Int. J. Simul. Multisci. Des. Optim. 2015, **6**, A1
© R. Chocat et al., Published by EDP Sciences, 2015
DOI: 10.1051/smdo/2015001

Available online at:
www.ijsmdo.org

RESEARCH ARTICLE

OPEN ACCESS

Modified Covariance Matrix Adaptation – Evolution Strategy algorithm for constrained optimization under uncertainty, application to rocket design

Rudy Chocat[1], Loïc Brevault[2,3], Mathieu Balesdent[2,*], and Sébastien Defoort[2]

[1] IFMA, EA3867, Laboratoires de Mécanique et Ingénieries, Clermont Université, CP 104488, 63000 Clermont-Ferrand, France
[2] Onera – The French Aerospace Lab, BP 80100, 91123 Palaiseau Cedex, France
[3] CNES – Launchers Directorate, 52 rue Jacques Hillairet, 75612 Paris, France

Received 9 October 2014 / Accepted 3 March 2015

Abstract – The design of complex systems often induces a constrained optimization problem under uncertainty. An adaptation of CMA-ES(λ, μ) optimization algorithm is proposed in order to efficiently handle the constraints in the presence of noise. The update mechanisms of the parametrized distribution used to generate the candidate solutions are modified. The constraint handling method allows to reduce the semi-principal axes of the probable research ellipsoid in the directions violating the constraints. The proposed approach is compared to existing approaches on three analytic optimization problems to highlight the efficiency and the robustness of the algorithm. The proposed method is used to design a two stage solid propulsion launch vehicle.

Key words: Evolutionary Strategy, Covariance Matrix Adaptation, CMA-ES, Uncertainty, Constrained optimization, Rocket design.

1 Introduction

The design of complex systems, such as aerospace vehicles, can be expressed as a non linear constrained optimization problem solving. The determination of the optimal system architecture requires a global exploration of the design space involving repeated evaluations of computationally expensive black box functions used to model the different disciplines (e.g. in aerospace: structure, propulsion, aerodynamics). Moreover, in the early design phases, uncertainties arise due to lack of knowledge of the system characteristics and the environment (e.g. propellant combustion rate, wind gust) and to the use of low fidelity analyses (for instance analytic code instead of numerically costly finite element analysis). Therefore, the design of a complex system at the conceptual design phase entails a constrained optimization problem under uncertainty that can be formulated as:

$$\begin{array}{ll} \text{Min} & \Xi[F(\mathbf{z}, \mathbf{U})] \\ \text{wrt} & \mathbf{z} \end{array} \tag{1}$$

$$\text{st} \quad \mathbf{K}[\mathbf{g}(\mathbf{z}, \mathbf{U})] \leq 0 \tag{2}$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \tag{3}$$

with:

- $\mathbf{z}$: the vector of design variables $\mathbf{z} \in \mathbb{R}^n$ (e.g. rocket diameter, propellant masses),
- $\mathbf{U}$: the vector of the uncertain variables $\mathbf{U} \in \mathbb{R}^p$ (e.g. wind gust during the rocket launch, material characteristics). In the paper, the probability theory is used to model uncertainty.
- F: $\mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}$: the performance function (e.g. propulsive speed increment ΔV, Gross Lift-Off Weight),
- Ξ: a measure of uncertainty for the performance function F (e.g. the expected value, a linear combination of expected value and standard deviation),
- $\mathbf{g}$: the vector of the inequality constraint functions, g_i, $i \in \{1, \ldots, m\}$ (e.g. maximal allowed stress in the structures, maximal tolerance on orbit injection accuracy),
- $\mathbf{K}$: a vector of measures of uncertainty for the inequality constraint functions (e.g. linear combination of expected value and standard deviation, probability of failure),
- $\mathbf{z}_{\min}$ and $\mathbf{z}_{\max}$ the lower and upper bounds for the design variables.

Being able to handle uncertainty at early design phases is essential to efficiently characterize the optimal system design

---

*e-mail: `mathieu.balesdent@onera.fr`

and its performances. It can reduce time and cost of the next design phases by avoiding redesign process [18]. However, solving the optimization problem (Eqs. (1)–(3)) is challenging due to the presence of both the uncertainty and inequality constraints. A brief overview of the existing methods to handle uncertainty and inequality constraints is provided in the next paragraphs.

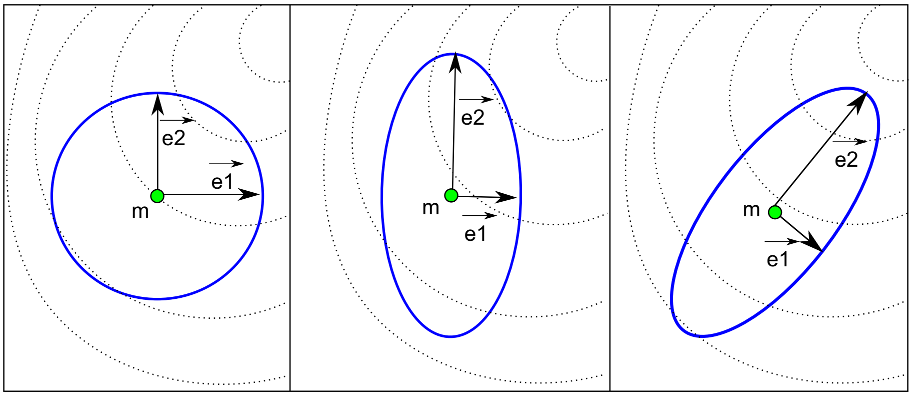
The presence of noise in the optimization problem (Eqs. (1)–(3)) results from the estimation of the measures of uncertainty (Ξ and $\mathbf{K}$). Several methods exist to compute these measures and the most classical approach is a numerical approximation by the Crude Monte Carlo (CMC) method. The CMC estimator of the uncertainty measure is also a random variable. In order to numerically handle the presence of noise in optimization, several approaches have been proposed:

- Re-sampling [3]: the re-sampling method consists of repeated evaluations of the objective and the constraint uncertainty measures for the same design variable value $\mathbf{z}$. Then, a statistics of the repeated samples is used instead of the single evaluation of the objective and the constraint measures to decrease the impact of noise. The main drawback of this approach is the increase of the computational cost due to the repeated evaluations of expensive functions.
- Surrogate model [3]: surrogate models of the objective function measure Ξ (and/or the constraint measures $\mathbf{K}$) are built from the evaluated measures. In general, surrogates smooth the noisy functions and decrease the impact of uncertainty in the optimization. The main drawback lies in the difficulty to build accurate surrogate models in high dimensions.
- Population based algorithms [10]: to address uncertainty, optimization algorithms relying on a population of candidates allow to increase the size of the population to enlarge its spread and to obtain more information and smooth the noise.

Moreover, the presence of uncertainty makes the use of classical optimization algorithms, such as gradient based optimization algorithms, not suited for optimization. Indeed, the gradients of the objective uncertainty measure or the constraint uncertainty measures are noisy resulting in possible erroneous descent directions. Diverse derivative-free algorithms have been proposed in the literature to solve optimization under uncertainty problems [15]. Among these, the population based optimization algorithms seem promising [6]. Swarm Intelligence (Particle Swarm Optimization [7], Artificial Bee Colonies [13], etc.), Differential Evolution [17], Evolutionary Algorithms (Genetic Algorithm [12]) or Evolution Strategies (Covariance Matrix Adaptation – Evolution Strategy [10]) have been investigated to solve noisy optimization problems. One of the current issue with the derivative-free algorithms is the constraint handling which relies mainly on heuristic approaches and is problem dependent [16]. A comprehensive overview of constraint handling in derivative-free algorithms is presented in the survey [16]. The most commonly applied methods to handle the constraints are:

- Death penalty [19]: it is the simplest method to handle constraints. The solution that does not satisfy the constraints is rejected and another potential solution is re-evaluated until one candidate solution satisfies the constraints. The advantage of the approach is that it does not modify the optimization algorithm but the method is very expensive because no information is learned from an unfeasible solution (a solution which does not satisfy the constraints) to characterize the non feasible space. Furthermore, if the feasible space is restricted compared to the design space, the computational cost becomes prohibitive because a high number of samples has to be generated to obtain feasible solutions.
- Penalization [5, 6]: this approach consists in replacing the objective function $\Xi\ [F(\mathbf{z}, \mathbf{U})]$ by a combination of the objective function and a penalization function Π such as: $\Xi\ [F(\mathbf{z}, \mathbf{U})] + \Pi\ (\mathbf{K}[\mathbf{g}(\mathbf{z}, \mathbf{U})])$. The penalization function can be fixed or can change as a function of the number of iterations. When a solution violates the constraints, the objective function is deteriorated by a factor proportional to the penalization function and the value of the constraints. Despite its simplicity, the main drawback of this approach lies in the determination of a suitable penalization function which depends on the objective function and the constraints and is thus problem dependent.
- Multi-objective [4]: this method transforms the optimization problem into a multi-objective optimization problem by considering the minimization of the violation of the constraints as an objective. Dedicated multi-objective optimization algorithms can be used, however, it often results in an increase of the computational cost [16].
- Surrogate model [14]: this approach builds a surrogate model based on the unfeasible solutions in order to approximate the non feasible zones. However, this approach requires enough unfeasible solutions to construct accurate surrogate models. Moreover, it can be difficult to build the surrogates in high dimension or if the constraints are highly non linear.

