

Recombination for Learning Strategy Parameters in the MO-CMA-ES

Thomas Voß¹ and Nikolaus Hansen² and Christian Igel¹

¹ {thomas.voss,christian.igel}@neuroinformatik.rub.de

Institut für Neuroinformatik

Ruhr-Universität Bochum

44780 Bochum, Germany

² hansen@lri.fr

Université de Paris-Sud

Centre de recherche INRIA Saclay – Île-de-France,

F-91405 Orsay Cedex, France

Abstract. The multi-objective covariance matrix adaptation evolution strategy (MO-CMA-ES) is a variable-metric algorithm for real-valued vector optimization. It maintains a parent population of candidate solutions, which are varied by additive, zero-mean Gaussian mutations. Each individual learns its own covariance matrix for the mutation distribution considering only its parent and offspring. However, the optimal mutation distribution of individuals that are close in decision space are likely to be similar if we presume some notion of continuity of the optimization problem. Therefore, we propose a lateral (inter-individual) transfer of information in the MO-CMA-ES considering also successful mutations of *neighboring individuals* for the covariance matrix adaptation. We evaluate this idea on common bi-criteria objective functions. The preliminary results show that the new adaptation rule significantly improves the performance of the MO-CMA-ES.

1 Introduction

The multi-objective covariance matrix adaptation evolution strategy (MO-CMA-ES) recently presented in [1,2,3] extends the single-objective CMA-ES [4,5,6] to real-valued vector optimization. The algorithm in [2] considers a population of individuals subject to multi-objective, indicator-based selection. Each of the individuals adapts its own variable-metric for generating offspring. Up until now, the update of the strategy parameters, that is, the covariance matrix and a global step-size parameter, considers only information within the genealogical tree of each individual. This work presents a new covariance matrix update procedure that incorporates information from multiple genealogies. The performance of the modified MO-CMA-ES with the enhanced adaptation scheme is empirically evaluated and compared to the performance of the original MO-CMA-ES.

The remainder of this work is organized as follows. Section two briefly describes the original MO-CMA-ES. In Section three, the new covariance matrix

adaptation procedure is presented. The empirical evaluation is summarized in section four. We close with the final conclusions and suggestions for future research directions.

2 The MO-CMA-ES

In the following, we briefly outline the MO-CMA-ES according to [1,2]. For a detailed description and a performance evaluation on bi-objective benchmark functions we refer to [1] (see also [2,3]).

The MO-CMA-ES relies on the non-dominated sorting selection scheme [7]. As in the SMS-EMOA [8], the hypervolume-indicator serves as second-level sorting criterion to rank individuals at the same *level of non-dominance*. In the following, we first describe the general ranking procedure and then summarize the other parts of the MO-CMA-ES.

Let A be a population, and let a, a' be two individuals in A . Let the non-dominated solutions in A be denoted by $\text{ndom}(A) = \{a \in A \mid \nexists a' \in A : a' \prec a\}$, where \prec denotes the Pareto-dominance relation. The elements in $\text{ndom}(A)$ are assigned a level of non-dominance of 1. The other ranks of non-dominance are defined recursively by considering the set A without the solutions with lower ranks [7]. Formally, let $\text{dom}_0(A) = A$, $\text{dom}_l(A) = \text{dom}_{l-1}(A) \setminus \text{ndom}_l(A)$, and $\text{ndom}_l(A) = \text{ndom}(\text{dom}_{l-1}(A))$ for $l \geq 1$. For $a \in A$ we define the level of non-dominance $\text{rank}(a, A)$ to be i iff $a \in \text{ndom}_i(A)$.

