[[k]][[1]], alea[[k]][[2]]]}, {k, 1, cont}];
    M2 = Select[Table[i, {i, 1, cont}],
      Apply[f, Column[Tes, 1][[#]]] <= Min[Column[Tes, 2]] &]; M2 = Min[M2];
    eMes = If[Apply[f, Flatten[Column[Tes, 1][[M2]]]] <= Apply[f, eMes],
      Flatten[Column[Tes, 1][[M2]]], eMes];
    Tuba = {Max[Append[Tuba, i]]};
    Luba = {Abs[Apply[f, eMes] - f[x0, y0]] / Abs[f[x0, y0]], Norm[eMes]};
   ]
  ]; Uba = Append[Uba, Tuba[[1]]]; Ruba = Append[Ruba, {Tuba[[1]], Luba}];
]
```

Fig. 3.1. The Mathematica implementation of the CGM algorithm

The algorithm introduced converges to a global minimizer under the hypothesis of continuity of the function f defined on a compact set. For notational purposes given three random variables X, Y, Z and $(a, b) \in \mathbb{R}^n \times \mathbb{R}^{n \times n}$ we will write $X \in \mathcal{N}(Y, Z)$ to mean that conditionally on $Y = a, Z = b$, $X \in \mathcal{N}(a, b)$, that is, X has Gaussian distribution with mean a and covariance matrix b.

Theorem 2 *Let $f : K \longrightarrow \mathbb{R}$ be a real valued continuous function defined over K, a compact set in $\mathbb{R}^n$, and let z be an unique global minimizer of f in K, that is: $f(z) = \min_{x \in K} f(x)$ and for all $x \in K$ we have $f(z) < f(x)$.*

For each $N \in \mathbb{N} \setminus \{0\}$ fixed, define almost surely and recursively the sequence $(T_j^N)_{j \in \mathbb{N}}$ by:

$$T_0^N = \left\{ X_{i_0}^0 : f(X_{i_0}^0) = \min_{1 \leq i \leq N} f(X_i^0) \;\; X_1^0, \ldots, X_N^0 \in \mathcal{U}(K) \;\; i.i.d. \right\} .$$

Next, for all $j \geq 1$

$$T_{j+1}^N := \begin{cases} T_j^N \;\; \text{if } f(T_j^N) < \min_{1 \leq i \leq N} \left\{ f(X_i^{j+1}) : \; X_i^{j+1} \in \mathcal{N}(T_j^N, \frac{c}{2^{j+1}}) \;\; i.i.d. \right\} \\ X_{i_0}^{j+1} \;\; \text{if } f(X_{i_0}^{j+1}) = \min_{1 \leq i \leq N} \left\{ f(X_i^{j+1}) : \; X_i^{j+1} \in \mathcal{N}(T_j^N, \frac{c}{2^{j+1}}) \;\; i.i.d. \right\} \\ \qquad\qquad \leq f(T_j^N) . \end{cases}$$

Then, for all $N \geq 1$ fixed, the sequence $(T_j^N)_{j \geq 0}$ is a uniformly integrable martingale which converges almost surely to a random variable T^N and the sequence $(T^N)_{N \geq 1}$ converges almost surely to z, the unique global minimizer of f.