Among the derivative-free algorithms, the Covariance Matrix Adaptation – Evolution Strategy (CMA-ES) [10] is particularly competitive for real-valued black box functions as highlighted in several extensive benchmarks [8, 9]. Moreover, a treatment of uncertainty has been proposed for CMA-ES and has been successfully tested in a benchmark of optimization under uncertainty problems [8]. However, the test problems used to evaluate the performances of CMA-ES are unconstrained optimization problems and only few studies focus on the application of CMA-ES to constrained optimization problems [2, 6]. An adaptive penalty function has been proposed to update the penalty coefficient as a function of the sum of the violated constraint values [6]. Arnold and Hansen [1] proposed a new approach to handle the constraints for a simplified version of CMA-ES: (1+1)-CMA-ES which involves one offspring candidate generated from one parent. Modified (1+1)-CMA-ES approximates the normal vector directions of the



N(0,I) N(0,D) N(0,C)

Figure 1. Three ellipsoids, depicting three different normal distributions, where **I** is the identity matrix, **D** is a diagonal matrix, and **C** is a positive definite covariance matrix. Thin dot lines depict objective function contour lines.

constraint boundaries in the vicinity of the current candidate solution. A control of the covariance of the distribution of the offspring candidate allows to get closer to the boundary of the feasible regions without violating them. The approach provides interesting results but is limited to (1+1)-CMA-ES which becomes inefficient in high dimensions (local convergence, large number of calls to the functions) [1].

The main objective of this paper is to adapt CMA-ES(λ, μ), which involves λ offspring candidates generated from μ parents, to efficiently solve constrained optimization problem under uncertainty. The proposed approach is based on a control of the covariance of the distribution of the offspring candidates through a modification of the covariance matrix that characterizes the probable research hypervolume in order to generate candidates without violating the constraints. The proposed adaptations take into account the specificities of the update and the selection mechanisms of CMA-ES(λ, μ) that differ from (1+1)-CMA-ES due to the presence of a population instead of a single candidate. The remainder of the paper is organized as follows. Section 2 provides an overview of the original CMA-ES(λ, μ) and the adapted (1+1)-CMA-ES to handle constraints. Section 3 presents the modifications of CMA-ES(λ, μ) to handle constraints in optimization problems. Section 4 evaluates the algorithm efficiency on three analytic test functions and on the design of a two stage solid rocket. A comparison with penalized CMA-ES(λ, μ) and modified (1+1)-CMA-ES is provided.

2 Description of CMA-ES(λ, μ) and modified (1+1)-CMA-ES

2.1 CMA-ES(λ, μ) algorithm

The Covariance Matrix Adaptation – Evolution Strategy (CMA-ES) introduced by Hansen et al. [10] belongs to the Evolution Strategy algorithm family. A brief overview of CMA-ES(λ, μ) is provided in the section in order to understand the proposed modifications for the constraint handling, for more information on the algorithm see [10]. CMA-ES(λ, μ) is used to solve unconstrained optimization problems. CMA-ES relies on a distribution model of a candidate population (parametrized multivariate normal distribution) in order to explore the design space. It is based on a selection and an adaptation process of the candidate population. In CMA-ES(λ, μ), at each generation, λ offspring candidates are generated from μ parents. At the next generation, to select the new parents from the offspring candidates, a (λ, μ)-selection is used, the μ best offspring candidates are chosen (with respect to their ranking according to the objective function). The multivariate normal distribution has an infinite support, but an iso-probability contour (for instance at ±3 standard deviation of the mean) is characterized by an ellipsoid delimiting a probable research hypervolume (Figure 1). Throughout the generations, the research hypervolume is updated in order to converge and to shrink around the global optimum. CMA-ES(λ, μ) generates the population by sampling a multivariate normal distribution:

$$\mathbf{z}_t^{(k+1)} \sim \mathbf{m}^{(k)} + \sigma^{(k)}\mathcal{N}\left(0,\mathbf{C}^{(k)}\right), \quad \text{for } t = 1,\ldots,\lambda \qquad (4)$$

with $\mathbf{z}_t^{(k+1)} \in \mathbb{R}^n$ an offspring candidate generated from a mean vector $\mathbf{m}^{(k)}$, a step size $\sigma^{(k)}$ and a multivariate normal distribution $\mathcal{N}(0, \mathbf{C}^{(k)})$ with zero mean and a covariance matrix $\mathbf{C}^{(k)} \in \mathbb{R}^n \times \mathbb{R}^n$. λ is the size of the population generated at each iteration (k). The normal distribution is characterized by a positive definite covariance matrix $\mathbf{C}^{(k)}$ in order to allow homothetic transformations and rotations of the probable research hypervolume (Figure 1). The update of the covariance matrix incorporates dependence between the past generations and between the μ best candidates from the previous generation [10]. The mean vector characterizes the center of the next population and is determined by a combination process through the weighting of the μ best candidates: $\mathbf{m}^{(k+1)} = \sum_{i=1}^{\mu} w_i \mathbf{z}_{(i)}^{(k+1)}$ with $\sum_{i=1}^{\mu} w_i = 1$, $w_1 > w_2 > \cdots > w_\mu > 0$ the weighting coefficients and $(\mathbf{z}_{(1)}, \ldots, \mathbf{z}_{(\mu)})$ the best candidates among the offspring ranked according to the fitness value. The weighting coefficients are determined based on the number of μ best candidates according to [10]. A simplified version of CMA-ES(λ, μ) is described in Algorithm 1. A detailed description of the selection and update mechanisms can be found in [10].

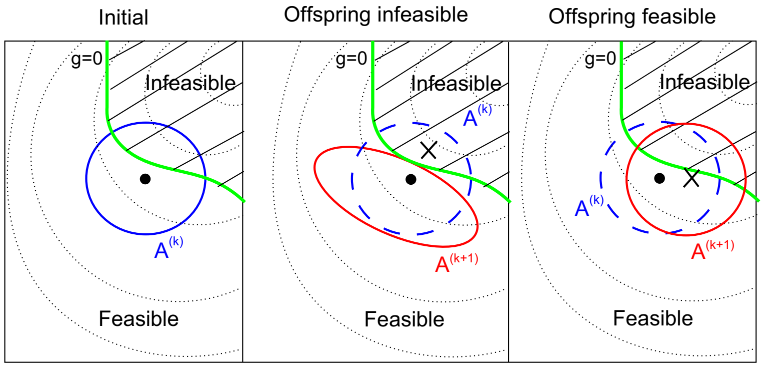


Figure 2. The dot is the parent, the cross is the generated offspring, the solid line circle is characterized by $\mathbf{A}^{(k)}$ defining the ellipsoid delimiting an iso-probability research hypervolume. At the center, the red ellipsoid represents the update of $\mathbf{A}^{(k+1)}$ in order to take into account the constraint violation by the offspring. On the right figure, the offspring does not violate the constraint resulting in a standard covariance matrix update.

Algorithm 1 CMA-ES(λ, μ) [10]

```
1) Initialize the covariance matrix $\mathbf{C}^{(0)} = \mathbf{I}$, the step size $\sigma^{(0)}$ and the selection parameters [10]
2) Initialize the mean vector $\mathbf{m}^{(0)}$ to a random candidate, $k \leftarrow 0$
while CMA-ES convergence criteria are not reached do
    3-1) Generate $\lambda$ new offspring candidates according to: $\mathbf{z}_t^{(k+1)} \sim \mathbf{m}^{(k)} + \sigma^{(k)}\mathcal{N}(0, \mathbf{C}^{(k)})$, $t \in \{1, \ldots, \lambda\}$
    3-2) Evaluate candidates and rank them based on the objective function
    3-3) Determine the mean vector given the weighting coefficients of the $\mu$ best candidates: $\mathbf{m}^{(k+1)} = \sum_{i=1}^{\mu} w_i \mathbf{z}_{(i)}^{(k+1)}$
    3-4) Update covariance matrix $\mathbf{C}^{(k+1)}$ and the step size $\sigma^{(k+1)}$ according to [10], $k \leftarrow k + 1$
end while
4) return best candidate $\mathbf{z}_{\mathbf{best}}$
```

The convergence criteria can either be based on the maximum number of iterations (function evaluations), the fitness value, the standard deviation of the current population smaller than a given tolerance, or the covariance matrix **C** which becomes numerically not positive definite. In the next paragraph, an adaptation of CMA-ES for uncertainty handling is described.