The hypervolume measure or \mathcal{S} -metric was introduced in [9] in the domain of evolutionary MOO. It can be defined as the Lebesgue measure Λ (i.e., the volume) of the union of hypercuboids in the objective space [10]:

$$\mathcal{S}_{a_{\text{ref}}}(A') = \Lambda \left(\bigcup_{a \in \text{ndom}(A')} \{(f_1(a'), \dots, f_M(a')) \mid a \prec a' \prec a_{\text{ref}}\} \right), \quad (1)$$

where a_{ref} is an appropriately chosen reference point. The contributing hypervolume of a point $a \in A' = \text{ndom}(A)$ is given by

$$\Delta_{\mathcal{S}}(a, A') = \mathcal{S}_{a_{\text{ref}}}(A') - \mathcal{S}_{a_{\text{ref}}}(A' \setminus \{a\}) . \quad (2)$$

Now we define the *contribution rank* $\text{cont}(a, A')$ of a . This is again done recursively. The element, say a , with the largest contributing hypervolume is assigned contribution rank 1. The next rank is assigned by considering $A' \setminus \{a\}$ etc. More precisely, let $c_1(A') = \text{argmax}_{a \in A'} \Delta_{\mathcal{S}}(a, A')$ and

$$c_i(A') = c_1 \left(A' \setminus \bigcup_{j=1}^{i-1} \{c_j(A')\} \right) \quad (3)$$

for $i > 1$. For $a \in A'$ we define the contribution rank $\text{cont}(a, A')$ to be i iff $a = c_i(A')$. In the ranking procedure ties are broken at random.

Finally, the following relation between individuals $a, a' \in A$ is defined:

$$a \prec_A a' \Leftrightarrow \text{rank}(a, A) < \text{rank}(a', A) \vee \left[\text{rank}(a, A) = \text{rank}(a', A) \wedge \text{cont}(a, \text{ndom}(A)) < \text{cont}(a', \text{ndom}(A)) \right] \quad (4)$$

In the $\mu_{\text{MO}} \times (1+1)$ -MO-CMA-ES, a candidate solution $a_i^{(g)}$, $i \in \{1, \dots, \mu_{\text{MO}}\}$, in generation g is a tuple $\left[\mathbf{x}_i^{(g)}, \bar{p}_{\text{succ},i}^{(g)}, \sigma_i^{(g)}, \mathbf{p}_{i,c}^{(g)}, \mathbf{C}_i^{(g)} \right]$, where

- $\mathbf{x}_i^{(g)}$ is the current search point,
- $\bar{p}_{\text{succ},i}^{(g)}$ is the smoothed success probability,
- $\sigma_i^{(g)}$ is the global step-size,
- $\mathbf{p}_{i,c}^{(g)}$ is the cumulative evolution path,
- $\mathbf{C}_i^{(g)}$ is the covariance matrix of the search distribution.

The standard version of the $\mu_{\text{MO}} \times (1+1)$ -MO-CMA-ES is given in Algorithm 1. The indicator function $\mathbb{I}(\cdot)$ evaluates to one if its argument is true and to zero otherwise.

Algorithm 1: $\mu_{\text{MO}} \times (1+1)$ -MO-CMA-ES

```

1  $g \leftarrow 0$ , initialize  $a_k^{(g)}$  for  $k \in \{1, \dots, \mu_{\text{MO}}\}$ 
2 repeat
3   for  $k = 1, \dots, \mu_{\text{MO}}$  do
4      $a_k^{(g+1)} \leftarrow a_k^{(g)}$ 
5      $\mathbf{x}_k^{(g+1)} \sim \mathbf{x}_k^{(g)} + \sigma_k^{(g)} \mathcal{N}(\mathbf{0}, \mathbf{C}_k^{(g)})$ 
6      $Q^{(g)} \leftarrow \{a_k^{(g+1)}, a_k^{(g)}\}$ 
7     for  $k = 1, \dots, \mu_{\text{MO}}$  do
8        $\sigma$ -update  $\left( a_k^{(g+1)}, \mathbb{I}(\mathbf{x}_k^{(g+1)} \prec_{Q^{(g)}} \mathbf{x}_k^{(g)}) \right)$ 
9       rank-one-update  $\left( a_k^{(g+1)}, \frac{\mathbf{x}_k^{(g+1)} - \mathbf{x}_k^{(g)}}{\sigma_k^{(g)}} \right)$ 
10       $\sigma$ -update  $\left( a_k^{(g)}, \mathbb{I}(\mathbf{x}_k^{(g+1)} \prec_{Q^{(g)}} \mathbf{x}_k^{(g)}) \right)$ 
11    for  $i \in \{1, \dots, \mu_{\text{MO}}\}$  do
12       $a_i^{(g+1)} \leftarrow Q_{\prec:i}^{(g)}$ 
13 until stopping criterion is met

```

For each of the μ_{MO} individuals, one offspring is sampled (lines 3–5). The decision whether a new candidate solution is better than its parent is made in the context of the population $Q^{(g)}$ of parent and offspring individuals due to the indicator-based selection strategy implemented in the algorithm. The covariance matrix of each offspring is adapted (line 9, see the procedure **rank-one-update**).