Proof. For all $j \geq 1$ define the σ algebras $\mathcal{G}_j^N = \sigma(T_0^N, \ldots T_j^N)$ and the sets:

$$A_{j+1}^N = \left\{ f(T_j^N) < \min_{1 \leq i \leq N} \left\{ f(X_i^{j+1}) : \; X_i^{j+1} \in \mathcal{N}(T_j^N, \frac{c}{2^{j+1}}) \text{ i.i.d.} \right\} \right\} \subset \Omega .$$

As a first fact, we have obviously that $A_{j+1}^N \in \mathcal{G}_j^N$. Let us remark that $(T_j^N)_{j \geq 0}$ is a a martingale with respect to the filtration $(\mathcal{G}_j^N)_{j \geq 0}$. This is a consequence of:

$$\begin{aligned} \mathbb{E}[T_{j+1}^N \mid \mathcal{G}_j^N] &= \mathbb{E}[T_j^N \mathbb{1}_{A_{j+1}^N} + X_{i_0}^{j+1} \mathbb{1}_{(A_{j+1}^N)^c} \mid \mathcal{G}_j^N] = \\ &= T_j^N \mathbb{1}_{A_{j+1}^N} + \mathbb{1}_{(A_{j+1}^N)^c} \mathbb{E}[X_{i_0}^{j+1} \mid \mathcal{G}_j^N] = T_j^N \;\; , \end{aligned}$$

as we have, by the definitions,

$$\mathbb{E}[X_{i_0}^{j+1} \mid \mathcal{G}_j^N] = \mathbb{E}[X_{i_0}^{j+1} \mid T_0^N, \ldots, T_j^N] = \mathbb{E}[X_{i_0}^{j+1} \mid T_j^N] = T_j^N \;\; .$$

As a third fact, we notice that as $T_j^N \in K$, which is a compact set, we then have for some constant $M > 0$ that $\left\| T_j^N \right\|_1 \leq M$. As a consequence of these three facts $(T_j^N)_{j \geq 0}$ is a uniformly integrable martingale which converges almost surely to an integrable random variable T^N. Observe now that, by construction, $f(T_{j+1}^N) \leq f(T_j^N)$ almost surely and so we have that $(f(T_j^N))_{j \geq 0}$ decreases to $f(T^N)$. Let us remark that for all i, j we have $f(T^N) \leq f(X_i^j)$. In fact, by definition:

$$\begin{aligned} \min_{1 \leq i \leq N} f(X_i^{j_0+1}) \geq & \begin{cases} f(T_{j_0}^N) \text{ in } A_{j_0+1}^N \\ f(X_{i_0}^{j_0+1}) \text{ in } (A_{j_0+1}^N)^c \end{cases} = \\ & = f(T_{j_0+1}^N) \mathbb{1}_{A_{j_0+1}^N} + f(T_{j_0+1}^N) \mathbb{1}_{(A_{j_0+1}^N)^c} = f(T_{j_0+1}^N) \;\; , \end{aligned}$$

so, if for some i_0, j_0 we had

$$f(T^N) > f(X_{i_0}^{j_0+1}) = \min_{1\leq i\leq N} f(X_i^{j_0+1}) \geq f(T_{j_0+1}^N),$$

we would also have $f(T^N) > f(T_{j_0+1}^N)$, which contradicts the properties of $(f(T_j^N))_{j\geq 0}$. We will now show that the sequence $(f(T^N))_{n\geq 1}$ converges to $f(z)$ in probability. For that purpose, we recall the definition and some simple properties of $E_t := \{x \in K : f(x) < t\}$ the set of points of K having a level, given by f, less then t. First, the monotony: $t < s \Rightarrow E_t \subseteq E_s$; secondly, E_s is open: $x_0 \in E_t \Rightarrow \exists \delta > 0 \;\; B_{\mathbb{R}^n}(x_0, \delta) \subseteq E_t$; finally, $E_t \neq \emptyset \Rightarrow z \in E_t$. Observe now that for all $\omega \in \Omega$ and all $\eta > 0$:

$$\mid f(T^N(\omega)) - f(z) \mid > \eta \Leftrightarrow \begin{cases} f(T^N(\omega)) < f(z) - \eta \text{ which is impossible;} \\ f(T^N(\omega)) > f(z) + \eta \Rightarrow T^N(\omega) \notin E_{f(z)+\eta} \; . \end{cases}$$

As a consequence, for all i, j we have $X_i^j(\omega) \notin E_{f(z)+\eta}$, as otherwise we would have $f(X_i^j(\omega)) < f(z) + \eta < f(T^N(\omega)) \leq f(X_i^j(\omega))$, which is impossible. As a result, we finally have:

$$\left\{\mid f(T^N) - f(z) \mid > \eta\right\} \subseteq \bigcap_{j=0}^{+\infty} \bigcap_{i=0}^{N} \left\{X_i^j \notin E_{f(z)+\eta}\right\} \; ,$$

which implies that $\mathbb{P}\left[\mid f(T^N) - f(z) \mid > \eta\right]$ is bounded above, for instance, by:

$$\left(\inf_{0\leq j<+\infty} \mathbb{P}\left[X_i^j \notin E_{f(z)+\eta}\right]\right)^N \leq \mathbb{P}\left[X_i^0 \notin E_{f(z)+\eta}\right]^N \; .$$

Now, X_i^0 being uniformly distributed over K we have, with η small enough: $\mathbb{P}\left[X_i^0 \notin E_{f(z)+\eta}\right] = \lambda(E_{f(z)+\eta}^c)/\lambda(K) < 1$. So, we have as wanted, for all $\eta > 0$ small enough: $\lim_{n\to+\infty} \mathbb{P}\left[\mid f(T^N) - f(z) \mid > \eta\right] = 0$.