2.2 CMA-ES(λ, μ) algorithm for optimization under uncertainty

Several features ensure the CMA-ES robustness with respect to the presence of uncertainty in the objective function: the population-based approach, the weighted averaging in the recombination process, the rank based and the non-elitist selection (not based on the best offspring candidate). However, if the noise is too high compared to the objective function (signal to noise ratio too low) it perturbs the algorithm convergence. An appropriate handling of uncertainty has been proposed by Hansen et al. [11] to overcome this issue. Modified selection and update mechanisms are performed when the noise is above a given threshold. It is based on a re-sampling approach and involves re-evaluation of the objective function. Because CMA-ES(λ, μ) is only based on the rank of the candidates, the effective noise is evaluated by monitoring changes or stability of the offspring candidate ranking. If the offspring candidate ranking is changed after the re-evaluation of the objective function, the ranking change of the offspring candidates is aggregated into a metric quantifying the uncertainty level [11]. If the noise is higher than a given uncertainty level threshold, the step size σ is increased. The increase of σ ensures that despite the noise, sufficient selection information is available [11].

A benchmark of algorithm dealing with optimization under uncertainty has been performed and the treatment of uncertainty with CMA-ES(λ, μ) allows to obtain accurate results [8]. However, CMA-ES(λ, μ) is not able to efficiently handle the constraints. Indeed, only penalization [5, 6] or surrogate based methods [14] have been proposed. These approaches can be effective but are problem dependent and require accurate tuning of hyper parameters. A simplified version of CMA-ES(λ, μ) called (1+1)-CMA-ES has been adapted to handle constraints and is detailed in the next section.

2.3 (1+1)-CMA-ES with constraint handling

(1+1)-CMA-ES is a simplified version [1] of CMA-ES(λ, μ) with only one offspring generated from one parent, "+" means that the selection is done between the parent and the offspring. As in CMA-ES(λ, μ), the offspring candidate solution is generated as:

$$\mathbf{z}^{(k+1)} \sim \mathbf{z}^{(k)} + \sigma^{(k)}\mathcal{N}\left(0, \mathbf{C}^{(k)}\right) \tag{5}$$

(1+1)-CMA-ES is easier to implement as only one offspring $\mathbf{z}^{(k+1)}$ is generated from one parent $\mathbf{z}^{(k)}$ at each generation and the selection is between the parent and the offspring. The update mechanisms for (1+1)-CMA-ES are detailed in [1].

To incorporate the handling of constraints, Arnold and Hansen [1] proposed to reduce the covariance of the distribution of the offspring candidate in the approximated directions of the normal vectors of the constraint boundaries in the vicinity of the current parent candidate solution. For that purpose, the matrix $\mathbf{A}^{(k)}$, which is the Cholesky decomposition of $\mathbf{C}^{(k)} : \mathbf{A}^{(k)}\mathbf{A}^{(k)^T} = \mathbf{C}^{(k)}$, is updated in case of constraint violations in order to avoid generating candidates in the next generations that will violate the constraints. $\mathbf{A}^{(k)}$ is used as it is easier to compute its inverse than for $\mathbf{C}^{(k)}$. A vector characterizing the constraints $\mathbf{v}_j^{(k)} \in \mathbb{R}^n$ is defined, initialized to be zero, and updated according to:

$$\mathbf{v}_j^{(k)} \leftarrow (1 - c_c)\mathbf{v}_j^{(k)} + c_c\mathbf{A}^{(k)}\mathbf{z}^{(k)} \quad \forall j \in \{1, \ldots, m\} \tag{6}$$

where $\mathbf{v}_j^{(k)}$ is an exponentially fading record of steps that have violated the constraints and c_c a parameter characterizing how fast the information present in $\mathbf{v}_j^{(k)}$ fades. In the generations in which the offspring candidate is unfeasible, the Cholesky matrix is updated according to:

$$\mathbf{A}^{(k)} \leftarrow \mathbf{A}^{(k)} - \frac{\beta}{\sum_{j=1}^{m} \mathbb{1}_{g_j(\mathbf{z}^{(k)})>0}} \sum_{j=1}^{m} \mathbb{1}_{g_j(\mathbf{z}^{(k)})>0} \frac{\mathbf{v}_j^{(k)}\mathbf{w}_j^{(k)^T}}{\mathbf{w}_j^{(k)^T}\mathbf{w}_j^{(k)}} \tag{7}$$

with $\mathbf{w}_j^{(k)} = \mathbf{A}^{(k)^{-1}}\mathbf{v}_j^{(k)}$, the indicator function associated to the constraint g_j: $\mathbb{1}_{g_j(\mathbf{z}^{(k)})>0}$ and β a parameter controlling the reduction of the covariance of the distribution. For $\beta = 0$, the algorithm is identical to the standard (1+1)-CMA-ES. The update of the matrix $\mathbf{A}^{(k)}$ allows to modify the scale and the orientation of the research hypervolume in order to be tangential to the constraints and to avoid its violation (Figure 2). Modified (1+1)-CMA-ES for constraint handling is interesting because it is not problem dependent. Experimental evaluations have been performed highlighting its efficiency for unimodal constrained optimization problems. However, as (1+1)-CMA-ES, it is not able to optimize multimodal functions and becomes inefficient in high dimensions [1].

To overcome these drawbacks, in the next section, an adaptation of CMA-ES(λ, μ) for constraint handling is proposed inspired from the (1+1)-CMA-ES approach.

3 Proposed adaptation of CMA-ES(λ, μ) for constraint handling

The proposed approach of CMA-ES(λ, μ) for constraint handling is based on the same approach as modified (1+1)-CMA-ES. However, it is necessary to adapt it to take into account the specificities of CMA-ES(λ, μ). Indeed, CMA-ES(λ, μ) generates a population instead of a single offspring candidate. Thus, each offspring candidate can potentially violate one or several constraints. Moreover, the selection of the μ best candidates is based on the rank of the objective function. However, these best candidates can also violate the constraints. Depending if the μ best candidates are feasible or not, or if only a fraction of them is feasible, or on the number of violated constraints, the covariance matrix used to generate the offspring candidates has to be modified in order to avoid the generation of unfeasible offspring candidates. The research hypervolume engendered by an iso-probability contour of the multivariate normal distribution $\mathcal{N}(0, \mathbf{C})$ can be represented by a n-dimensional ellipsoid. In the proposed approach, the constraint handling method allows to reduce the semi-principal axes of the research ellipsoid in the directions violating the constraints. The eigenvalues of the covariance matrix $\mathbf{C}$ control the length of the semi-principal axes. The decrease of the eigenvalues reduces the semi-principal axis lengths. The covariance matrix, which is symmetric positive definite, can be decomposed according to:

$$\mathbf{C} = \mathbf{PDDP}^T = \mathbf{PD}^2\mathbf{P}^T \tag{8}$$

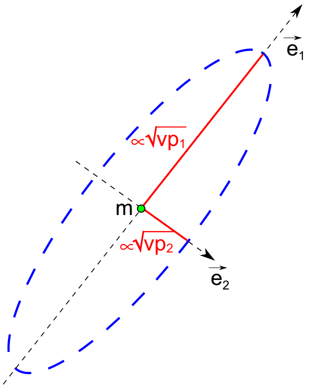


Figure 3. Parametrization of the ellipsoid defining the probable research hypervolume.

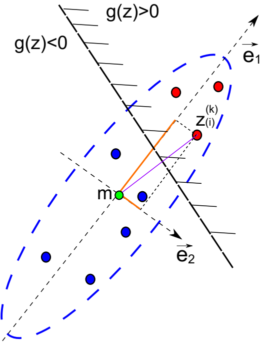


Figure 4. Violation of the constraint and projection over the eigenvectors, blue = feasible, red = unfeasible candidates.

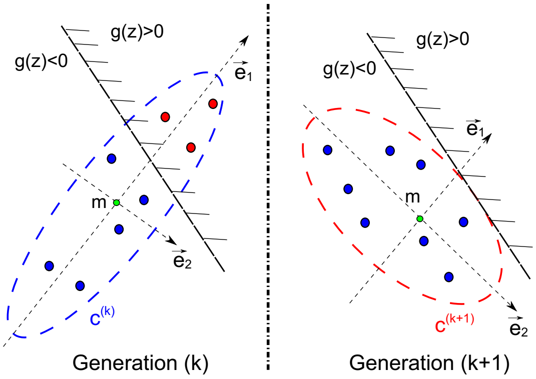


Figure 5. Evolution of the covariance matrix due to the constraint violation, blue = feasible, red = unfeasible candidates.

where $\mathbf{P}$ is an orthogonal matrix such that: $\mathbf{P}\mathbf{P}^T = \mathbf{P}^T\mathbf{P} = \mathbf{I}$. The columns of $\mathbf{P}$ form an orthogonal basis of eigenvectors of $\mathbf{C}$. $\mathbf{D} = \mathrm{diag}(\sqrt{vp_1}, \ldots, \sqrt{vp_n})$ is a diagonal matrix with the square roots of eigenvalues of $\mathbf{C}$. As illustrated in Figure 3, the square roots of the covariance matrix eigenvalues $\sqrt{vp_i}$ are proportional to the semi-principal axis lengths of the ellipsoid defining the sampling hypervolume.

