A Merit Function Approach for Evolution Strategies

Youssef Diouane*

July 2, 2020

Abstract

In this paper, we extend a class of globally convergent evolution strategies to handle general constrained optimization problems. The proposed framework handles quantifiable relaxable constraints using a merit function approach combined with a specific restoration procedure. The unrelaxable constraints, when present, can be treated either by using the extreme barrier function or through a projection approach. Under reasonable assumptions, the introduced extension guarantees to the regarded class of evolution strategies global convergence properties for first order stationary constraints. Numerical experiments are carried out on a set of problems from the CUTEst collection as well as on known global optimization problems.

Keywords: Constrained optimization; derivative-free optimization; evolution strategy; merit function; global convergence.

1 Introduction

In this paper, we are interested in constrained derivative-free optimization problems [3], i.e.,

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in \Omega = \Omega_{\text{qr}} \cap \Omega_{\text{ur}}, \end{array} \tag{1}$$

where the objective function f is assumed to be locally Lipschitz continuous. The feasible region $\Omega \subset \mathbb{R}^n$ of this problem includes two categories of constraints [32]. The first, denoted by Ω_{qr} and known as quantifiable relaxable (QR) constraints, or soft constraints, is allowed to be violated during the optimization process and may need to be satisfied only approximately or asymptotically. Such a set of constraints will be assumed, in the context of this paper, to be of the form:

$$\Omega_{\text{qr}} = \{x \in \mathbb{R}^n | \forall i \in \{1, \ldots, r\}, c_i(x) \leq 0\},$$

where the functions c_i are locally Lipschitz continuous. The second category of constraints, denoted by $\Omega_{\text{ur}} \subset \mathbb{R}^n$, pools all unrelaxable (UR) constraints (also known as hard constraints), for such constraints no violation is allowed and they require satisfaction during the entire optimization process.

*ISAE-SUPAERO, Université de Toulouse, 31055 Toulouse Cedex 4, France (youssef.diouane@isae-supaero.fr).

Evolution strategies (ES's) [37] are evolutionary algorithms designed for global optimization in a continuous space, and that lead to promising results on practical optimization problems [7, 38, 8]. In [16, 17], the authors dealt with a large class of ES's, where a certain number λ of points (called offspring) are randomly generated in each iteration, among which $\mu \leq \lambda$ of them (called parents) are selected. ES's have been growing rapidly in popularity and used for solving challenging optimization problems [24, 6].

In [17], the authors proposed a general globally convergent framework for unrelaxable constraints using two different approaches. The first relies on techniques inspired from directional direct-search methods [13, 28], where one uses an extreme barrier function to prevent unfeasible displacements together with the possible use of directions that conform to the local geometry of the feasible region. The second approach was based on enforcing all the generated sample points to be feasible, while using a projection mapping approach. Both proposed strategies were compared to some of the best available solvers for minimizing a function without derivatives. The numerical results confirmed the competitiveness of the two approaches in terms of efficiency as well as robustness. Motivated by the recent availability of massively parallel computing platforms, the authors in [15] proposed a highly parallel globally convergent ES (inspired by [17]) adapted to the full-waveform inversion setting. By combining model reduction and ES's in a parallel environment, the authors contributed solving realistic instances of the full-waveform inversion problem.

In the context of ES's, many algorithms have been proposed in the literature to adapt ES's to solve constrained optimization problems [10]. Coello [11] and Kramer [30] outlined a comprehensive survey of the most popular constraints handling methods currently used with ES's. Recently, the authors in [1] proposed an adaptation of a class of ES's to handle QR constraints by using an augmented Lagrangian framework. The proposed approach was showed to enjoy good local and invariant convergence properties. To the best of our knowledge, all the ES's proposed suffer from the lack of global convergence guarantees when applied to general constrained optimization problems.

In the context of deterministic derivative-free optimization (DFO), only few works looked at both kinds (relaxable and unrelaxable) of constraints separately. For instance, Audet and Dennis [5] outlined a globally convergent direct-search approach based on a progressive barrier, which combined an extreme barrier approach for unrelaxable constraints and non-dominance filters [19] to handle QR constraints. More recently, the authors in [2] extended the progressive barrier approach, developed in [5], to cover the setting of a derivative-free trust-region method. Within the framework of directional direct-search methods, Vicente and Gratton [22] proposed an alternative where one handles QR constraints by means of a merit function. Under the appropriate assumptions, the latter approach ensured global convergence by imposing a sufficient decrease condition on a merit function combining information from both objective function and constraint violation. Another two-phases derivative-free approach was proposed in [33] to specifically handle the case where finding a feasible point is easier than minimizing the objective function.

In this paper, inspired by the merit function approach for direct search methods [22], we propose to adapt a class of ES algorithms (as proposed in [17]) to handle both QR and unrelaxable constraints. The class of ES algorithms obtained relies essentially on a merit function (eventually with a restoration procedure) to decide and control the distribution of the offspring points. The merit function is a standard penalty-based function that has already been proposed in the context of ES [11]. The main advantage of the proposed approach is to ensure a form of global convergence. Namely, under reasonable assumptions, this paper presents the first globally convergent ES framework handling both QR and UR constraints.

The proposed convergence theory generalizes the ES framework in [17] by including QR constraints, all in the spirit of the proposed merit function for directional direct search methods [22]. The contribution of this paper is twofold. First, we propose an adaptation of the merit function approach algorithm to the ES setting, a detailed convergence theory of the proposed approach is given. Second, we provide a practical implementation and extensive tests on a set of problems from the CUTEst collection as well as on known global optimization problems. The performance of our proposed solver is compared to (a) the progressive barrier approach implemented in the NOMAD solver [31], (b) the directional direct search method as proposed in [22] and (c) an adaptation of a well known ES using an augmented Lagrangian approach to handle QR constraints [1].

The paper is organized as follows. The proposed merit function approach is given in Section 2 with a detailed description of the changes introduced in a class of ES algorithms in order to handle general constraints. The convergence results of the adapted approach are then detailed in Section 3. In Section 4, we test the proposed algorithm on a set of problems from the CUTEst collection as well as on known global optimization problems. Finally, we make some concluding remarks in Section 5.

2 A globally convergent ES for general constraints

This paper focuses on a class of ES's, denoted by $(\mu/\mu_W, \lambda)$-ES, which evolves a single candidate solution. In fact, at the $k-$th iteration, a new population $y^1_{k+1}, \ldots, y^\lambda_{k+1}$ (called offspring) is generated around a weighted mean x_k of the previous parents (candidate solution). The symbol "$/\mu_W$" in $(\mu/\mu_W, \lambda)$-ES specifies that μ parents are "recombined" into a weighted mean. The parents are selected as the μ best offspring of the previous iteration in terms of the objective function value. The mutation operator of the new offspring points is done by $y^i_{k+1} = x_k + \sigma^{\mathrm{ES}}_k d^i_k$, $i = 1, \ldots, \lambda$, where d^i_k is drawn from a certain distribution $\mathcal{C}_k$ and σ^{ES}_k is a chosen step size. The weights used to compute the means belong to the simplex set $S = \{(\omega^1, \ldots, \omega^\mu) \in \mathbb{R}^\mu : \sum_{i=1}^{\mu} w^i = 1, w^i \geq 0, i = 1, \ldots, \mu\}$. The $(\mu/\mu_W, \lambda)$-ES adapts the sampling distribution to the landscape of the objective function. An adaptation mechanism for the step size parameter is also possible. The latter increases or decreases depending on the landscape of the objective function. One relevant instance of such an ES is covariance matrix adaptation ES (CMA-ES) [25].

In [16, 17], the authors proposed a framework for making a class of ES's enjoying some global convergence properties while solving optimization problems possibly with UR constraints. In fact, in [16], by imposing a sufficient decreasing condition on the objective function value, the proposed algorithm monitored the step size σ_k to ensure its convergence to zero (which leads then to the existence of a stationary point). The imposed sufficient decreasing condition is applied directly to the weighted mean x^{trial}_{k+1} of the new parents. By sufficient decreasing condition we mean $f(x^{\mathrm{trial}}_{k+1}) \leq f(x_k) - \rho(\sigma_k)$, where $\rho(\cdot)$ is a forcing function [28], i.e., a positive, nondecreasing function satisfying $\rho(\sigma)/\sigma \to 0$ when $\sigma \to 0$. To handle UR constraints [17], one starts with a feasible iterate x_0 and then aviods stepping outside the feasible region by means of a barrier approach. In this context, the sufficient decrease condition is applied not to f but to the extreme barrier function $f_{\Omega_{\mathrm{ur}}}$ associated with f with respect to the constraints set Ω_{ur} [4] (also known as the death penalty function in the terminology of evolutionary algorithms), which is defined by:

$$f_{\Omega_{\mathrm{ur}}}(x) \;=\; \begin{cases} f(x) & \text{if } x \in \Omega_{\mathrm{ur}}, \\ +\infty & \text{otherwise.} \end{cases}$$

The extreme barrier function is formally introduced in [3]. The obtained ES approach is detailed in [17, Algorithm 2.1]. The global convergence of the algorithm is achieved by establishing that some type of directional derivatives are nonnegative at limit points of refining subsequences along certain limit directions (see [17, Theorem 2.1]).

The challenge of this paper consists in extending [17, Algorithm 2.1] to a globally convergent framework that takes into account both QR and UR constraints. The author acknowledges that a preliminary version of this work was produced during his PhD thesis [14, Chapter 5]. In what comes next, we define the merit function as follows:

$$M(x) \;=\; \begin{cases} f(x) + \bar{\delta} g(x) & \text{if } x \in \Omega_{\mathrm{ur}}, \\ +\infty & \text{otherwise.} \end{cases}$$

where $\bar{\delta} > 0$ is a given positive constant and g defines a constraint violation function with respect to QR constraints. The ℓ_1-norm is commonly used to define the constraint violation function, i.e.,

$$g(x) = \sum_{i=1}^{r} \max\{c_i(x), 0\}.$$

Other choices for g exist, for instance, using the ℓ_2-norm i.e., $g(x) = \sum_{i=1}^{r} \max\{c_i(x), 0\}^2$. We note that the same constraint violation function g is used within the progressive barrier approach [5], that was in turn inspired by the filter approach of Fletcher and Leyffer [19]. The merit function will be used to evaluate a trial step and hence decide whether such step will be accepted or not. The extension of the globally convergent ES to a general constrained setting can be seen as a combination of two approaches, a feasible one where either the extreme barrier or a projection operator will be used to handle the UR constraints, and a merit function approach (possibly with a restoration procedure) to handle QR constraints.

The description of the proposed framework is as follows. For a given iteration k, a trial mean parent x_{k+1}^{trial} is computed as the weighted mean of the μ best points in terms of the merit function value. The current trial mean parent will be considered as a "***Successful point***" if one of the two following situations occur. The first scenario arises when one is sufficiently away from the feasible region (i.e., $g(x_k) > C\rho(\sigma_k)$ for some constant $C > 1$) and x_{k+1}^{trial} sufficiently decreases the constraint violation function g (i.e., $g_{\Omega_{\mathrm{ur}}}(x_{k+1}^{\mathrm{trial}}) < g(x_k) - \rho(\sigma_k)$, where $g_{\Omega_{\mathrm{ur}}}$ denotes the extreme barrier function associated with g with respect to Ω_{ur}). The second situation occurs when the merit function is sufficiently decreased (i.e., $M(x_{k+1}^{\mathrm{trial}}) < M(x_k) - \rho(\sigma_k)$).