We now observe that the above convergence is, in fact, almost sure convergence, that is, $(f(T^N))_{N\geq 1}$ converges to $f(z)$ almost surely. This is a consequence of the well known fact that a non increasing sequence of random variables converging in probability, converges almost surely and the facts proved above that show:

$$\begin{array}{ccc} f(T_j^{N+1}) & \leq & f(T_j^N) \\ \downarrow_{j\to+\infty} & & \downarrow_{j\to+\infty} \\ f(T^{N+1}) & \leq & f(T^N) \; . \end{array}$$

Consider $\Omega^{'} \subset \Omega$ such that $\mathbb{P}[\Omega^{'}] = 1$ such that the above convergence takes place over $\Omega^{'}$ and observe that for every $\omega \in \Omega^{'}$ the sequence $(T^N(\omega))_{N\geq 1}$ is a sequence of points in the compact set K. As a consequence, every convergent subsequence $(T^{N_k}(\omega))_{N\geq 1}$ of $(T^N(\omega))_{N\geq 1}$ converges to z. In fact, if $\lim_{k\to+\infty} T^{N_k}(\omega) = y \in K$ then $\lim_{k\to+\infty} f(T^{N_k}(\omega)) = f(y)$ and by the result we just proved $\lim_{k\to+\infty} f(T^{N_k}(\omega)) = f(z)$. This implies that $f(y) = f(z)$ and

as z is a unique minimizer we have that $y = z$. We now conclude the proof of the theorem by noticing that $(T^N(\omega))_{N\geq 1}$ converges to z because if otherwise we would have: $\exists \epsilon > 0\ \forall N \exists N_m > N \quad \mid T^{N_m}(\omega) - z \mid > \epsilon$. As $(T^{N_m}(\omega))_{m\geq 1}$ is a sequence of points in K which is a compact set, by Bolzano Weierstrass theorem it has a convergent subsequence. This subsequence must converge to z which can not occur by the definition of $(T^{N_m}(\omega))_{m\geq 1}$.

4 Computational results

CGM algorithm was compared with other algorithms. With Styblinski-Tang function, we compared algorithms A (simple random search), B (localized random search), C (enhanced random search), from [9, p. 38–48], and ARS (accelerated random search) from [1]. The following notations are used. N is the

```
Jota = 400; Uva = {};
Ce = 2^0.5;
Rol = 10^(-4);
For[j = 1, j ≤ Jota, j++,
 XisEne = {Random[Real, {xm, xM}], Random[Real, {xm, xM}]};
 ErEne = 1;
 ind = 0;
 For[
  i = 1,
  And[i ≤ 25000, Abs[Apply[f, {x0, y0}] - Apply[f, XisEne]] > 10^(-10),
   Norm[{x0, y0} - XisEne] > 10^(-10)], i++,
  YupEne = {Random[Real, {XisEne[[1]] - ErEne * (xM - xm) / 2,
      XisEne[[2]] + ErEne * (xM - xm) / 2}], Random[Real,
     {XisEne[[1]] - ErEne * (xM - xm) / 2, XisEne[[1]] + ErEne * (xM - xm) / 2}]};
  If[Apply[f, YupEne] < Apply[f, XisEne], And[XisEne = YupEne, ErEne = 1],
   ErEne = ErEne / Ce];
  If[ErEne < Rol, ErEne = 1,];
  ind = i;
  Vaca = {ind, Abs[Apply[f, XisEne] - f[x0, y0]] / Abs[f[x0, y0]], Norm[XisEne]};
 ]; Uva = Append[Uva, Vaca];
]
```

Fig. 4.1. The Mathematica implementation of the ARS algorithm used.