At the generation (k), between the step 3-3) and 3-4) of Algorithm 1, if any of the m constraints is violated by any of the μ best offspring candidates, the covariance matrix is modified according to Algorithm 2.

Algorithm 2 Proposed CMA-ES(λ, μ) covariance matrix modification

$$3.3.1)\ \text{Diagonalize } \mathbf{C}^{(k)} \text{ such that } \mathbf{P}\mathbf{D}^{(k)^2}\mathbf{P}^T = \mathbf{C}^{(k)} \tag{9}$$

$$3.3.2)\ \mathbf{S}^{(k)} \leftarrow \mathbf{P}\mathbf{D}^{(k)^2}\mathbf{P}^T - \gamma \mathbf{P}\,\mathrm{diag}\left(\sqrt{vp_{m_1}^{(k)}}, \ldots, \sqrt{vp_{m_n}^{(k)}}\right)^2 \mathbf{P}^T \tag{10}$$

$$3.3.3)\ \mathbf{C}^{(k)} \leftarrow \left[\frac{\det(\mathbf{C}^{(k)})}{\det(\mathbf{S}^{(k)})}\right]^{1/n} \mathbf{S}^{(k)} \tag{11}$$

with:

$$\sqrt{vp_{m_i}^{(k)}} = \left(\sqrt{vp_i^{(k)}} \sum_{j=1}^{m} \frac{\sum_{l=1}^{\mu} \mathbb{1}_{g_j(\mathbf{z}_{(l)}^{(k)})>0} w_{lj} \mathrm{Proj}_{\vec{e}_i}\left[\mathbf{z}_{(l)}^{(k)} - \mathbf{m}^{(k)}\right]}{\sum_{t=1}^{\mu} \mathbb{1}_{g_j(\mathbf{z}_{(t)}^{(k)})>0} w_{tj}} \right) \tag{12}$$

The covariance matrix is diagonalized, equation (9) and the eigenvalues $vp_i^{(k)}$ of the covariance matrix $\mathbf{C}^{(k)}$ are modified. The new eigenvalues are the former eigenvalues minus a term $vp_{m_i}^{(k)}$ taking into account the violation of the constraints. $\sqrt{vp_{m_i}^{(k)}}$, equation (12), is a function of the former eigenvalues $\sqrt{vp_i^{(k)}}$, of the indicator function of the constraint $\mathbb{1}_{g_j(\mathbf{z}_{(l)}^{(k)})>0}$, of the weighting coefficients w_{lj} and of the projection $\mathrm{Proj}_{\vec{e}_i}\left[\mathbf{z}_{(l)}^{(k)} - \mathbf{m}^{(k)}\right]$ of the distance between an ordered candidate violating the constraints $\mathbf{z}_{(l)}^{(k)}$ and the mean point $\mathbf{m}^{(k)}$ in the direction of the eigenvector $\vec{e}_i$ corresponding to the eigenvalue $vp_i^{(k)}$. γ is a parameter similar to β in (1+1)-CMA-ES. For $\gamma = 0$ the proposed algorithm is similar to the classical CMA-ES(λ, μ). For each constraint g_j, the μ_{cj} candidates among the μ best candidates that violate the constraint are ranked according to the constraint value. The weighting coefficients w_{ij} for each constraint g_j are defined according to the same rule as for the recombination process for the calculation of $\mathbf{m}^{(k)}$:

$$w_{ij} = \frac{\ln(\mu_{cj}+1) - \ln(i)}{\mu_{cj}\ln(\mu_{cj}+1) - \sum_{k=1}^{n}\ln(k)} \tag{13}$$

and $\sum_{i=1}^{\mu} w_{ij} = 1$ where: $w_{1j} \geq \cdots \geq w_{\mu j} \geq 0$. w_{1j} is associated to the candidate that violates the most the constraint g_j and $w_{1\mu_c}$ with the candidate that violates the less the constraint. For the candidates among the μ best that do not violate the constraint, the indicator function is equal to zero and therefore these candidates do not participate in the modification of the covariance matrix. The projection of the violation distance along the eigenvector (Figure 4) allows to reduce the covariance matrix in the direction orthogonal to the constraint violation.

Equation (11) allows to keep the hypervolume of the ellipsoid constant before and after the modification of the covariance matrix in order to avoid premature convergence. The volume of the ellipsoid is reduced in the direction orthogonal to the constraints but is increased in the direction tangential to the constraints (Figure 5). The modified CMA-ES(λ, μ) algorithm for constraint handling is detailed in Algorithm 3.

Algorithm 3 Proposed modified CMA-ES(λ, μ) for constraint handling

```
1) Initialize the covariance matrix $\mathbf{C}^{(0)} = \mathbf{I}$, the step size $\sigma^{(0)}$ and the selection parameters [10]
2) Initialize the mean vector $\mathbf{m}^{(0)}$ to a random candidate, $k \leftarrow 0$
while CMA-ES convergence criterion is not reached do
    3-1) Generate $\lambda$ new offspring candidates according to: $\mathbf{z}_t^{(k+1)} \sim \mathbf{m}^{(k)} + \sigma^{(k)}\mathcal{N}(0, \mathbf{C}^{(k)})$, $t \in \{1, \ldots, \lambda\}$
    3-2) Evaluate candidates and sort them based on the objective function
    if all the $\mu$ best candidates are infeasible then
        Modify the covariance matrix according to Algorithm 2.
        Return to step 3.1)
    else
        If all the $\mu$ best candidates are feasible
        Determine the mean vector given the weightings of the $\mu$ best candidates: $\mathbf{m}^{(k+1)} = \sum_{i=1}^{\mu} w_i \mathbf{z}_{(i)}^{(k+1)}$
        Update covariance matrix $\mathbf{C}^{(k+1)}$ according to [10]
        end if
    else
        If at least one of the $\mu$ best candidates is infeasible and at least one is feasible
        Modify the covariance matrix according to Algorithm 2.
        Use the feasible candidates to determine the mean vector $\mathbf{m}^{(k+1)}$
        Use the feasible candidates to update covariance matrix $\mathbf{C}^{(k+1)}$ according to [10] end if
    end if
    3-3) Update the step size $\sigma^{(k+1)}$ according to [10], $k \leftarrow k + 1$
end while
4) return best candidate $\mathbf{z}_{best}$
```

The evolution of the ellipsoid between the generations (k) and (k + 1) if one of the μ best candidates violates a constraint is illustrated in Figure 5. The modification of $\mathbf{C}^{(k)}$ allows homothetic transformations in order to avoid to generate candidates in the non feasible zone.

If the mean vector $\mathbf{m}^{(k)}$ after the combination process is not feasible, instead of reducing the covariance matrix, the ellipsoid hypervolume is increased in order to generate candidates

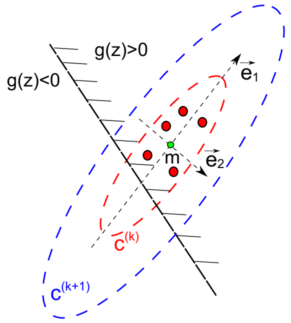


Figure 6. Modification of the covariance matrix due to the mean vector constraint violation, blue = feasible, red = unfeasible candidates.

in the feasible zone. Therefore, the ellipsoid hypervolume is increased according to:

$$\mathbf{S}^{(k)} \leftarrow \mathbf{P}\mathbf{D}^{(k)^2}\mathbf{P}^T + \gamma\mathbf{P}\mathrm{diag}\left(\sqrt{vp^{(k)}_{m_1}},\ldots,\sqrt{vp^{(k)}_{m_n}}\right)^2\mathbf{P}^T \quad (14)$$

The mean vector is displaced to the best feasible candidate generated at the next generation (Figure 6).

The modified CMA-ES(λ, μ) allows to take into account the constraints without degrading the objective function by penalization and avoids to tune the penalization parameters. Moreover, the proposed algorithm relies on the same update and selection mechanisms of the original CMA-ES(λ, μ) adapted for constraint handling and it keeps the invariance and unbiased design principles of CMA-ES(λ, μ) [10]. In the next sections, the proposed algorithm is tested on a benchmark of analytic functions and on the design of a two stage rocket in order to evaluate its performances.