Before checking whether the trial point is successful or not, the algorithm will try first to restore the feasibility or at least decrease the constraint violation if needed. The restoration process will be activated if the current mean parent x_k is far away from the feasible region and the trial point x_{k+1}^{trial} sufficiently decreases the constraint violation function g but not the merit function. More specifically, a "***Restoration identifier***" will be activated if one has

$$g_{\Omega_{\mathrm{ur}}}(x_{k+1}^{\mathrm{trial}}) \;<\; g(x_k) - \rho(\sigma_k) \quad \text{and} \quad g(x_k) > C\rho(\sigma_k)$$

and

$$M(x_{k+1}^{\mathrm{trial}}) \;\geq\; M(x_k).$$

The restoration algorithm will be left as far as progress on the reduction of the constraint violation can not be achieved all without any considerable increase in f. The complete description of the restoration procedure is given in Algorithm 2.

As a result, the main iteration of the proposed merit function approach can be divided into two steps: restoration and minimization. In the restoration step the aim is to decrease infeasibility (by minimizing essentially the function $g_{\Omega_{\mathrm{ur}}}$) while in the minimization step the objective function f is improved over a relaxed set of constraints by using the merit function M. The final approach obtained is described is given in Algorithm 1.

For both algorithms (main and restoration), our global convergence analysis will be performed independently of the choice of the distribution $\mathcal{C}_k$, the weights $(\omega_k^1, \ldots, \omega_k^\mu) \in S$, and the step size σ_k^{ES}. Therefore, the update of the ES parameters is left unspecified at this stage. However, the distribution $\mathcal{C}_k$ will be very useful in ensuring that a central convergence assumption (related to the density of the directions in the unit sphere) can be seen as reasonable. In fact, by choosing the distribution $\mathcal{C}_k$ to be multivariate normal distribution with mean zero, one can guarantee the density of the directions with a probability one. We will give more details on that in the next section.

Note that we also impose bounds on all directions d_k^i used by the algorithm. This modification is, however, very mild since the lower bound $d_{\min}$ can be chosen very close to zero and the upper bound $d_{\max}$ set to a very large number. The construction of the set of directions $\{\tilde{d}_k^i\}$ can be done with respect to the local geometry of the UR constraints as proposed in [17, Section 2.2].

3 Global convergence

The convergence results presented in this section are in the vein of those first established for the merit function approach for direct search methods [22]. For the convergence analysis, we will consider a sequence of iterations generated by Algorithm 1 without any stopping criterion. The analysis is organized depending on the number of times restoration is entered.

3.1 Case 1: the restoration algorithm is never entered after a certain order

When the restoration is entered finite times, one can guarantee that a subsequence of the step sizes $\{\sigma_k\}$ will converge to zero. In fact, due to the sufficient decrease condition imposed on the merit function along the iterates (or in the constraints violation function if the iterates are sufficiently away from the feasible region) and the control on the step size (reduced at least by β_2 for unsuccessful iterations), one can ensure the existence of a subsequence K of unsuccessful iterates driving the step size to zero.

Lemma 3.1 *Let f be bounded below and assuming that the restoration is not entered after a certain order. Then,*

$$\liminf_{k \to +\infty} \sigma_k = 0.$$

Proof. Suppose that there exists a $\bar{k} > 0$ and $\sigma > 0$ such that $\sigma_k > \sigma$ and $k \geq \bar{k}$ is a given iteration of Algorithm 1. If there is an infinite sequence J_1 of successful iterations after $\bar{k}$, this leads to a contradiction with the fact that g and f are bounded below.

Algorithm 1: A globally convergent ES for general constraints (Main)

Data: choose positive integers λ and μ such that $\lambda \geq \mu$. Select an initial $x_0 \in \Omega_{\text{ur}}$ and evaluate $f(x_0)$. Choose initial step lengths $\sigma_0, \sigma_0^{\text{ES}} > 0$ and initial weights $(\omega_0^1, \ldots, \omega_0^\mu) \in S$. Choose constants $\beta_1, \beta_2, d_{\min}, d_{\max}$ such that $0 < \beta_1 \leq \beta_2 < 1$ and $0 < d_{\min} < d_{\max}$. Select a forcing function $\rho(\cdot)$

for $k = 0, 1, \ldots$ **do**

Step 1: compute new sample points $Y_{k+1} = \{y_{k+1}^1, \ldots, y_{k+1}^\lambda\}$ such that

$$y_{k+1}^i \;=\; x_k + \sigma_k \tilde{d}_k^i, \; i = 1, \ldots, \lambda,$$

where the directions $\tilde{d}_k^i$'s are computed from the original ES directions d_k^i's (which in turn are drawn from a chosen ES distribution $\mathcal{C}_k$ and scaled if necessary to satisfy $d_{\min} \leq \|d_k^i\| \leq d_{\max}$).;

Step 2: evaluate $M(y_{k+1}^i)$, $i = 1, \ldots, \lambda$, and reorder the offspring points in $Y_{k+1} = \{\tilde{y}_{k+1}^1, \ldots, \tilde{y}_{k+1}^\lambda\}$ by increasing order: $M(\tilde{y}_{k+1}^1) \leq \cdots \leq M(\tilde{y}_{k+1}^\lambda)$.

Select the new parents as the best μ offspring sample points $\{\tilde{y}_{k+1}^1, \ldots, \tilde{y}_{k+1}^\mu\}$, and compute their weighted mean

$$x_{k+1}^{\text{trial}} \;=\; \sum_{i=1}^{\mu} \omega_k^i \tilde{y}_{k+1}^i;$$

Step 3: if $x_{k+1}^{\text{trial}} \notin \Omega_{\text{ur}}$ **then**

the iteration is declared unsuccessful;

else

if x_{k+1}^{trial} *is a "**Restoration identifier**"* **then**

enter Restoration (with $k_r = k$);

else

if x_{k+1}^{trial} *is a "**Successful point**"* **then**

declare the iteration successful, set $x_{k+1} = x_{k+1}^{\text{trial}}$, and $\sigma_{k+1} \geq \sigma_k$ (for example $\sigma_{k+1} = \max\{\sigma_k, \sigma_k^{\text{ES}}\}$);

else

the iteration is declared unsuccessful;

end

end

end

if *the iteration is declared unsuccessful* **then**

set $x_{k+1} = x_k$ and $\sigma_{k+1} = \beta_k \sigma_k$, with $\beta_k \in (\beta_1, \beta_2)$;

end

Step 4: update the ES step length σ_{k+1}^{ES}, the distribution $\mathcal{C}_{k+1}$, and the weights $(\omega_{k+1}^1, \ldots, \omega_{k+1}^\mu) \in S$;

end

In fact, since ρ is a nondecreasing positive function, one has $\rho(\sigma_k) \geq \rho(\sigma) > 0$. Hence, if $g(x_{k+1}) < g(x_k) - \rho(\sigma_k)$ and $g(x_k) > C\rho(\sigma_k)$ for all $k \in J_1$, then

$$g(x_{k+1}) < g(x_k) - \rho(\sigma),$$

Algorithm 2: A globally convergent ES for general constraints (Restoration)

Data: Start from $x_{k_r} \in \Omega_{\mathrm{ur}}$ given from the Main algorithm and consider the same parameter as therein.

for $k = k_r, k_r + 1, k_r + 2, \ldots$ **do**

Step 1: compute new sample points $Y_{k+1} = \{y^1_{k+1}, \ldots, y^\lambda_{k+1}\}$ such that

$$y^i_{k+1} \;=\; x_k + \sigma_k \tilde{d}^i_k, \; i = 1, \ldots, \lambda,$$

where the directions $\tilde{d}^i_k$'s are computed from the original ES directions d^i_k's (which in turn are drawn from a chosen ES distribution $\mathcal{C}_k$ and scaled if necessary to satisfy $d_{\min} \le \|d^i_k\| \le d_{\max}$);

Step 2: evaluate $g_{\Omega_{\mathrm{ur}}}(y^i_{k+1})$, $i = 1, \ldots, \lambda$, and reorder the offspring points in $Y_{k+1} = \{\tilde{y}^1_{k+1}, \ldots, \tilde{y}^\lambda_{k+1}\}$ by increasing order: $g_{\Omega_{\mathrm{ur}}}(\tilde{y}^1_{k+1}) \le \cdots \le g_{\Omega_{\mathrm{ur}}}(\tilde{y}^\lambda_{k+1})$.
Select the new parents as the best μ offspring sample points $\{\tilde{y}^1_{k+1}, \ldots, \tilde{y}^\mu_{k+1}\}$, and compute their weighted mean

$$x^{\mathrm{trial}}_{k+1} \;=\; \sum_{i=1}^{\mu} \omega^i_k \tilde{y}^i_{k+1};$$

Step 3: if $x^{\mathrm{trial}}_{k+1} \notin \Omega_{\mathrm{ur}}$ **then**
  the iteration is declared unsuccessful;
else
  if $g(x^{\mathrm{trial}}_{k+1}) \;<\; g(x_k) - \rho(\sigma_k)$ *and* $g(x_k) > C\rho(\sigma_k)$ **then**
    the iteration is declared successful, set $x_{k+1} = x^{\mathrm{trial}}_{k+1}$, and $\sigma_{k+1} \ge \sigma_k$ (for example $\sigma_{k+1} = \max\{\sigma_k, \sigma^{\mathrm{ES}}_k\}$);
  else
    the iteration is declared unsuccessful;
  end
end
if *the iteration is declared unsuccessful* **then**
  if $M(x^{\mathrm{trial}}_{k+1}) \;<\; M(x_k)$ **then**
    leave Restoration and return to the Main algorithm (starting at a new $(k+1)-$th iteration using x_{k+1} and σ_{k+1});
  else
    set $x_{k+1} = x_k$ and $\sigma_{k+1} = \beta_k \sigma_k$, with $\beta_k \in (\beta_1, \beta_2)$;
  end
end
Step 4: update the ES step length $\sigma^{\mathrm{ES}}_{k+1}$, the distribution $\mathcal{C}_{k+1}$, and the weights $(\omega^1_{k+1}, \ldots, \omega^\mu_{k+1}) \in S$;

end

which obviously contradicts the boundness below of g by 0. Thus there must exist an infinite subsequence $J_2 \subseteq J_1$ of iterates for which $M(x_{k+1}) \;<\; M(x_k) - \rho(\sigma_k)$. Hence,

$$M(x_{k+1}) \;<\; M(x_k) - \rho(\sigma) \qquad \text{for all } k \in J_2.$$

Thus $M(x_k)$ tends to -∞ which is a contradiction, since both f and g are bounded below.

The proof is thus completed if there is an infinite number of successful iterations. However, if no more successful iterations occur after a certain order, then this also leads to a contradiction. The conclusion is that one must have a subsequence of iterations driving σ_k to zero. ■

Theorem 3.1 *Let f be bounded below and assuming that the restoration is not entered after a certain order.*

There exists a subsequence K of unsuccessful iterates for which $\lim_{k \in K} \sigma_k = 0$. Moreover, if the sequence $\{x_k\}$ is bounded, there exists an x_* and a refining subsequence K' such that $\lim_{k \in K} x_k = x_*$.