number of random points drawn at each repetition; M will denote the number of repetitions in the simulation; J will be number of steps in the repetition; $\overline{J}$ represents the sample mean of the number of steps J necessary to achieve a prescribed result, taken over the whole set of repetitions of the simulation; $SD(J)$ is the sample standard deviation of J. The stopping criterion for the number of steps j in the simulation will be: $j \leq J_0 \wedge \mid f(z_n) - f(z) \mid > 10^{-10} \wedge \mid z_n - z \mid > 10^{-10}$, J_0 being the number of steps we decide to impose as an upper bound and z_n being the estimated minimizer at step n of a repetition. A true minimizer of the function is z. The function evaluation accuracy criterion after j steps is:

$$\Delta f(z_j) = \begin{cases} (f(z_j) - f(z))/f(z) & \text{if } f(z) \neq 0 \\ f(z_j) & \text{if } f(z) = 0\ . \end{cases}$$

The function argument evaluation accuracy criterion after j steps is:

$$\Delta z_j = \begin{cases} \|(z_j - z)/z\| & \text{if } z \neq 0 \\ \|z_j\| & \text{if } z = 0\ . \end{cases}$$

The averages or sample means over M repetitions with $j(k)$ steps at repetition k are: $\overline{\Delta z} = (1/M)\sum_{k=1}^{M} \Delta z_{j(k)}$, $\overline{z} = (1/M)\sum_{k=1}^{M} z_{j(k)}$, $\overline{\Delta f(z)} = (1/M)\sum_{k=1}^{M} \Delta f(z_{j(k)})$ and $\overline{f((z)} = (1/M)\sum_{k=1}^{M} f(z_{j(k)})$.

Table 1. Styblinski-Tang function; Repetitions: 400; ARS number of evaluations: 25000

Algorithm	$\overline{f(z)}$	SD($f(z)$)	$\overline{E}$	$SD(E)$
A	-78.2732	2.8×10^{-3}	25000	–
B	-78.3201	6.1581×10^{-4}	25000	–
C	-78.3201	5.9301×10^{-4}	25000	–
ARS (381)	-78.3323	1.35255×10^{-8}	24778	1818
CGM	-78.3323	2.72116×10^{-11}	15783	970

Table 2. CGM algorithm; random draws: 500; maximum number of steps: 50

Function	$\overline{J}$	SD(J)	$\overline{\Delta f(z)}$	$\overline{\Delta z}$
Gaussian 1 (371)	33.372	1.91627	$9.34019 - \times 10^{-12}$	8.96686×10^{-7}
Gaussian 2 (97)	37.0103	1.70474	7.38679×10^{-12}	2.49709×10^{-7}
Griewank	30.475	1.873	4.65529×10^{-11}	9.63914×10^{-6}
Himmelblau	32.1675	1.91012	4.21595×10^{-11}	1.09551
Jennrich-Sampson	49.7375	2.13541	4.90973×10^{-11}	8.77548×10^{-7}
Rastrigin (399)	32.3885	1.75567	2.29124×10^{-11}	4.93828×10^{-7}
Rosenbrock	30.5325	1.90868	4.30236×10^{-11}	6.17981×10^{-6}
Freudenstein-Roth	33.9725	1.83033	4.42997×10^{-11}	5.23501×10^{-7}
Styblinski-Tang	31.5675	1.93903	5.13851×10^{-13}	3.45016×10^{-7}

Table 3. ARS algorithm; Maximum number of function evaluations: 25000

Function	$\overline{N}$	SD(N)	$\overline{\Delta f(z)}$	$\overline{\Delta z}$
Gaussian 1	24984.8	303.95	5.34489×10^{-9}	2.03585×10^{-5}
Gaussian 2 (356)	25000	0	7.45836×10^{-8}	2.27671×10^{-5}
Griewank	25000	0	4.75247×10^{-8}	2.90102×10^{-4}
Himmelblau	25000	0	5.03447×10^{-1}	6.76862×10^{-1}
Jennrich-Sampson	24900.3	1426.31	1.00239×10^{-9}	1.24142×10^{-5}
Rastrigin	23991.4	3954.14	7.65517×10^{-10}	2.67892×10^{-6}
Rosenbrock (398)	24518.6	2543.41	2.89627×10^{-9}	4.59241×10^{-5}
Freudenstein-Roth	25000	0	4.80997×10^{-2}	1.63671×10^{-2}
Styblinski-Tang (382)	24777.9	1862.48	7.08203×10^{-11}	3.6433×10^{-6}

ARS and the CGM algorithms were compared for nine different test functions from [1] and [4]. The numerical results presented in the tables show that CGM outperforms all other algorithms tried, in precision and with less function evaluations. For one of the test functions (Gaussian 2) a further test, run with 2700 random draws at each step of the repetitions (instead of 500 used for the tables study) gave a result with the prescribed precision for all repetitions, in accordance with theorem 2. In the tables, the number between parenthesis in a given line report correct locations of the global minimum among the repetitions performed.