4 Benchmark on analytic optimization problems

The proposed modified CMA-ES(λ, μ) is tested and compared to a penalized version of CMA-ES(λ, μ) with a constant penalization function, to the death penalty applied to CMA-ES(λ, μ) and to the modified (1+1)-CMA-ES on a benchmark of three analytic functions. The benchmark consists of a modified Six Hump Camel problem in 2 dimensions, the G04 optimization problem [8] in 5 dimensions and a modified Rosenbrock problem in 20 dimensions. These optimization problems are used in order to evaluate the proposed algorithm for different design space dimensions and different types of and numbers of inequality constraints (linear, non linear). In the following, the benchmark problems are introduced with the results. A discussion and a synthesis of the results for all the tests are provided in Section 4.4.

In the three problem formulations, the expected value is computed by Crude Monte Carlo method (CMC). A sample of 1000 points is used to estimate the expected value of the objective function. For each method (Modified CMA-ES(λ, μ), Death Penalty CMA-ES(λ, μ), Penalization CMA-ES(λ, μ), Modified (1+1)-CMA-ES) the optimization is repeated 50 times. The initialization is chosen randomly in the design space and the same initialization and the same random number seed are used for the four optimization algorithms. The same stopping criterion is used for all the algorithms: the distance in the design space between the mean vector and the best point found $\| \mathbf{m}^{(k)} - \mathbf{z}_{\mathbf{best}} \|^2 < 10^{-3}$ must be lower than a tolerance 20 times in a row.

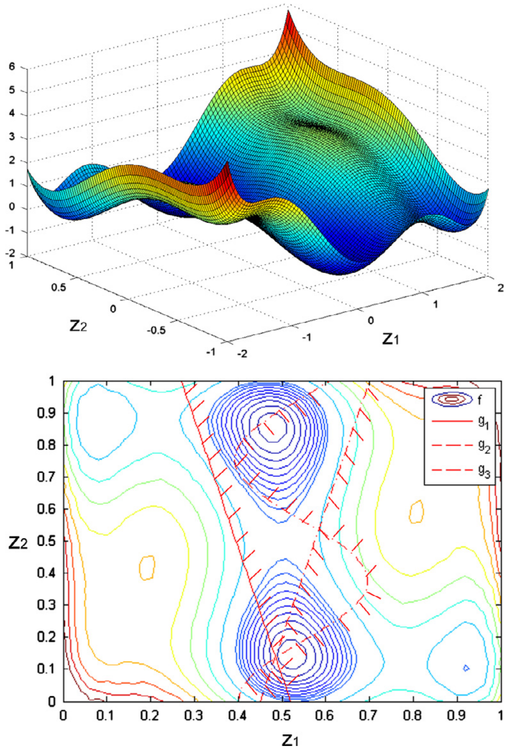


Figure 7. Modified Six Hump Camel function and constraints.

4.1 Modified Six Hump Camel problem

A modified version of the Six Hump Camel problem is used in order to introduce uncertainty and three inequality constraints. The formulation of the problem is the following:

$$\mathrm{Min} \quad \mathbb{E}\big[f_{6-\mathrm{hump}}(z_1, z_2) + f_{6-\mathrm{hump}}(z_1\cos(U) + z_2\sin(U), - z_1\sin(U) + z_2\cos(U))\big] \quad (15)$$

$$\mathrm{wrt} \quad \mathbf{z} = [z_1, z_2]$$

$$\mathrm{st} \quad g_1(z_1, z_2) = z_1 + z_2/4 - 0.52 \leq 0 \quad (16)$$

$$g_2(z_1, z_2) = z_1 + 0.01z_2 - 0.7 + 0.30\cos(60z_2^2/6) \leq 0 \quad (17)$$

$$g_3(z_1, z_2) = z_1 - z_2/4 - 0.45 \leq 0 \quad (18)$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \quad (19)$$

Table 1. Results of modified Six Hump Camel problem. Average over 50 optimizations (in parenthesis the Relative Standard Deviation (RSD) – $\sigma/\mathbb{E}$).

Results	Modified CMA-ES(λ, μ)	Death penalty CMA-ES(λ, μ)	Penalization CMA-ES(λ, μ)	Modified (1+1)-CMA-ES
Objective function (global optimum[1])	-1.875 (0.80%)	-1.875 (0.89%)	-1.874 (0.88%)	-1.865 (0.88%)
Objective function (local optimum[2])	-0.347 (0.75%)	-0.347 (0.84%)	-0.347 (0.79%)	-0.348 (0.85%)
Constraint functions	-4.72×10^{-4} (96%)	-2.57×10^{-4} (75%)	-6.43×10^{-4} (68%)	-4.86×10^{-4} (127%)
Number of objective function + constraint evaluations	1211 (13%)	1395 (12%)	1618 (12%)	781 (16%)
Percentage of convergence to the global minimum	37%	22%	33%	48%
Percentage of convergence to the local minimum	63%	78%	67%	52%

[1] Statistics based on the optimizations that converged to the global optimum.
[2] Statistics based on the optimizations that converged to the local optimum.

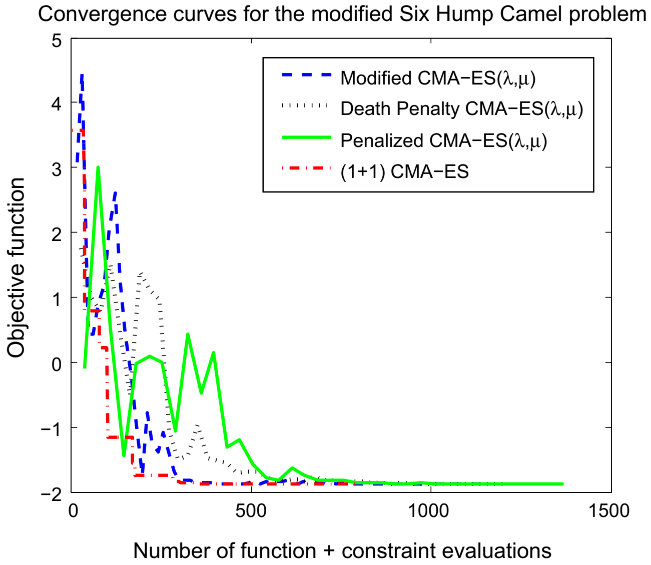


Figure 8. Convergence curves of the Six Hump Camel problem in 2 dimensions, based on one optimization run.

with $\mathbf{z} \in [-3, \quad 3] \times [-2, \quad 2]$, $f_{6-\text{hump}}(z_1, z_2) = (4 - 2.1z_1^2 + z_1^4/3) + z_1 z_2 + (4z_2^2 - 4)z_2^2$ and U a random variable distributed according to a normal distribution $U \sim \mathcal{N}(0, 0.05)$.

Representations of the function and the constraints are provided in Figure 7. The problem has one local optimum and one global optimum. The results are presented in Table 1 and the convergence curves for one optimization are given in Figure 8.

4.2 G04 optimization problem [8]

The G04 optimization problem involves 6 inequality constraints and is defined as following:

$$\text{Min} \quad G_{04}(\mathbf{z}) = 5.3578547 \times z_3^2 + 0.8356891 \times z_1 \times z_5 + 37.293239 \times z_1 - 40792.141 \qquad (20)$$

$$\text{wrt} \quad z = [z_1, z_2, z_3, z_4, z_5]$$

$$\text{st} \quad g_1(\mathbf{z}) = u(\mathbf{z}) - 92 \leq 0 \qquad (21)$$

$$g_2(\mathbf{z}) = -u(\mathbf{z}) \leq 0 \qquad (22)$$

$$g_3(\mathbf{z}) = v(\mathbf{z}) - 110 \leq 0 \qquad (23)$$

$$g_4(\mathbf{z}) = -v(\mathbf{z}) + 90 \leq 0 \qquad (24)$$

$$g_5(\mathbf{z}) = w(\mathbf{z}) - 25 \leq 0 \qquad (25)$$

$$g_6(\mathbf{z}) = -w(\mathbf{z}) + 20 \leq 0 \qquad (26)$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \qquad (27)$$

with $\mathbf{z} \in \mathbb{R}^5$, $\mathbf{z}_{\min} = [78, 33, 27, 27, 27]$, $\mathbf{z}_{\max} = [102, 45, 45, 45, 45]$ and:

$$u(\mathbf{z}) = 85.334407 + 0.0056858 \times z_2 \times z_5 + 0.0006262 \times z_1 \times z_4 - 0.0022053 \times z_3 \times z_5 \qquad (28)$$

$$v(\mathbf{z}) = 80.51249 + 0.0071317 \times z_2 \times z_5 + 0.0029955 \times z_1 \times z_2 + 0.0021813 \times z_3^2 \qquad (29)$$

$$w(\mathbf{z}) = 9.300961 + 0.0047026 \times z_3 \times z_5 + 0.0012547 \times z_1 \times z_3 + 0.0019085 \times z_3 \times z_4 \qquad (30)$$

The results are presented in Table 2 and the convergence curves for one optimization are given in Figure 9.

Table 2. Results of G04 optimization problem. Average over 50 optimizations (in parenthesis the RSD – $\sigma/\mathbb{E}$).