Proof. From Lemma 3.1, there must exist an infinite subsequence K of unsuccessful iterates for which σ_{k+1} goes to zero. In such a case we have $\sigma_k = (1/\beta_k)\sigma_{k+1}$, $\beta_k \in (\beta_1, \beta_2)$, and $\beta_1 > 0$, and thus $\sigma_k \to 0$, for $k \in K$, too.

The second part of the theorem is proved by extracting a convergent subsequence $K' \subset K$ for which x_k converges to x_*. ■

Global convergence will be achieved by establishing that some type of directional derivatives are nonnegative at limit points of refining subsequences along certain limit directions (known as refining directions). By refining subsequence [4], we mean a subsequence of unsuccessful iterates in the Main algorithm (see Algorithm 1) for which the step-size parameter converges to zero.

Assuming that h is Lipschitz continuous around the point $x_* \in \Omega_{\mathrm{ur}}$, it is possible to use the Clarke-Jahn generalized derivative along a direction d

$$h^\circ(x_*; d) \;=\; \limsup_{\substack{x \to x_*,\, x \in \Omega_{\mathrm{ur}} \\ t \downarrow 0,\, x + td \in \Omega_{\mathrm{ur}}}} \frac{h(x+td) - h(x)}{t}.$$

The latter derivative, proposed by Jahn [27], can be seen as an adaptation of the Clarke generalized directional derivative [9] to the presence of constraints. We note that definition of $h^\circ(x_*; d)$ required that $x + td \in \Omega_{\mathrm{ur}}$ for $x \in \Omega_{\mathrm{ur}}$ arbitrarily close to x_* which can be guaranteed if d is hypertangent to Ω_{ur} at x_*. In what comes next, $B(x; \epsilon)$ will denote the closed ball formed by all points with a distance of no more than ϵ to x.

Definition 3.1 *A vector $d \in \mathbb{R}^n$ is said to be a hypertangent vector to the set $\Omega_{\mathrm{ur}} \subseteq \mathbb{R}^n$ at the point x in Ω_{ur} if there exists a scalar $\epsilon > 0$ such that*

$$y + tw \in \Omega_{\mathrm{ur}}, \quad \forall y \in \Omega_{\mathrm{ur}} \cap B(x; \epsilon), \quad w \in B(d; \epsilon), \quad \textit{and} \quad 0 < t < \epsilon.$$

The hypertangent cone to Ω_{ur} at x, denoted by $T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x)$, is the set of all hypertangent vectors to Ω_{ur} at x. Then, the Clarke tangent cone to Ω_{ur} at x (denoted by $T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x)$) can be defined as the closure of the hypertangent cone $T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x)$. The Clarke tangent cone generalizes the notion of tangent cone in Nonlinear Programming [36], and the original definition $d \in T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x)$ is given below.

Definition 3.2 *A vector $d \in \mathbb{R}^n$ is said to be a Clarke tangent vector to the set $\Omega_{\mathrm{ur}} \subseteq \mathbb{R}^n$ at the point x in the closure of Ω_{ur} if for every sequence $\{y_k\}$ of elements of Ω_{ur} that converges to x and for every sequence of positive real numbers $\{t_k\}$ converging to zero, there exists a sequence of vectors $\{w_k\}$ converging to d such that $y_k + t_k w_k \in \Omega_{\mathrm{ur}}$.*

For a direction v in the tangent cone, we consider the Clarke-Jahn generalized derivative to $\Omega_{\rm ur}$ at x_* as the limit

$$h^\circ(x_*;v) \;=\; \lim_{d\in T^{\rm H}_{\Omega_{\rm ur}}(x_*),d\to v} h^\circ(x_*;d)$$

(see [4]). A point $x_* \in \Omega_{\rm ur}$ is considered Clarke stationary if $h^\circ(x_*;d) \geq 0$, $\forall d \in T^{\rm CL}_{\Omega_{\rm ur}}(x_*)$.

An important ingredient used in our convergence analysis is the notion of refining direction [4], associated with a convergent refining subsequence K. A refining direction is defined as the limit point of $\{a_k/\|a_k\|\}$ for all $k \in K$ sufficiently large such that $x_k + \sigma_k a_k \in \Omega_{\rm ur}$, where $a_k = \sum_{i=1}^{\mu} \omega_k^i \tilde{d}_k^i$.

The following convergence result concerns the determination of feasibility.

Theorem 3.2 *Let $a_k = \sum_{i=1}^{\mu} \omega_k^i d_k^i$ and assume that f is bounded below. Suppose that the restoration is not entered after a certain order. Let $x_* \in \Omega_{\rm ur}$ be the limit point of a convergent subsequence of unsuccessful iterates $\{x_k\}_K$ for which $\lim_{k\in K}\sigma_k = 0$. Assume that g is Lipschitz continuous near x_* with constant $\nu_g > 0$.*

If $d \in T^{\rm H}_{\Omega_{\rm ur}}(x_*)$ is a refining direction associated with $\{a_k/\|a_k\|\}_K$, then either $g(x_*) = 0$ or $g^\circ(x_*;d) \geq 0$.

Proof. Let d be a limit point of $\{a_k/\|a_k\|\}_K$. Then, a subsequence K' of K must exist such that $a_k/\|a_k\| \to d$ on K'. On the other hand, we have for all k

$$x_{k+1}^{\rm trial} \;=\; \sum_{i=1}^{\mu} \omega_k^i \tilde{y}_{k+1}^i \;=\; x_k + \sigma_k \sum_{i=1}^{\mu} \omega_k^i d_k^i \;=\; x_k + \sigma_k a_k,$$

Since the iteration $k \in K'$ is unsuccessful, $g(x_{k+1}^{\rm trial}) \geq g(x_k) - \rho(\sigma_k)$ or $g(x_k) \leq C\rho(\sigma_k)$, and then either there exists an infinite number of the first inequality or the second one as follows:

1. For the case where there exists a subsequence $K_1 \subseteq K'$ such that $g(x_k) \leq C\rho(\sigma_k)$, it is trivial to obtain $g(x_*) = 0$ using both the continuity of g and the fact that σ_k tends to zero in K_1.

2. For the case where there exists a subsequence $K_2 \subseteq K'$ such that the sequence $\{a_k/\|a_k\|\}_{K_2}$ converges to $d \in T^{\rm H}_{\Omega_{\rm ur}}(x_*)$ in K_2 and the sequence $\{\|a_k\|\sigma_k\}_{k\in K_2}$ goes to zero in K_2 (a_k is bounded above for all k, and so $\sigma_k\|a_k\|$ tends to zero when σ_k does). Thus one must have necessarily for k sufficiently large in K_2, $x_k + \sigma_k a_k \in \Omega_{\rm ur}$ such that

$$g(x_k + \sigma_k a_k) \geq g(x_k) - \rho(\sigma_k).$$

From the definition of the Clarke-Jahn generalized derivative along directions $d \in T^{\rm H}_{\Omega_{\rm ur}}(x_*)$,

$$\begin{aligned} g^\circ(x_*;d) \;&=\; \limsup_{x\to x_*,t\downarrow 0,x+td\in\Omega_{\rm ur}} \frac{g(x+td)-g(x)}{t} \\ &\geq\; \limsup_{k\in K_2} \frac{g(x_k+\sigma_k\|a_k\|d)-g(x_k)}{\sigma_k\|a_k\|} \\ &=\; \limsup_{k\in K_2} \frac{g(x_k+\sigma_k\|a_k\|(a_k/\|a_k\|))-g(x_k)}{\sigma_k\|a_k\|} - g_k, \end{aligned}$$

where,

$$g_k = \frac{g(x_k + \sigma_k a_k) - g(x_k + \sigma_k \|a_k\| d)}{\sigma_k \|a_k\|}$$

from the Lipschitz continuity of g near x_*

$$\begin{aligned} g_k &= \frac{g(x_k + \sigma_k a_k) - g(x_k + \sigma_k \|a_k\| d)}{\sigma_k \|a_k\|} \\ &\leq \nu_g \left\| \frac{a_k}{\|a_k\|} - d \right\| \end{aligned}$$

tends to zero on K_2. Finally,

$$\begin{aligned} g^\circ(x_*; d) &\geq \limsup_{k \in K_2} \frac{g(x_k + \sigma_k a_k) - g(x_k) + \rho(\sigma_k)}{\sigma_k \|a_k\|} - \frac{\rho(\sigma_k)}{\sigma_k \|a_k\|} - g_k \\ &= \limsup_{k \in K_2} \frac{g(x_k + \sigma_k a_k) - g(x_k) + \rho(\sigma_k)}{\sigma_k \|a_k\|}. \end{aligned}$$

One then obtains $g^\circ(x_*; d) \geq 0$.

■

Moreover, assuming that the set of the refining directions $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$, associated with $\{a_k/\|a_k\|\}_K$, is dense in the unit sphere. One can show that the limit point x_* is Clarke stationary for the flowing optimization problem, known as the constraint violation problem:

$$\begin{aligned} \min \quad & g(x) \\ s.t. \quad & x \in \Omega_{\mathrm{ur}}. \end{aligned} \tag{2}$$

Theorem 3.3 *Let $a_k = \sum_{i=1}^{\mu} \omega_k^i d_k^i$ and assume that f is bounded below. Suppose that the restoration is not entered after a certain order. Assume that the directions $\tilde{d_k^i}$'s and the weights ω_k^i's are such that (i) $\sigma_k \|a_k\|$ tends to zero when σ_k does, and (ii) $\rho(\sigma_k)/(\sigma_k \|a_k\|)$ also tends to zero.*

Let $x_* \in \Omega_{\mathrm{ur}}$ be the limit point of a convergent subsequence of unsuccessful iterates $\{x_k\}_K$ for which $\lim_{k \in K} \sigma_k = 0$ and that $T^{\mathrm{CL}}_{\Omega}(x_*) \neq \emptyset$. Assume that g is Lipschitz continuous near x_* with constant $\nu > 0$

Then either (a) $g(x_*) = 0$ (implying $x_* \in \Omega_{\mathrm{qr}}$ and thus $x_* \in \Omega$) or (b) if the set of refining directions $d \in T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x_*)$ associated with $\{a_k/\|a_k\|\}_{K'}$ (where K' is a subsequence of K for which $g(x_k + \sigma_k a_k) \geq g(x_k) - \rho(\sigma_k)$) is dense in $T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x_*) \cap \{d \in \mathbb{R}^n : \|d\| = 1\}$, then $g^\circ(x_*; v) \geq 0$ for all $v \in T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x_*)$ and x_* is a Clarke stationary point of the constraint violation problem (2).