5 Conclusion

To achieve its goal, any random search algorithm has simultaneously to detect the region where the global minimum is located and to achieve enough precision in the calculation of the minimizer. Practically, and to the limits of machine and software precision used this is obtained, respectively, by an increasing number of random trials and, by concentrating these trials in the favorable region. CGM algorithm, hereby introduced and studied, always attained better precision than ARS; we got also perfect global minimum location for all the test functions tried, provided (in three cases) the number of random draws was sufficiently augmented. The CGM convergence result was proved under mild but fully verifiable hypothesis in sharp contrast with our formulation of a Solis and Wets's type theorem for adaptive random search with an hypothesis of difficult if not impossible verification even, in very simple cases.

References

1. Appel, M. J., Labarre, R., Radulović, D.: On accelerated random search. *SIAM J. Optim.*, **14**, (2003) no. 3, 708–731.
2. Guimier, A.: Modélisation d'algorithmes d'optimisation à stratégie aléatoire. *Calcolo.* **23** (1986) no. 1, 21–34.
3. Horst, R., Pardalos, P. M. (ed.): *Handbook of Global Optimization.* Kluwer Academic Publishers 1995.
4. Moré, J. J., Garbow, B. S., Hillstrom, K. E.:Testing Unconstrained Optimization Software, *ACM Transactions on Mathematical Software* **7**, No.1, (1981) 17–41.
5. Pintér, J. D.: *Global optimization in action. Continuous and Lipschitz optimization: algorithms, implementations and applications.*, Nonconvex Optimization and Its Applications. 6. Dordrecht: Kluwer Academic Publishers. xxvii 1996.
6. Raphael, B., Smith, I.F.C.: A direct stochastic algorithm for global search. *Appl. Math. Comput.* **146**, No.2-3, (2003) 729–758.
7. Shi, D., Peng, J.: A new theoretical framework for analyzing stochastic global optimization algorithms. *J. Shanghai Univ.* **3**, No.3, (1999) 175-180.
8. Solis, F. J., Wets, R. J.-B.: Minimization by random search techniques. *Math. Oper. Res.*, **6**, (1981) no. 1, 19–30.
9. Spall, J. C.: *Introduction to stochastic search and optimization. Estimation, simulation, and control.* Wiley-Interscience Series in Discrete Mathematics and Optimization. Hoboken, NJ: Wiley. xx, 2003.
10. Stephens, C.P., Baritompa, W.: Global optimization requires global information. *J. Optimization Theory Appl.* **96**, (1998) No.3, 575–588.
11. Yin, G.: Rates of convergence for a class of global stochastic optimization algorithms. *SIAM J. Optim.* **10**, No.10, (1999) 99–120 .
12. Zabinsky, Z. B.: *Stochastic adaptive search for global optimization.* Nonconvex Optimization and Its Applications 72. Boston, MA: Kluwer Academic Publishers. xviii, 2003.