Results	Modified CMA-ES(λ, μ)	Death penalty CMA-ES(λ, μ)	Penalization CMA-ES(λ, μ)	Modified (1+1)-CMA-ES
Objective function	−30 487 (0.48%)	−28 105 (6.10%)	−28 758 (12%)	−30 452 (0.46%)
Number of objective function + constraint evaluations	1618 (33%)	9120 (40%)	13 139 (107%)	7048 (93%)

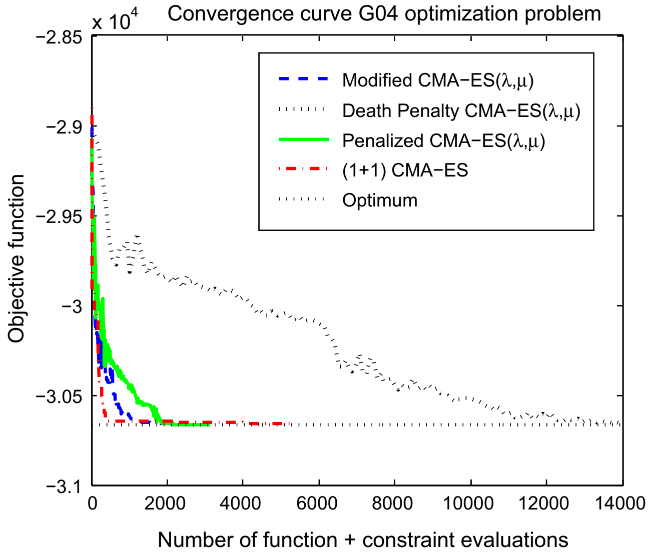


Figure 9. Convergence curves of the G04 optimization problem, based on one optimization run.

4.3 Modified Rosenbrock problem

The Rosenbrock optimization problem has been modified in order to incorporate uncertainty and an inequality constraint (Figure 10). The problem is formulated as following:

$$\text{Min} \quad \mathbb{E}\left[100\sum_{i=1}^{n-1}\left(z_{i+1}-z_i^2\right)^2+\sum_{i=1}^{n-1}\left(1-z_i\right)^2+U\right] \tag{31}$$

$$\text{wrt} \quad z=[z_1,\ldots,z_{20}]$$

$$\text{st} \quad g(\mathbf{z})=2-\prod_{i=1}^{n} z_i \leq 0 \tag{32}$$

with $n = 20$, $\mathbf{z} \in \mathbb{R}^{20}$ and U a random variable distributed according to $U \sim \mathcal{U}(-0.1, 0.0)$ a uniform distribution.

The results are presented in Table 3 and the convergence curves for one optimization are given in Figure 11.

4.4 Result and synthesis

The analytic test cases involve different dimensions (2, 5 and 20) and different number of constraints (1, 3 and 6) in order to evaluate the efficiency of the proposed modified CMA-ES(λ, μ) on various optimization problems. A qualitative synthesis of the obtained results is given in Figure 12. For all

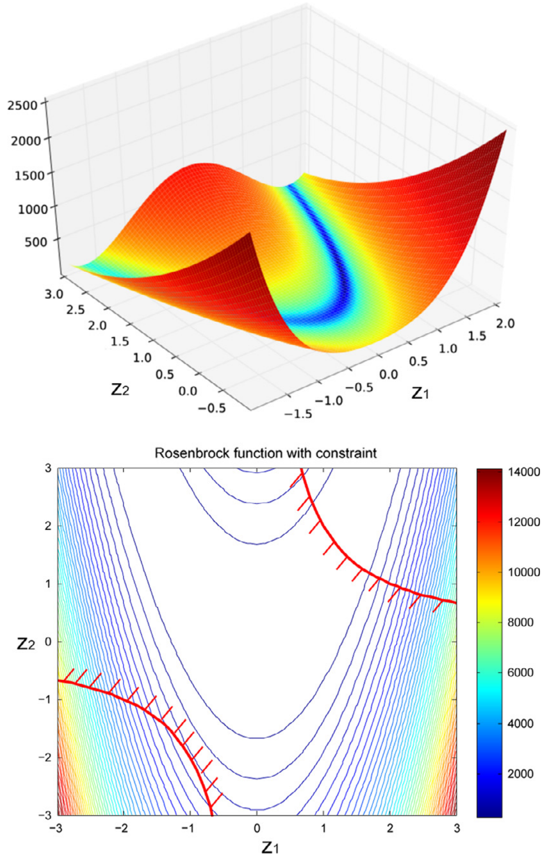


Figure 10. Modified Rosenbrock function and the constraints in 2 dimensions.

the three criteria (number of evaluations, robustness to initialization and value of the optimum), the lower value the better the quality of the method for the given criterion.

The Six Hump Camel problem has one local optimum and one global optimum. All the optimization algorithms converge either to the local or the global optimum. It illustrates the robustness property of the algorithms with respect to the initialization (relative standard deviation ~0.85% for all the algorithms). The found optima are all feasible. Modified (1+1)-CMA-ES converges in 48% of the optimization runs to the global optimum and the proposed modified CMA-ES (λ, μ) in 37% of the cases. The penalization and the death penality approaches converge only in 33% and 22% of the optimization runs to the global optimum. The number of calls to the objective function and the constraints is in increasing order: Modified (1+1)-CMA-ES (781), modified CMA-ES(λ, μ) (1211), Death Penalty CMA-ES(λ, μ) (1395) and Penalization

Table 3. Results of constrained Rosenbrock problem. Average over 50 optimizations (in parenthesis the RSD – $\sigma/\mathbb{E}$).

Results	Modified CMA-ES(λ, μ)	Death penalty CMA-ES(λ, μ)	Penalization CMA-ES(λ, μ)	Modified (1+1)-CMA-ES
Objective function	0.046 (0.12%)	0.109 (42.77%)	0.096 (26%)	0.0841 (53.85%)
Constraint functions	-5.05×10^{-4} (96%)	−0.74 (75%)	1.45 (68%)	−0.4294 (127%)
Number of objective function + constraint evaluations	12 798 (11%)	2708 (33%)	2689 (49%)	3481 (28%)

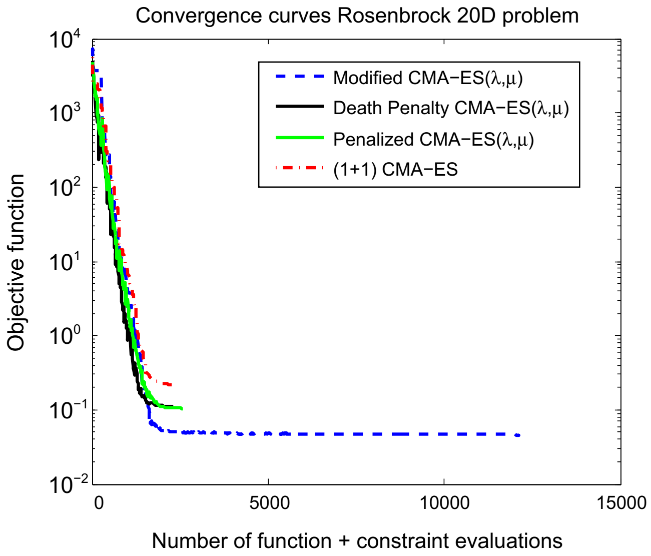


Figure 11. Convergence curves of the modified Rosenbrock problem in 20 dimensions, based on one optimization run.

CMA-ES(λ, μ) (1618). Modified (1+1)-CMA-ES is more efficient in this test case due to the low dimension and the simplicity of the optimization problem. The proposed modified CMA-ES(λ, μ) provides better results than the penalization and the death penalty approaches.

In the G04 problem, only the proposed modified CMA-ES(λ, μ) and the modified (1+1)-CMA-ES converge to the global minimum (with sufficient robustness with respect to the initialization). The number of calls to the objective function and the constraints is lower in the proposed algorithm (1618) compared to modified (1+1)-CMA-ES (7048) and the relative standard deviation is lower in the proposed approach. Moreover, the proposed approach converges efficiently to the global optimum. The Death Penalty and the penalization approaches do not succeed to reach the global optimum and are not robust to the initialization.

In the modified Rosenbrock problem, only the proposed modified CMA-ES(λ, μ) reaches the global optimum (with sufficient robustness (RSD: 0.12%) to the initialization). All the other algorithms are not robust to the initialization and do not converge to the global optimum. The number of calls to the objective function and the constraints is larger for the proposed approach (12 798) compared the other algorithms.

Consequently, from the benchmark, in small dimensions (<8), the modified (1+1)-CMA-ES provides good results in terms of convergence to the global optimum and robustness with respect to the initialization, however, as expected, in large dimensions, it presents issues to converge to the global optimum. The proposed modified CMA-ES(λ, μ) succeeds in small and large dimensions to find the global optimum. Moreover, this algorithm appears as robust to the initialization. In the next section, the proposed algorithm is used to design a two stage solid rocket and is compared to the existing CMA-ES based optimization algorithms.