Proof. See the proof of [22, Theorem 4.2]. ■

We point out that the assumption regarding the directions $\{a_k/\|a_k\|\}_K$, in particular their density in the unit sphere, applies to a given refining subsequence K'' and not to the whole sequence of iterates. However, such a strengthening of the requirements on the density of the directions seems necessary for these types of directional methods [4]. By choosing the distribution

$\mathcal{C}_k$ in the algorithm to be a multivariate normal distribution with mean zero (the most commonly used choice in the literature), the density of the directions a_k in the unit sphere is guaranteed with a probability 1. In particular for such choice of $\mathcal{C}_k$, one has for any $y \in \mathbb{R}^n$ such that $\|y\| = 1$ and for any $\alpha \in (0, 1)$, there exists a positive constant η such that

$$\mathbb{P}\left(\cos(A_k/\|A_k\|, y) \geq 1 - \alpha, \|A_k\| \geq \epsilon\right) \geq \eta,$$

where A_k is a random variable whose realization is $a_k = \sum_{i=1}^{\mu} \omega_k^i \tilde{d}_k^i$. The justification of such a claim is discussed in further detail in [16].

We now move to an intermediate optimality result. As in [22], we will not use $x_* \in \Omega_{\text{qr}}$ explicitly in the proof but only $g^\circ(x_*; d) \leq 0$. The latter inequality describes the cone of first order linearized directions under feasibility assumption $x_* \in \Omega_{\text{qr}}$.

Theorem 3.4 *Let $a_k = \sum_{i=1}^{\mu} \omega_k^i d_k^i$ and assume that f is bounded below. Suppose that the restoration is not entered after a certain order.*

Let $x_* \in \Omega_{\text{ur}}$ be the limit point of a convergent subsequence of unsuccessful iterates $\{x_k\}_K$ for which $\lim_{k \in K} \sigma_k = 0$. Assume that g and f are Lipschitz continuous near x_*.

If $d \in T^{\text{H}}_{\Omega_{\text{ur}}}(x_*)$ is a refining direction associated with $\{a_k/\|a_k\|\}_K$ such that $g^\circ(x_*; d) \leq 0$. Then $f^\circ(x_*; d) \geq 0$.

Proof. By assumption there exists a subsequence $K' \subseteq K$ such that the sequence $\{a_k/\|a_k\|\}_{K'}$ converges to $d \in T^{\text{H}}_{\Omega_{\text{ur}}}(x_*)$ in K' and the sequence $\{\|a_k\|\sigma_k\}_{K'}$ goes to zero in K', Thus one must have necessarily for k sufficiently large in K', $x_{k+1}^{\text{trial}} = x_k + \sigma_k a_k \in \Omega_{\text{ur}}$.

Since the iteration $k \in K'$ is unsuccessful, one has $M(x_{k+1}^{\text{trial}}) \geq M(x_k) - \rho(\sigma_k)$, and thus

$$\frac{f(x_k + \sigma_k a_k) - f(x_k)}{\|a_k\|\sigma_k} \quad \geq \quad -\bar{\delta}\frac{g(x_k + \sigma_k a_k) - g(x_k)}{\|a_k\|\sigma_k} - \frac{\rho(\sigma_k)}{\sigma_k\|a_k\|} \tag{3}$$

On the other hand,

$$\begin{aligned} f^\circ(x_*; d) &= \limsup_{x \to x_*, t \downarrow 0, x + td \in \Omega} \frac{f(x + td) - f(x)}{t} \\ &\geq \limsup_{k \in K'} \frac{f(x_k + \sigma_k\|a_k\|d) - f(x_k)}{\sigma_k\|a_k\|} \\ &= \limsup_{k \in K'} \frac{f(x_k + \sigma_k\|a_k\|(a_k/\|a_k\|)) - f(x_k)}{\sigma_k\|a_k\|} - f_k, \end{aligned}$$

where,

$$f_k \quad = \quad \frac{f(x_k + \sigma_k a_k) - f(x_k + \sigma_k\|a_k\|d)}{\sigma_k\|a_k\|},$$

which then implies from (3)

$$\begin{aligned} f^\circ(x_*; d) &\geq \limsup_{k \in K'} \frac{f(x_k + \sigma_k\|a_k\|(a_k/\|a_k\|)) - f(x_k)}{\sigma_k\|a_k\|} - f_k, \\ &\geq \limsup_{k \in K'} -\bar{\delta}\frac{g(x_k + \sigma_k a_k) - g(x_k)}{\|a_k\|\sigma_k} - \frac{\rho(\sigma_k)}{\sigma_k\|a_k\|} - f_k \\ &\geq \limsup_{k \in K'} -\bar{\delta}\frac{g(x_k + \sigma_k\|a_k\|d) - g(x_k)}{\sigma_k\|a_k\|} + \bar{\delta}g_k - \frac{\rho(\sigma_k)}{\sigma_k\|a_k\|} - f_k, \end{aligned}$$

where

$$g_k \quad = \quad \frac{g(x_k + \sigma_k a_k) - g(x_k + \sigma_k \|a_k\| d)}{\sigma_k \|a_k\|}.$$

From the assumption $g^\circ(x_*; d) \le 0$, one has

$$\limsup_{k \in K'} \frac{g(x_k + \sigma_k \|a_k\| d) - g(x_k)}{\sigma_k \|a_k\|} \le \limsup_{x \to x_*, t \downarrow 0, x+td \in \Omega_{\mathrm{ur}}} \frac{g(x+td) - g(x)}{t} \le 0,$$

one obtains then

$$f^\circ(x_*; d) \quad \ge \quad \limsup_{k \in K'} \bar{\delta} g_k - \frac{\rho(\sigma_k)}{\sigma_k \|a_k\|} - f_k. \tag{4}$$

The Lipschitz continuity of both g and f near x_* guaranties that the quantities f_k and g_k tend to zero in K'. Thus, the proof is completed since the right-hand-side of (4) tends to zero in K'. ■

Finally, we derive the complete optimality result.

Theorem 3.5 *Assuming that f is bounded below and that Restoration is not entered after a certain order.*

Let $x_* \in \Omega_{\mathrm{ur}}$ be the limit point of a convergent subsequence of unsuccessful iterates $\{x_k\}_{k \in K}$ for which $\lim_{k \in K} \sigma_k = 0$. Assume that g and f are Lipschitz continuous near x_*.

Assume that the set

$$T(x_*) = T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*) \cap \{d \in \mathbb{R}^n : \|d\| = 1, g^\circ(x_*, d) \le 0\} \tag{5}$$

has a non-empty interior.

Let the set of refining directions be dense in $T(x_*)$. Then $f^\circ(x_*, v) \ge 0$ for all $v \in T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x_*)$ such that $g^\circ(x_*, v) \le 0$, and x_* is a Clarke stationary point of the problem (1).

Proof. See the proof of [22, Theorem 4.4]. ■

Now, we provide the analysis of the two other cases, namely when (a) an infinite run of consecutive steps inside Restoration or (b) one enters the restoration an infinite number of times.

3.2 Case 2: the restoration algorithm is entered and never left

In this case, by a refining subsequence below, we mean a subsequence of unsuccessful Restoration iterates for which the step-size parameter converges to zero.

Theorem 3.6 *Assume that f is bounded below and that the restoration is entered and never left.*

(i) Then there exists a refining subsequence.

(ii) Let $x_* \in \Omega_{\mathrm{ur}}$ be the limit point of a convergent subsequence of unsuccessful of iterates $\{x_k\}_K$ for which $\lim_{k \in K} \sigma_k = 0$. Assume that g is Lipschitz continuous near x_*, and let $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$ be a corresponding refining direction. Then either $g(x_*) = 0$ or $g^\circ(x_*; d) \ge 0$.

(iii) Let $x_* \in \Omega_{\mathrm{ur}}$ be the limit point of a convergent subsequence of unsuccessful of iterates $\{x_k\}_K$ for which $\lim_{k\in K} \sigma_k = 0$. Assume that g and f are Lipschitz continuous near x_*, and let $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$ be a corresponding refining direction such that $g^\circ(x_*; d) \le 0$. Then $f^\circ(x_*; d) \ge 0$.

(iv) Assume that the interior of the set $T(x_*)$ given in (5) is non-empty. Let the set of refining directions be dense in $T(x_*)$. Then $f^\circ(x_*, v) \ge 0$ for all $v \in T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x_*)$ such that $g^\circ(x_*, v) \le 0$, and x_* is a Clarke stationary point of the problem (1).

Proof. (*i*) There must exist a refining subsequence K within this call of the restoration, by applying the same argument of the case where one has $g(x_{k+1}) < g(x_k) - \rho(\sigma_k)$ and $g(x_k) > C\rho(\sigma_k)$ for an infinite subsequence of successful iterations (see the proof of Theorem 3.1). By assumption there exists a subsequence $K' \subseteq K$ such that the sequence $\{a_k/\|a_k\|\}_{k\in K'}$ converges to $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$ in K' and the sequence $\{\|a_k\|\sigma_k\}_{k\in K'}$ goes to zero in K'. Thus one must have necessarily for k sufficiently large in K', $x^{\mathrm{trial}}_{k+1} = x_k + \sigma_k a_k \in \Omega_{\mathrm{ur}}$.

(*ii*) Since the iteration $k \in K'$ is unsuccessful in the Restoration, $g(x_k + \sigma_k a_k) \ge g(x_k) - \rho(\sigma_k)$ or $g(x_{k+1}) \le C\rho(\sigma_k)$, and the proof follows an argument already seen (see the proof of Theorem 3.2).

(*iii*) Since at the unsuccessful iteration $k \in K'$, Restoration is never left, so one has $M(x_k + \sigma_k a_k) \ge M(x_k)$, and the proof follows an argument already seen (see the proof of Theorem 3.4).

(*iv*) The same proof as [22, Theorem 4.4]. ■

3.3 Case 2: the restoration algorithm is entered and left infinite times

Theorem 3.7 *Consider Algorithm 1 and assume that f is bounded below. Assume that Restoration is entered and left an infinite number of times.*

(i) Then there exists a refining subsequence.

(ii) Let $x_* \in \Omega_{\mathrm{ur}}$ be the limit point of a convergent subsequence of unsuccessful of iterates $\{x_k\}_K$ for which $\lim_{k\in K} \sigma_k = 0$. Assume that g is Lipschitz continuous near x_*, and let $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$ be a corresponding refining direction. Then either $g(x_*) = 0$ (implying $x_* \in \Omega_r$ and thus $x_* \in \Omega$) or $g^\circ(x_*; d) \ge 0$.

(iii) Let $x_* \in \Omega_{\mathrm{ur}}$ be the limit point of a convergent subsequence of unsuccessful of iterates $\{x_k\}_K$ for which $\lim_{k\in K} \sigma_k = 0$. Assume that g and f are Lipschitz continuous near x_*, and let $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$ be a corresponding refining direction such that $g^\circ(x_*; d) \le 0$. Then $f^\circ(x_*; d) \ge 0$.

(iv) Assume that the interior of the set $T(x_*)$ given in (5) is non-empty. Let the set of refining directions be dense in $T(x_*)$. Then $f^\circ(x_*, v) \ge 0$ for all $v \in T^{\mathrm{CL}}_{\Omega_{\mathrm{ur}}}(x_*)$ such that $g^\circ(x_*, v) \le 0$, and x_* is a Clarke stationary point.

Proof. (*i*) Let $K_1 \subseteq K$ and $K_2 \subseteq K$ be two subsequences where Restoration is entered and left respectively.

Since the iteration $k \in K_2$ is unsuccessful in the Restoration, one knows that the step size σ_k is reduced and never increased, one then obtains that σ_k tends to zero. By assumption there exists a subsequence $K' \subseteq K_2$ such that the sequence $\{a_k/\|a_k\|\}_{k\in K'}$ converges to $d \in T^{\mathrm{H}}_{\Omega_{\mathrm{ur}}}(x_*)$ in K_2 and the sequence $\{\|a_k\|\sigma_k\}_{k\in K'}$ goes to zero in K'.

(*ii*) For all $k \in K'$, one has $g(x_k + \sigma_k a_k) \ge g(x_k) - \rho(\sigma_k)$ or $g(x_k) \le C\rho(\sigma_k)$, one concludes that either $g(x_*) = 0$ or $g^\circ(x_*; d) \ge 0$.

(*iii*) For all $k \in K'$, one has $M(x_k + \sigma_k a_k) \ge M(x_k)$, and from this we conclude that $f^\circ(x_*; d) \ge 0$ if $g^\circ(x_*; d) \le 0$.

(iv) The same proof as [22, Theorem 4.4]. ■

To sum up, the analysis of the global convergence of Algorithm 1 was provided depending on the number of times the restoration procedure is entered. When the restoration is entered finite times, Theorem 3.2 showed that the limit points of certain subsequences of iterates are either feasible or Clarke stationary for the constraint violation problem (2). Theorem 3.5 showed then that such limit points are Clarke stationary for the optimization problem (1). Our analysis provide similar feasibility and optimality results for the two remaining cases (i.e., when the restoration is "entered but never left" or "entered and left an infinite number of times"), see Theorems 3.6 and 3.7.

4 Numerical experiments

In this section, we evaluate the performance of the proposed merit function approach using different solvers, different comparison procedures, and a large collection of non-linear constrained optimization problems. All the procedures were implemented in Matlab and run using Matlab 2019a on a MacBook Pro 2,4 GHz Intel Core i5, 4 GB RAM.

4.1 Problems tested and testing strategies

In what comes next, as a benchmark test, we will use 40 small-scale constrained test problems as given in [2] (those problems are extracted from the CUTEst collection [20]). The dimensions of the tested problems do not exceed 9 variables, with eventually bound constraints and no more than 13 nonlinear constraints (see [2, Table 1] for a detailed description on all the tested problems). For each test problem, the initial point provided by CUTEst is used, the latter respects the bound contraints but does not necessarily satisfy the nonlinear constraints.

To illustrate the obtained results, we will use the two well-known testing strategies: data profiles [35] and performance profiles [18]. For data profiles, we use the following convergence test

$$f^0_{\mathrm{max}} - f_\Omega(x) \;\geq\; (1-\alpha)(f^0_{\mathrm{max}} - f_{\mathrm{min}}),$$

while for the performance profiles, we make use of

$$f_\Omega(x) - f_{\mathrm{min}} \;\;\leq\;\; \alpha(f_{\mathrm{min}} + 1),$$

where α is the level accuracy and f^0_{max} represents the largest value among all the feasible objective function values initially visited by all the tested solvers (i.e., $f^0_{\mathrm{max}} = \max_s f^0_s$ where f^0_s represents the objective function value at the first feasible point visited by the solver s). The value f_{min} represents the best feasible solution found by the tested solvers. A tolerance of 10^{-7} for constraint violation is used to consider a point as being feasible. We note that, if a solver fails to find a feasible starting point for a given problem, the problem is considered as unsolved, in this case the convergence test is not used. The performance and data profiles are computed for a maximum of 3000 function evaluations. For the stochastic solvers, we will describe our results using the median data/performance profile obtained over 20 runs.

4.2 Implementation choices

Algorithm 1 and Algorithm 2 are implemented in Matlab. The obtained implementation will be called **ES-MF**. Most of the parameter choices followed those in [17] (where some of the user-specified parameters are the same used by directional direct search methods and CMA-ES). In particular, the values of λ and μ and of the initial weights are those of CMA-ES for unconstrained optimization (see [23]): $\lambda = 4 + \text{floor}(3\log(n))$, $\mu = \text{floor}(\lambda/2)$, where floor$(\cdot)$ rounds to the nearest integer, and $\omega_0^i = a_i/(a_1 + \cdots + a_\mu)$, $a_i = \log(\lambda/2 + 1/2) - \log(i)$, $i = 1, \ldots, \mu$. The choices of the distribution $\mathcal{C}_k$ and of the update of σ_k^{ES} also followed CMA-ES for unconstrained optimization. As used in most directional direct search implementations, the forcing function selected was $\rho(\sigma) = 10^{-4}\sigma^2$. To reduce the step length in unsuccessful iterations we used $\sigma_{k+1} = 0.9\sigma_k$ which corresponds to setting $\beta_1 = \beta_2 = 0.9$. For successful iterations we set $\sigma_{k+1} = \max\{\sigma_k, \sigma_k^{\text{CMA-ES}}\}$ (with $\sigma_k^{\text{CMA-ES}}$ the CMA step size used in ES). The directions d_k^i, $i = 1, \ldots, \lambda$, were scaled if necessary to obey the safeguards $d_{\min} \le \|d_k^i\| \le d_{\max}$, with $d_{\min} = 10^{-10}$ and $d_{\max} = 10^{10}$. The initial step size is estimated using only the bound constraints: If there is a pair of finite lower and upper bounds for a variable, then σ_0 is set to the half of the minimum of such distances, otherwise $\sigma_0 = 1$.

4.3 Sensitivity analysis

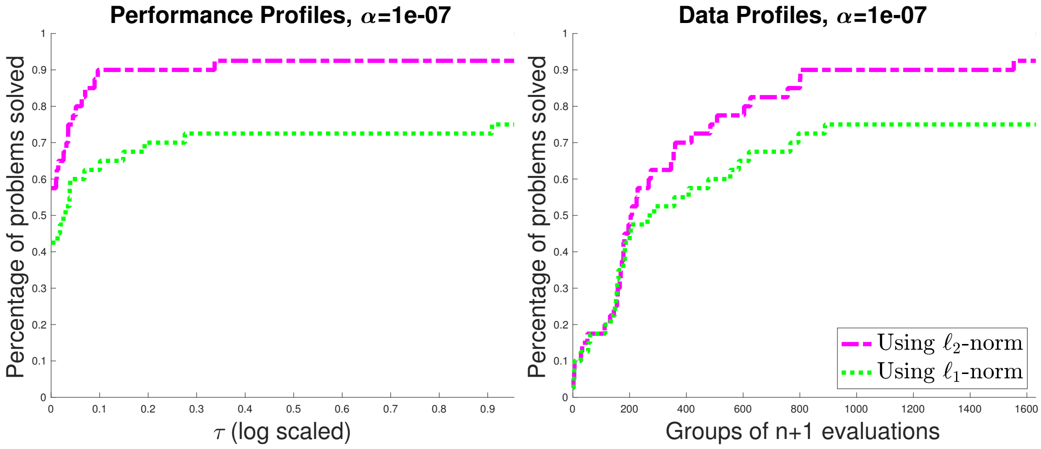
The proposed evolution strategy introduces some user-specified control parameters and their performances might depend on the setting of these parameters. A full sensitivity analysis of all the control parameters of the merit function approach can be computationally demanding and is beyond the scope of this paper. Hence, this subsection focuses on the sensitivity of **ES-MF** with respect to the newly introduced control parameters, namely, the constants $\bar{\delta}$ and C as well as the choice of norm type used to evaluate g.

Figure 1 shows their performance and the data profiles using different choices for the constants $\bar{\delta}$ and C as well as for the norm type used to evaluate the constraint violation function g. With respect to the choice the norm in g, see Figure 1(a), one can see that the use of ℓ_2-norm is clearly favorable to our approach in particular with a large budget of objective function evaluations. The choice of working with the ℓ_2-norm to evaluate g was shown to perform better for the progressive barrier approach used in MADS [5].

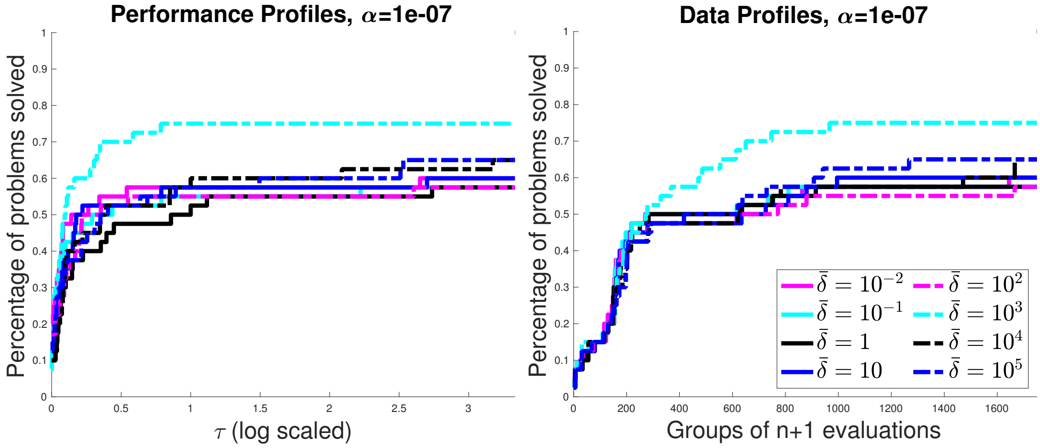
Regarding the $\bar{\delta}$ parameter, we tested 8 different values varied in range 10^{-2} and 10^5, see Figure 1(b). The obtained profiles show that, for a small budget of evaluations, **ES-MF** is not sensitive to the value of $\bar{\delta}$. For a larger budget, the performance changes slightly probably due to the stochastic nature of the solver. However, on the tested problems, one value of $\bar{\delta} = 10^3$ is shown to be very favorable to the **ES-MF** solver.

Next, for the parameter C, we tested 8 different values varied in range 10^{-2} and 10^5, see Figure 1(c). Again, the obtained profiles change slightly. We suspect that the slight changes in the performance are just due to the stochastic nature of the solver and consider that **ES-MF** is not very sensitive to the choice of the parameter C.

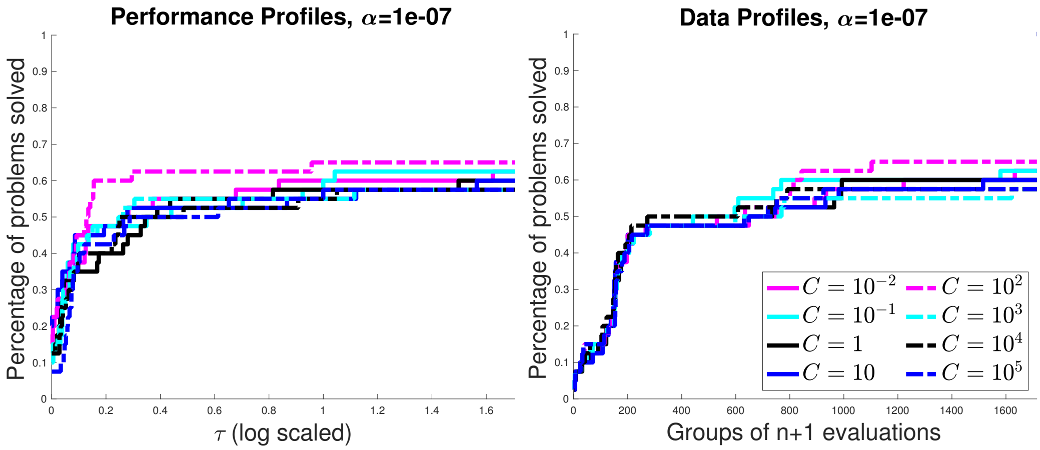
In what comes next, for the solver **ES-MF**, we set by default $\bar{\delta} = 1$, $C = 1$, and use the ℓ_2-norm to define the constraint violation function g.