5 Two stage solid propulsion rocket design

A multidisciplinary design problem consisting in maximizing the propulsive speed increment ΔV provided by a two stage rocket under geometrical and physical feasibility constraints is solved. The conceptual design models use simplified analysis of a two stage cylindrical solid propellant rocket motor. The multidisciplinary analysis involves four disciplines: the propulsion, the mass and sizing, the structure and the performance and constraint assessment (Figure 13). At the early design phase, model uncertainties exist and are taken into account. Two uncertainties are considered: the density of the propellant ρ and the ultimate strength σ for the rocket case material (Figure 14).

The problem is formulated as follows:

$$\text{Max} \quad \mathbb{E}[\Delta V(\mathbf{z}, \mathbf{U})] \tag{33}$$

$$\text{wrt} \quad \mathbf{z} = [D_{t1}, D_{s1}, P_{c1}, M_{p1}, D_{t2}, D_{s2}, P_{c2}, M_{p2}]$$

$$\text{st} \quad P_f[g_1(\mathbf{z}, \mathbf{U}) \geq 0] \leq 10^{-2} \tag{34}$$

$$P_f[g_2(\mathbf{z}, \mathbf{U}) \geq 0] \leq 10^{-2} \tag{35}$$

$$P_f[g_3(\mathbf{z}, \mathbf{U}) \geq 0] \leq 10^{-2} \tag{36}$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \tag{37}$$

with: $\mathbf{U} = [U_1, U_2]$ with $U_1 \sim \mathcal{N}(1, 0.02)$ the uncertainty of the density of the propellant (ρ) and $U_2 \sim \mathcal{N}(1, 0.05)$ the uncertainty of the ultimate strength limit (σ) for the rocket case material. The design variables are described in Table 4. An overview of the disciplines is detailed in the next paragraphs.

Propulsion. The propulsion discipline computes for a given set of propellant characteristics (density ρ, combustion speed, flame temperature, heat capacity ratio), the thrust T,

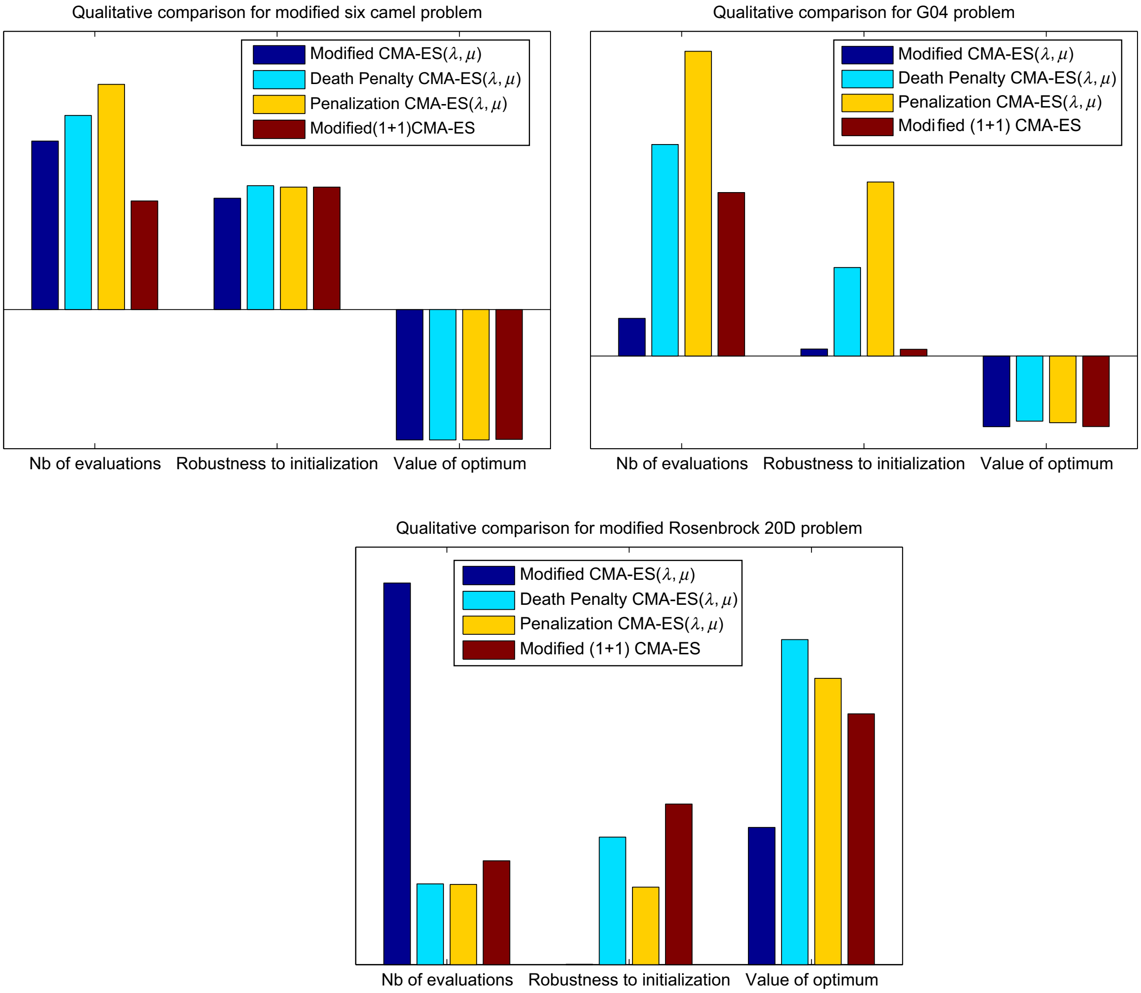


Figure 12. Qualitative results obtained for the different test cases.

the mass flow rate $\dot{m}$, the thrust coefficient c_T and the characteristic velocity $c*$, under the assumption of constant thrust. The discipline takes nozzle shapes Dt, D_s and combustion pressure P_c as inputs. The used propellant is the Butargols with polybutadiene binder without aluminium additive.

Mass and Sizing. The mass and sizing discipline computes the dry mass m_d and the geometry of the two stage solid propulsion rocket. The dry mass involves the mass of the rocket case and the mass of the nozzle and the pyrotechnic igniter. The rocket geometry consists of the initial combustion area, the packaging ratio and the size of the central channel. The overall dimensions (rocket length L = 22 m and diameter D = 1.07 m) are considered as fixed.

Structure. The structure discipline computes the tank walls thickness (t) which is sized under the combustion pressure based on the material characteristics (yield strength, ultimate strength limit) and rocket geometry. Moreover, it computes the stress in the rocket case.

Performance and constraints. The performance is the propulsive speed increment ΔV and the expected value of ΔV is the objective function to be maximized. CMC based on 1000 samples is used to compute the expected value of the propulsive speed increment. The three constraints are: $g_1(.)$ which ensures packaging ratio (Propellant volume/Available volume) that has to be inferior to 87%, $g_2(.)$ which ensures that central channel diameter is 30% greater than the nozzle throat diameter and $g_3(.)$ which ensures that combustion area is greater than the minimum feasible (area of central channel walls). The probabilities of failure for the three constraints have to be inferior to 1%. The probabilities of failure are computed with a CMC of 10^4 samples, numerically corresponding to a relative standard error ($\sigma/\mathbb{E}$) of the probability estimation in the order of 5%.

5.1 Results

The optimization for each algorithm is repeated 10 times. All the optimizations start from the same baseline given in Table 4. The same stopping criterion is used for all the optimization algorithms: the distance in the design space between the mean vector and the best point found

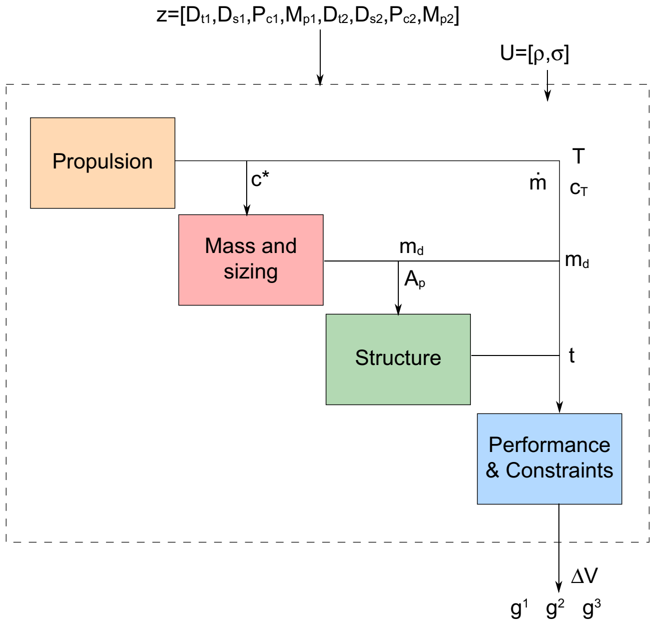


Figure 13. Design Structure Matrix for the two stage solid rocket.