(a) $\bar{\delta} = 1$ and $C = 1$



(b) ℓ_2-norm for g and $C = 1$.

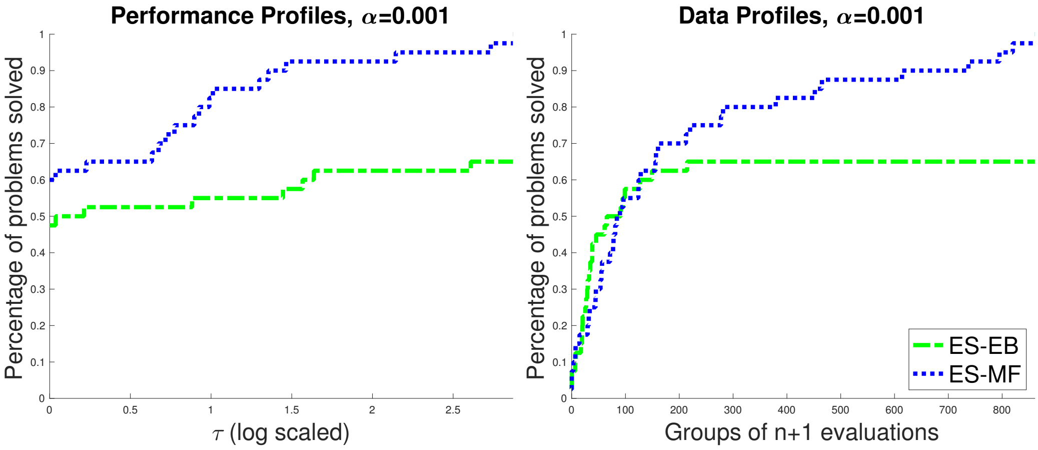


(c) ℓ_2-norm for g and $\bar{\delta} = 1$.

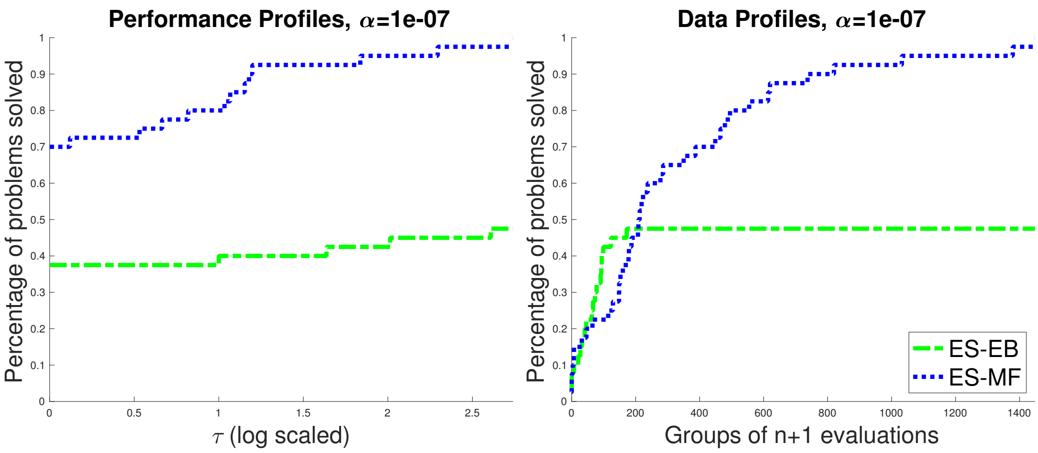
Figure 1: Median profiles for the solver **ES-MF** computed using 40 problems from the CUTEst set and different control parameters.

4.4 The extreme barrier versus the merit function for ES

In this subsection, we present a comparison between **ES-MF** and **ES-EB** from [17] (**ES-EB** can be seen as a particular instance of **ES-MF** where all the constraints are UR). Since the solver **ES-EB** requires a feasible starting point, when the starting point is infeasible, finding a feasible point is accomplished by minimizing the constraint violation function g.



(a) Considering the level of accuracy $\alpha = 10^{-3}$.



(b) Considering the level of accuracy $\alpha = 10^{-7}$.

Figure 2: Median profiles for the solvers **ES-MF** and **ES-EB** using 40 problems from the CUTEst set.

Figure 2 depicts the resulting performance and data profiles considering two levels of accuracy 10^{-3} and 10^{-7}. One can see that the extreme barrier approach is not able to solve more than 50% of the problems (as shown by the performance profiles). The data profiles indicate that the extreme barrier can be competitive for small budgets. Overall, the merit function approach is outperforming the extreme barrier approach. Thus, relaxing the constraints clearly makes it possible to reach better optimal solutions which motivates the use of the merit function approach **ES-MF** instead of **ES-EB**.

4.5 Comparison of solvers using the problems from the CUTEst collection

To quantify the efficiency of **ES-MF**, we include in our numerical comparison the solvers **MADS-PB**, **DDS-MF**, and **CSA-AL**:

- **MADS-PB** [5]: a mesh adaptive direct search (MADS) method where a progressive barrier (PB) approach has been implemented [5] to handle QR constraints. The progressive barrier approach, proposed in MADS, enjoys similar convergence properties as for our algorithm, hence, a comparison between the two solvers is very meaningful. For the MADS solver, we used the implementation given in the NOMAD package [31], version 3.9.1 (C++ version linked to Matlab via a mex interface). This solver is deterministic.
- **DDS-MF** [22]: a Matlab implementation of a directional direct search (DDS) method where a merit function (MF) is used to handle QR constraints. The parameter choices followed those given in the numerical section of [22]. We recall that **ES-MF** is inspired from the **DDS-MF** method, hence including the latter solver in the comparison can be also very meaningful. We note also that this is the first time **DDS-MF** is compared using an extensive test set. The behavior of the solver is stochastic as it generates randomly (at most) $n + 1$ directions at each iteration of the algorithm.
- **CSA-AL** [1]: a Matlab implementation of CMA-ES using an augmented Lagrangian approach to handle QR constraints. For the CMA-ES part, we used the same choice of parameters as for **ES-MF**, for the parameters associated with the augmented Lagrangian part we chose the values given in [1].

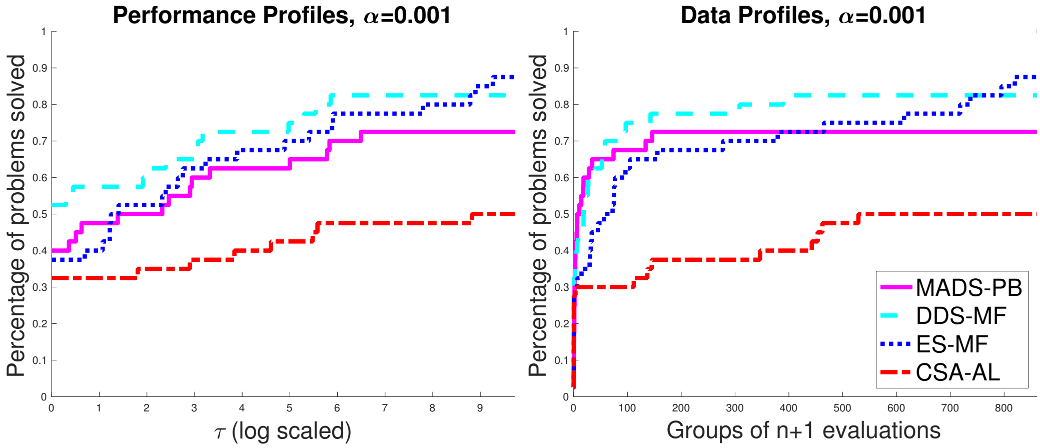
For all the solvers, we consider that all the nonlinear constraints are QR except the bounds which are treated using an ℓ_2-projection.

Figure 3 reports the median (out of 20 runs) profiles considering the two accuracy levels 10^{-3} and 10^{-7}. Clearly, for all the runs, **CSA-AL** is performing the worst among all the tested solvers. For the resulting data profiles, one can see that with a small budget, **DDS-MF** and **MADS-PB** exhibit better performance than the **ES-MF**. However, when the budget is getting larger, **ES-MF** performs the best. From the resulting performance profiles, one can see that in terms of efficiency (i.e., small values of τ), **DDS-MF** is shown to be best. The **ES-MF** solver performs better in terms of robustness (i.e., large values of τ).

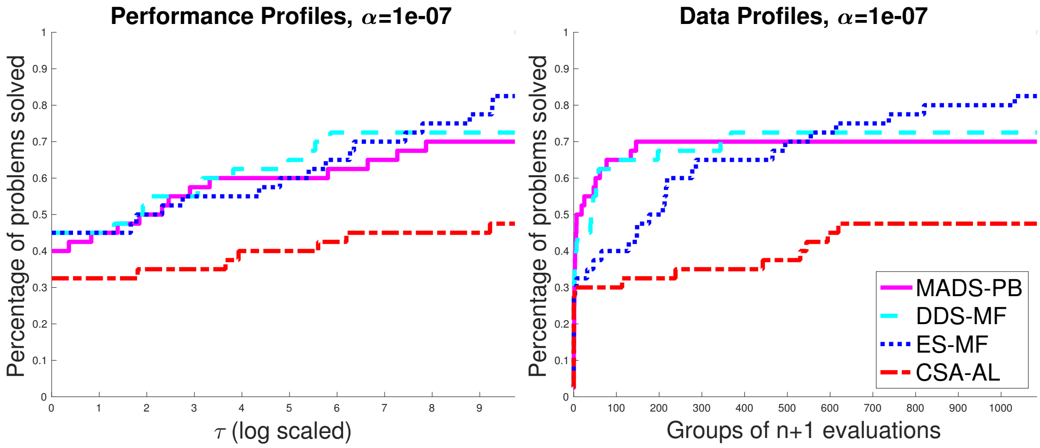
In conclusion, first, clearly the **ES-MF** solver leads to very good results compared to **CSA-AL**. In fact, in our tests, **CSA-AL** showed difficulties finding feasible points while making progress on the objective function. We stress that the main difference between the two evolution strategies is the restoration procedure, the latter helps **ES-MF** to progress better towards feasible zones without severe deterioration in terms of the objective function value. Second, **ES-MF** can be very competitive with both solvers **DDS-MF** and **MADS-PB**, in particular when using a large number of function evaluations.

4.6 Comparison of solvers using global optimization test problems

To confirm the results obtained when using CUTEst problems, we perform complementary tests using a set of problems with a diversity of features and the kind of difficulties that appear in constrained global optimization. The test set is that used in [26, 29, 34] and comprises 12 well-known test problems (see Table 1). The problems `G2`, `G3`, and `G8` are originally maximization problems and were converted to minimization.



(a) Considering the level of accuracy $\alpha = 10^{-3}$.



(b) Considering the level of accuracy $\alpha = 10^{-7}$.

Figure 3: Median profiles for the solvers **ES-MF**, **MADS-PB**, **DDS-MF**, and **CSA-AL**, using 40 problems from the CUTEst set.

In addition to such problems, we include three realistic problems. The first one is the tension-compression string (`TCS`) problem [12], the aim is to minimize the weight of a tension-compression string subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter and on design variables. The design variables are the mean coil diameter; the wire diameter and the number of active coils. The second problem is the well known welded beam design (`WBD`) problem [12] where a welded beam is designed with a minimum cost subject to constraints on shear stress; bending stress in the beam; buckling load on the bar; end deflection of the beam; and side constraints. The third optimization problem is a multidisciplinary design optimization (`MDO`) problem [39, 21] where a simplified wing design (built around a tube) is looked at. For this problem, one tries to minimize the range of the aircraft under coupled aero-structural constraints. The problem has 7 optimization variables corresponding to the geometry of the wing. The details of the three realistic problems features are included in Table 1.

To allow the analysis of the asymptotic efficiency and the robustness of the tested solvers, we generate performance and data profile using a larger maximal number of function evaluation

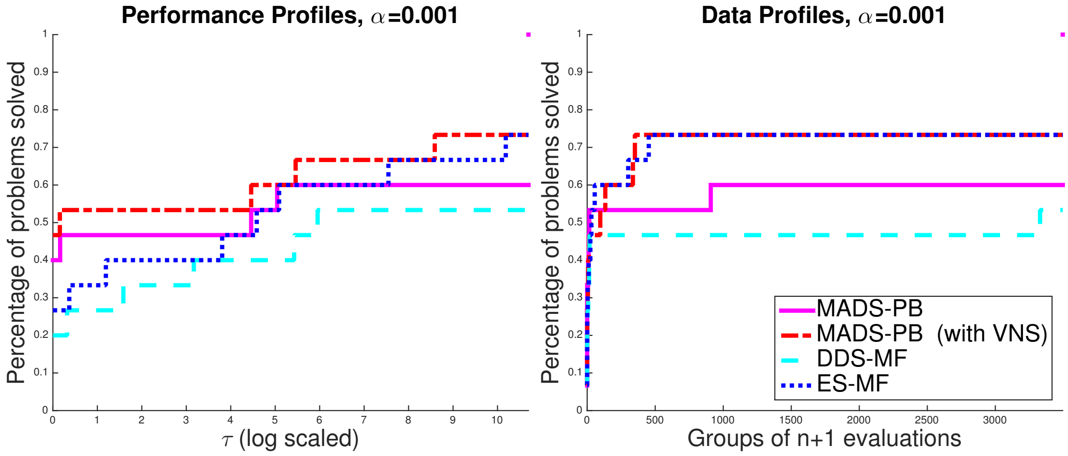
Problem	n	m	# UB	# LB	$f(x_0)$	$g(x_0)$	f_{opt}
G1	13	9	13	13	-228.028	93357.8	-15
G2	20	2	20	20	-0.0641952	0	-0.803619
G3	20	1	20	20	$-5.53267e-07$	0.582395	-1
G4	5	6	5	5	-24703.8	4.58618	-30665.5
G6	2	2	2	2	777287	$1.78677e+08$	-6961.81
G7	10	8	10	10	1154.69	410492	24.3062
G8	2	2	2	2	$-6.40052e-09$	4322.48	-0.095825
G9	7	4	7	7	156193	$3.67173e+06$	680.63
G10	8	6	8	8	20711.3	6.01742	7049.33
G11	2	1	2	2	4.97537	3.95049	0.75
G12	3	1	3	3	-0.532992	0	-1
G13	5	3	5	5	7.97186	71.9042	0.0539498
TCS	3	4	3	3	$3.51385e+07$	$2.15037e+10$	5868.76
WBD	4	6	4	4	278.59	1150.36	0.0126653
MDO	7	3	7	7	-10.6934	$2.3618e+07$	-16.61011

Table 1: Description of the features of the 15 global optimization problems: the dimension n, the number of the QR constraints m, the number of the lower bounds # LB, the number of the upper bounds # UB, the initial objective value $f(x_0)$, the initial constraints violation $g(x_0)$, and the best known feasible solution f_{opt}.

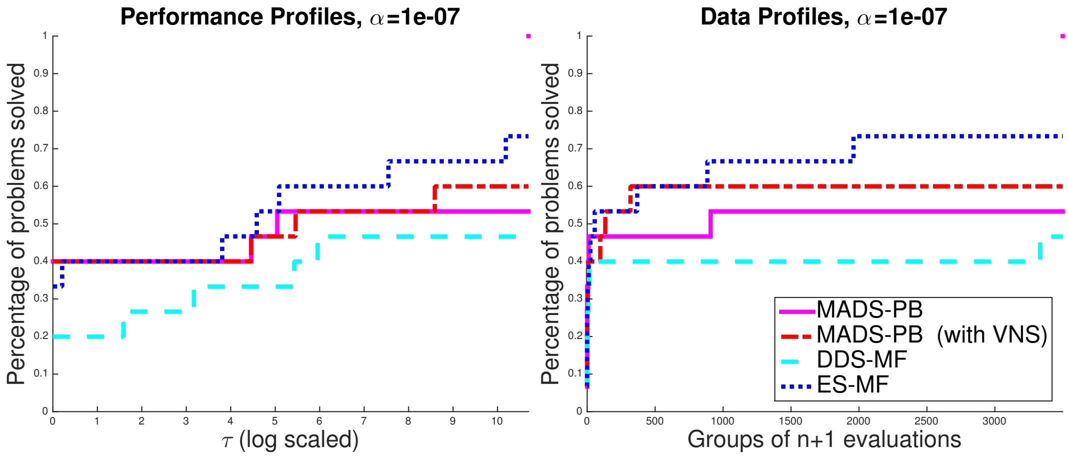
of 10^4. The starting point x_0 is chosen to be the same for all solvers and set to $(LB + UB)/2$ where LB are the lower bound constraints and UB are the upper bound constraints. We consider that all the constraints as QR except the bounds on the design variables which are treated using the ℓ_2-projection for all the solvers. We note that problems `G3`, `G11`, and `WBD` contain equality constraints. When a constraint is of the form $c_i^e(x) = 0$, we use the following relaxed inequality constraint instead $c_i(x) = |c_i^e(x)| - 10^{-5} \leq 0$. We describe our finding using the median performance and data profiles over 20 runs.

Figure 4 reports the obtained profiles for the solvers **MADS-PB**, **DDS-MF** and **ES-MF** using a maximal budget of 10^4. Additionally, we include the profiles of a variant of the solver **MADS-PB** where the variable neighborhood search (VNS) strategy is enabled to enhance its global performance (by setting the flag `vns_search` to 1 in the NOMAD package). The latter solver is denoted by **MADS-PB (with VNS)** in Figure 4. We note also that the solver **CSA-AL** is no longer included in the comparison as it displayed the worst results in our tests (it produced unfeasible solutions on most of the tested problems). Clearly, one can see that, unlike the previous test bed, the **ES-MF** solver outperforms the solvers **MADS-PB** and **DDS-MF**, particularly when considering a large function evaluations. For the low accuracy level (i.e., $\alpha = 10^{-3}$), enabling the VNS option improves significantly the efficiency of **MADS-PB**. For such accuracy, the solver **MADS-PB (with VNS)** reaches better efficiency performance compared to **ES-MF**. However, considering a higher accuracy level (i.e., $\alpha = 10^{-7}$) tends to degrade the performance of **MADS-PB (with VNS)** compared to **ES-MF**.

Tables 2 and 3 depict the final obtained results for the solvers **MADS-PB**, **DDS-MF**, **MADS-PB (with VNS)** and **ES-MF**, using a maximal budget of 10^4 function evaluations. For each problem, we display the optimal objective value found by the solver $f(x_*)$, the associ-



(a) Considering the level of accuracy $\alpha = 10^{-3}$.



(b) Considering the level of accuracy $\alpha = 10^{-7}$.

Figure 4: Median profiles for the solvers **ES-MF**, **MADS-PB**, and **DDS-MF**, using 15 global optimization test problems.

ated constrained violation $g(x_*)$, and the number of objective function evaluations $\#f$ needed to reach x_*. When a solver returns a flag error or encounters an internal problem, we display "$*$". At the solution x_*, one requires at least a tolerance of 10^{-5} on the constraint violation to consider x_* as feasible with respect to QR constraints. Considering the median run, **ES-MF** converged to a feasible solution for all the problems, **MADS-PB** converged as well to a feasible point for all the problems, except the `TCS` problem for which **MADS-PB** returns a flag error. The **DDS-MF** solver could not converge to a feasible solution for three problems `G2`, `G4`, and `G5`. In terms of the objective function value, one can see clearly that **ES-MF** is outperforming both solvers **MADS-PB** and **DDS-MF**. As expected, in terms of function evaluations, **MADS-PB** required in general less function evaluations than **ES-MF** to converge to a solution (but not necessarily better then the one found by **ES-MF**). The use of the variable neighborhood search option within MADS improves significantly its performance, **MADS-PB (with VNS)** is displaying similar performances compared to the **ES-MF**.

Pb	$f(x_*)$			$\#f$			$g(x_*)$		
	Best	Median	Worst	Best	Median	Worst	Best	Median	Worst
MADS-PB									
G1	-12.4531	-12.4531	-12.4531	4202	4202	4202	2e-26	2e-26	2e-26
G2	-0.321533	-0.321533	-0.321533	8194	8194	8194	0	0	0
G3	-0.00101297	-0.00101297	-0.00101297	10000	10000	10000	0	0	0
G4	-30665.5	-30665.5	-30665.5	1846	1846	1846	8.5e-27	8.5e-27	8.5e-27
G6	-6961.81	-6961.81	-6961.81	427	427	427	7.3e-27	7.3e-27	7.3e-27
G7	30.0027	30.0027	30.0027	2161	2161	2161	2.9e-26	2.9e-26	2.9e-26
G8	-0.095825	-0.095825	-0.095825	350	350	350	0	0	0
G9	680.915	680.915	680.915	1769	1769	1769	5e-27	5e-27	5e-27
G10	7973.6	7973.6	7973.6	10000	10000	10000	4.5e-06	4.5e-06	4.5e-06
G11	0.7499	0.7499	0.7499	9355	9355	9355	1e-26	1e-26	1e-26
G12	-1	-1	-1	425	425	425	0	0	0
G13	0.679994	0.679994	0.679994	10000	10000	10000	0	0	0
TCS	*	*	*	*	*	*	*	*	*
WBD	2.21815	2.21815	2.21815	3625	3625	3625	1e-26	1e-26	1e-26
MDO	-16.6007	-16.6007	-16.6007	6837	6837	6837	0	0	0
DDS-MF									
G1	-14.6929	-11.8944	-7.76563	4529	10000	10000	0	0	0
G2	-0.268315	-0.195197	-0.174585	8237	9364	10000	0	0	0
G3	-0.245346	-0.000195272	-0	980	10000	10000	0	0	2.8e-05
G4	-32217.4	-29246.5	-23837.1	10000	10000	10000	0	0.7	6
G6	-7495.49	-7331.06	-7206.23	10000	10000	10000	0.023	0.054	0.11
G7	24.8165	26.2708	30.9808	10000	10000	10000	0	0	0
G8	-0.095825	-0.095825	-0.0258078	285	324	10000	0	0	0
G9	681.499	683.972	691.198	10000	10000	10000	0	0	9.3e-07
G10	3714.74	6463.86	8790.21	6079	10000	10000	0.014	0.086	0.44
G11	0.748826	0.749978	0.750995	10000	10000	10000	0	4.7e-08	1.2e-06
G12	-0.986446	-0.554001	-0.553667	10000	10000	10000	0	2.2e-10	5.8e-08
G13	0.0932763	0.903758	8.50155	10000	10000	10000	0	3.7e-08	1
TCS	0.0154595	0.0514077	0.0547682	10000	10000	10000	0	0	2.8e-06
WBD	2.26572	4.03345	24.2009	684	2103	10000	0	0	39
MDO	-15.8881	-15.3359	-14.0585	585	1028	1738	0	0	0

Table 2: Obtained results with **MADS-PB** and **DDS-MF**, using 15 global optimization test problems.