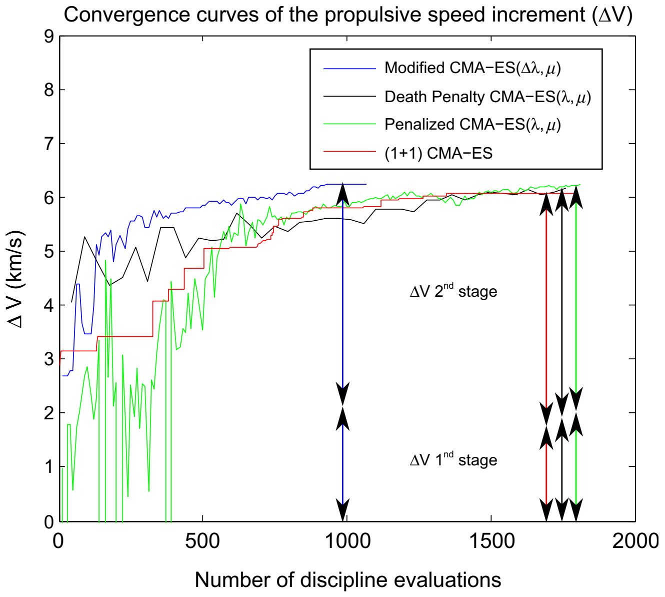


Figure 14. Convergence curves of the propulsive speed increment for the two stage solid propulsion rocket.

Table 4. Design variables for the two-stage rocket.

Variables	Symbol	Domain of definition	Baseline
1st stage nozzle throat diameter	D_{t1}	[0.05, 1] m	0.75 m
2nd stage nozzle throat diameter	D_{t2}	[0.05, 1] m	0.75 m
1st stage nozzle exit diameter	D_{s1}	[0.5, 1.4] m	1 m
2nd stage nozzle exit diameter	D_{s2}	[0.5, 1.4] m	1 m
1st stage combustion pressure	P_{c1}	[1, 500] bar	100 bar
2nd stage combustion pressure	P_{c2}	[1, 500] bar	100 bar
1st stage propellant mass	M_{p1}	[2000, 15 000] kg	8000 kg
2nd stage propellant mass	M_{p1}	[2000, 15 000] kg	8000 kg

Table 5. Results of the two stage rocket optimization. Average over 10 optimizations (in parenthesis the RSD – $\sigma/\mathbb{E}$).

Results	Modified CMA-ES(λ, μ)	Death penalty CMA-ES(λ, μ)	Penalization CMA-ES(λ, μ)	Modified (1+1)-CMA-ES
Design variables	[0.321, 0.9081, 205.46, 12 428.1, 0.637, 1.288, 21.08, 7911.2]	[0.449, 1.240, 64.12, 10 781.4, 0.535, 1.112, 32 785, 9726.6]	[0.283, 0.7131, 181.83, 12 462.5, 0.571, 1.39, 26.20, 9316.9]	[0.3558, 0.9961, 199.87, 11 750.2, 0.5476, 1.314, 31.78, 9723.6]
Objective function	6234.1 (0.95%)	6093.2 (1.55%)	6097.4 (1.43%)	6080.1 (1.19%)
Constraint functions	[−0.01, −0.0023,−0.0014, −0.01, −0.0023,−0.0032]	[−0.01, −0.025, −0.01, −0.01, −0.01, −0.0037]	[−0.01, −0.0079, −0.0053, −0.01, −0.0054, −0.0067]	[−0.01, −0.0086, −0.0037, −0.01, −0.0028, −0.0046]
Number of discipline evaluations	1127 (0.73%)	1758 (0.88%)	1813 (1.54%)	1675 (0.94%)

$||\mathbf{m}^{(k)} - \mathbf{z}_{\text{best}}|| < 10^{-3}$ must be under a tolerance 20 times in a row. The algorithms do not converge to the same optimum. Modified CMA-ES (λ, μ) provides a better optimum in terms of propulsive speed increment: 6234.1 m/s with a better robustness. The proposed algorithm converges on the average in 1127 discipline evaluations. The other optimization algorithms converge in the same order of number of discipline evaluations (~1750). Only four constraints are active at the optimum. The better optimum found by the proposed approach is essential as it has a better propulsive speed increment which could be used to increase the payload mass (Table 5).

6 Conclusion

The design of complex systems often induces a constrained optimization problem under uncertainty. In this paper, an adaptation of CMA-ES(λ, μ) has been proposed in order to efficiently handle the constraints. The probable research hypervolume engendered by an iso-probability contour of the multivariate normal distribution $\mathcal{N}(0, \mathbf{C})$ used to generate the candidate population is modified. The constraint handling method allows to reduce the semi-principal axes of the iso-probable research ellipsoid in the directions violating the constraints by decreasing the eigenvalues of the covariance matrix $\mathbf{C}$. The proposed approach has been tested with three analytic optimization problems highlighting the efficiency of the algorithm and the robustness with respect to the initialization. The proposed method has been used to design a two stage solid propulsion launch vehicle. A better optimum has been found with the proposed approach with respect to the existing CMA-ES based optimization algorithms resulting in a potential increase in the payload mass.

Acknowledgements. The work of L. Brevault is part of a CNES/ONERA PhD thesis.

References

1. Arnold DV, Hansen N. 2012. A (1+1)-CMA-ES for constrained optimisation, in Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference, New York, NY, USA, p. 297–304, ACM.
2. Beyer H-G, Finck S. 2012. On the design of constraint Covariance Matrix Self-Adaptation Evolution Strategies including a cardinality constraint. IEEE Transactions on Evolutionary Computation, 16(4), 578–596.
3. Branke J, Schmidt C. 2003. Selection in the presence of noise, in Genetic and Evolutionary Computation Conference, Chicago, IL, USA.
4. Coello Coello CA. 2000. Constraint-handling using an evolutionary multiobjective optimization technique. Civil Engineering Systems, 17(4), 319–346.
5. Collange G, Reynaud S, Hansen N. 2010. Covariance Matrix Adaptation Evolution Strategy for multidisciplinary optimization of expendable launcher family, in 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, Fort-Worth, TX.
6. De Melo VV, Iacca G. 2014. A modified Covariance Matrix Adaptation Evolution Strategy with adaptive penalty function and restart for constrained optimization. Expert Systems with Applications, 41(16), 7077–7094.
7. Eberhart RC, Kennedy J. 1995. A new optimizer using particle swarm theory. Proceedings of the sixth international symposium on micro machine and human science, New York, NY, p. 39–43.
8. Hansen N. 2009. Benchmarking a BI-population CMA-ES on the BBOB-2009 noisy testbed, in Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers, Montreal, Qubec, Canada, p. 2397–2402, ACM.
9. Hansen N, Auger A, Ros R, Finck S, Pošík P. 2010. 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, p. 1689–1696, ACM.
10. Hansen N, Müller S, Koumoutsakos P. 2003. Reducing the time complexity of the derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). Evolutionary Computation, 11(1), 1–18.
11. Hansen N, Niederberger AS, Guzzella L, Koumoutsakos P. 2009. A method for handling uncertainty in evolutionary optimization with an application to feedback control of combustion. IEEE Transactions on Evolutionary Computation, 13(1), 180–197.
12. Holland JH. 1975. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. University of Michigan, USA.
13. Karaboga D. 2005. An idea based on honey bee swarm for numerical optimization, Technical report, Technical report-tr06, Erciyes University, engineering faculty, computer engineering department.
14. Kramer O, Barthelmes A, Rudolph G. 2009. Surrogate constraint functions for CMA Evolution Strategies, in KI 2009: Advances in Artificial Intelligence, Springer, Verlag Berlin Heidelberg, p. 169–176.
15. Larson JM. 2012. Derivative free optimization of noisy functions. PhD thesis, University of Colorado.
16. Mezura-Montes E, Coello Coello CA. 2011. Constraint-handling in nature-inspired numerical optimization: past, present and future. Swarm and Evolutionary Computation, 1(4), 173–194.
17. Price K, Storn RM, Lampinen JA. 2006. Differential evolution: a practical approach to global optimization, Springer, Verlag Berlin Heidelberg.
18. Rowell LF, Korte JJ. 2003. Launch vehicle design and optimization methods and priority for the advanced engineering environment. NASA Technical Report NASA/TM-2003, 212654.
19. Schwefel H-P, Rudolph G. 1995. Contemporary evolution strategies, Springer, Verlag Berlin Heidelberg.

Cite this article as: Chocat R, Brevault L, Balesdent M & Defoort S: Modified Covariance Matrix Adaptation – Evolution Strategy algorithm for constrained optimization under uncertainty, application to rocket design. Int. J. Simul. Multisci. Des. Optim., 2015, **6**, A1.