5 Conclusion

In this paper, we proposed a globally convergent class of ES algorithms where a merit function is used to decide and control the distribution of the generated points. The proposed approach included a restoration procedure which is entered whenever a decrease on the constraint violation function is achieved while the objective function is being considerably increased. The obtained algorithm generalized the work [17] by including quantifiable relaxable constraints. In the spirit of what is achieved in [22], the proposed convergence analysis was organized depending on the number of times Restoration is entered.

We provided numerical tests on problems from the CUTEst collection and a global optimization test bed. The results showed the potential of the proposed merit approach compared

Pb	$f(x_*)$			$\#f$			$g(x_*)$		
	Best	Median	Worst	Best	Median	Worst	Best	Median	Worst
ES-MF									
G1	-15.0003	-15.0003	-12.4537	10000	10000	10000	1.2e-07	1.7e-07	5.6e-07
G2	-0.756445	-0.716013	-0.252014	5851	10000	10000	0	0	1.9e-10
G3	-1.00565	-1.00538	-1.03027	10000	10000	10000	0	2.7e-06	3e-06
G4	-30665.5	-30664.8	-30649.1	10000	10000	10000	0	0	9.6e-05
G6	-7865.39	-6953.54	-6369.01	4493	8406	10000	0	1.4e-06	9.7e-05
G7	24.3035	24.3037	24.3062	10000	10000	10000	1.1e-08	1.3e-08	1.5e-06
G8	-0.095825	-0.095825	-0.0273164	1492	1653	10000	0	0	2.7e-08
G9	680.629	680.629	680.629	7231	8526	10000	3.6e-07	3.6e-07	3.6e-07
G10	7086.26	11177.6	18860.8	7288	9899	10000	0	4.3e-05	9.4e-05
G11	0.7499	0.7499	0.7499	2830	3522	10000	1.6e-09	2.5e-09	3.9e-07
G12	-1	-0.960558	-0.783887	1457	3533	4281	0	1.6e-09	8.6e-09
G13	0.0539573	0.438745	1	5465	10000	10000	1.2e-16	1.8e-09	3.2e-08
TCS	0.0126649	0.0126688	0.0132221	6598	10000	10000	1.1e-12	1.8e-10	6.9e-10
WBD	2.19747	2.21258	2.53771	8488	10000	10000	2.6e-10	2.8e-08	1.7e-08
MDO	-16.612	-16.612	-16.6119	5031	10000	10000	0	0	1.1e-14
MADS-PB (with VNS)									
G1	-15	-15	-15	10000	10000	10000	0	0	0
G2	-0.697381	-0.697381	-0.697381	10000	10000	10000	0	0	0
G3	-0.0870995	-0.0870995	-0.0870995	10000	10000	10000	0	0	0
G4	-30665.5	-30665.5	-30665.5	10000	10000	10000	0	0	0
G6	-6961.81	-6961.81	-6961.81	6523	6523	6523	3.2e-27	3.2e-27	3.2e-27
G7	24.8226	24.8226	24.8226	10000	10000	10000	0	0	0
G8	-0.095825	-0.095825	-0.095825	6505	6505	6505	0	0	0
G9	680.632	680.632	680.632	10000	10000	10000	0	0	0
G10	7087.99	7087.99	7087.99	10000	10000	10000	0	0	0
G11	0.7499	0.7499	0.7499	10000	10000	10000	0	0	0
G12	-1	-1	-1	10000	10000	10000	0	0	0
G13	0.781443	0.781443	0.781443	10000	10000	10000	0	0	0
TCS	*	*	*	*	*	*	*	*	*
WBD	2.21815	2.21815	2.21815	10000	10000	10000	1e-26	1e-26	1e-26
MDO	-16.6054	-16.6054	-16.6054	10000	10000	10000	0	0	0

Table 3: Obtained results with **ES-MF** and **MADS-PB (with VNS)**, using 15 global optimization test problems.

to existing direct search DFO solvers, in particular when using a large number of function evaluations.

References

[1] A. Atamna, A. Auger, and N. Hansen. On invariance and linear convergence of evolution strategies with augmented Lagrangian constraint handling. *Theoretical Computer Science*, 0:1–53, 2018.

[2] C. Audet, A. R. Conn, S. Le Digabel, and M. Peyrega. A progressive barrier derivative-free trust-region algorithm for constrained optimization. *Comput. Optim. Appl.*, 71(2):307–329, 2018.

[3] C. Audet and W. Hare. *Derivative-Free and Blackbox Optimization*. Springer Series in Operations Research and Financial Engineering. Springer International Publishing, Cham, Switzerland, 2017.

[4] C. Audet and J. E. Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM J. Optim.*, 17:188–217, 2006.

[5] C. Audet and J. E. Dennis Jr. A progressive barrier for derivative-free nonlinear programming. *SIAM J. Optim.*, 20(1):445–472, 2009.

[6] A. Auger, D. Brockhoff, and N. Hansen. Benchmarking the local metamodel CMA-ES on the noiseless BBOB'2013 test bed. In *Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation*, GECCO '13 Companion, pages 1225–1232, New York, NY, USA, 2013. ACM.

[7] A. Auger, N. Hansen, Z. J. Perez, R. Ros, and M. Schoenauer. Experimental comparisons of derivative free optimization algorithms. In *8th International Symposium on Experimental Algorithms*, pages 3–15. Springer Verlag, 2009.

[8] Z. Bouzarkouna. *Well placement optimization.* PhD thesis, University Paris-Sud - Laboratoire de Recherche en Informatique, 2012.

[9] F. H. Clarke. *Optimization and Nonsmooth Analysis.* John Wiley & Sons, New York, 1983. Reissued by SIAM, Philadelphia, 1990.

[10] C. A. C. Coello. List of references on constraint-handling techniques used with evolutionary algorithms. `https://www.cs.cinvestav.mx/~constraint/constbib.pdf`.

[11] C. A. C. Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering*, 91:1245–1287, January 2002.

[12] C. A. Coello Coello and E. M. Montes. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Advanced Engineering Informatics*, 16:193–203, 2002.

[13] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization.* MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.

[14] Y. Diouane. *Globally convergent evolution strategies with application to Earth imaging problem in geophysics.* PhD thesis, Institut National Polytechnique de Toulouse, 2014.

[15] Y. Diouane, S. Gratton, X. Vasseur, L. N. Vicente, and H. Calandra. A parallel evolution strategy for an earth imaging problem in geophysics. *Optim. Eng.*, 17:3–26, 2016.

[16] Y. Diouane, S. Gratton, and L. N. Vicente. Globally convergent evolution strategies. *Math. Program.*, 152:467–490, 2015.

[17] Y. Diouane, S. Gratton, and L. N. Vicente. Globally convergent evolution strategies for constrained optimization. *Comput. Optim. Appl.*, 62:323–346, 2015.

[18] E. D. Dolan, J. J. Moré, and T. S. Munson. Optimality measures for performance profiles. *SIAM J. Optim.*, 16:891–909, 2006.

[19] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Math. Program.*, 91:239–269, 2002.

[20] N. I. M. Gould, D. Orban, and P. L. Toint. CUTEst: a Constrained and Unconstrained Testing Environment with safe threads for mathematical optimization. *Comput. Optim. Appl.*, 60:545–557, 2015.

[21] R. B. Gramacy and S. Le Digabel. The mesh adaptive direct search algorithm with treed gaussian process surrogates. *Pacific J. Optim.*, 11:719–747, 2015.

[22] S. Gratton and L. N. Vicente. A merit function approach for direct search. *SIAM J. Optim.*, 24:1980–1998, 2014.

[23] N. Hansen. The CMA Evolution Strategy: A tutorial. Available at `https://www.lri.fr/~hansen/cmatutorial.pdf`, 2011.

[24] N. Hansen, A. Auger, R. Ros Raymond, S. Finck, and P. Pošík. Comparing results of 31 algorithms from the black-box optimization benchmarking bbob-2009. In *Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*, GECCO '10, pages 1689–1696, New York, NY, USA, 2010. ACM.

[25] N. Hansen, A. Ostermeier, and A. Gawelczyk. On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. In L. Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms, Pittsburgh*, pages 57–64, 1995.

[26] W. Hock and K. Schittkowski. *Test Examples for Nonlinear Programming Codes*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1981.

[27] J. Jahn. *Introduction to the Theory of Nonlinear Optimization*. Springer-Verlag, Berlin, 1996.

[28] T. G. Kolda, R. M. Lewis, and V. Torczon. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Rev.*, 45:385–482, 2003.

[29] S. Koziel and Z. Michalewicz. Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. *Evol. Comput.*, 7:19–44, March 1999.

[30] O. Kramer. A review of constraint-handling techniques for evolution strategies. *Appl. Comp. Intell. Soft Comput.*, 2010:1–19, January 2010.

[31] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm. *ACM Trans. Math. Software*, 37:1–15, 2011.

[32] S. Le Digabel and S.M. Wild. A Taxonomy of Constraints in Simulation-Based Optimization. Technical Report G-2015-57, Les cahiers du GERAD, 2015.

[33] J. M. Martínez and F. N. C. Sobral. Constrained derivative-free optimization on thin domains. *J. Global Optim.*, 56(3):1217–1232, 2013.

[34] Z. Michalewicz and M. Schoenauer. Evolutionary algorithms for constrained parameter optimization problems. *Evol. Comput.*, 4:1–32, 1996.

[35] J. J. Moré and S. M. Wild. Benchmarking derivative-free optimization algorithms. *SIAM J. Optim.*, 20:172–191, 2009.

[36] J. Nocedal and S. J. Wright. *Numerical optimization*. Springer series in operations research and financial engineering. Springer, New York, NY, 2. ed. edition, 2006.

[37] I. Rechenberg. *Evolutionsstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution*. Frommann-Holzboog, 1973.

[38] L.M. Rios and N.V. Sahinidis. Derivative-free optimization: A review of algorithms and comparison of software implementations. *J. Global Optim.*, 56:1247–1293, 2010.

[39] C. Tribes, J.-F. Dubé, and J.-Y. Trépanier. Decomposition of multidisciplinary optimization problems: formulations and application to a simplified wing design. *Optim. Eng.*, 37:775–796, 2005.