Subsequently, the step-sizes $\sigma_k^{(g)}$ and $\sigma_k'^{(g+1)}$ of parent and offspring individuals $a_k^{(g)}$ and $a_k^{(g+1)}$ are updated (line 8 and 10, see the procedure σ -update). Finally, the new parent population is selected from the set of parent and offspring individuals according to the indicator-based selection scheme (lines 11–12). Here, $Q_{\prec, i}^{(g)}$ denotes the i th best individual in $Q^{(g)}$ ranked by non-dominated sorting and the contributing hypervolume according to (4) (see also [1]).

Procedure σ -update($a = [\mathbf{x}, \bar{p}_{\text{succ}}, \sigma, \mathbf{p}_c, \mathbf{C}]$, p_{succ})

- 1 $\bar{p}_{\text{succ}} \leftarrow (1 - c_p)\bar{p}_{\text{succ}} + c_p p_{\text{succ}}$
 - 2 $\sigma \leftarrow \sigma \exp\left(\frac{1}{d} \frac{\bar{p}_{\text{succ}} - p_{\text{succ}}^{\text{target}}}{1 - p_{\text{succ}}^{\text{target}}}\right)$
-

The (external) strategy parameters are the population size, initial global step size, target success probability $p_{\text{succ}}^{\text{target}}$, step-size damping d , success rate averaging parameter c_p , cumulation time horizon parameter c_c , and covariance matrix learning rate c_{cov} . Default values as given in [1] and used in this paper are: $d = 1 + n/2$, $p_{\text{succ}}^{\text{target}} = (5 + \sqrt{1/2})^{-1}$, $c_p = p_{\text{succ}}^{\text{target}} / (2 + p_{\text{succ}}^{\text{target}})$, $c_c = 2 / (n + 2)$, $c_{\text{cov}} = 2 / (n^2 + 6)$ and $p_{\text{thresh}} = 0.44$. The initial global step sizes $\sigma_i^{(0)}$ are set dependent on the problem (e.g., in the case of box constraints, see below, with $x_i^u - x_i^l = x_j^u - x_j^l$ for $1 \leq i, j \leq n$ to $0.6 \cdot (x_1^u - x_1^l)$).

Procedure rank-one-update($a = [\mathbf{x}, \bar{p}_{\text{succ}}, \sigma, \mathbf{p}_c, \mathbf{C}]$, $\mathbf{z} \in \mathbb{R}^n$)

- 1 **if** $\bar{p}_{\text{succ}} < p_{\text{thresh}}$ **then**
 - 2 $\mathbf{p}_c \leftarrow (1 - c_c)\mathbf{p}_c + \sqrt{c_c(2 - c_c)}\mathbf{z}$
 - 3 $\mathbf{C} \leftarrow (1 - c_{\text{cov}})\mathbf{C} + c_{\text{cov}}\mathbf{p}_c\mathbf{p}_c^T$
 - 4 **else**
 - 5 $\mathbf{p}_c \leftarrow (1 - c_c)\mathbf{p}_c$
 - 6 $\mathbf{C} \leftarrow (1 - c_{\text{cov}})\mathbf{C} + c_{\text{cov}}(\mathbf{p}_c\mathbf{p}_c^T + c_c(2 - c_c)\mathbf{C})$
-

When in this study the MO-CMA-ES is applied to a benchmark problem \mathbf{f} with box constraints, we consider a penalized fitness function

$$\mathbf{f}^{\text{penalty}}(\mathbf{x}) = \mathbf{f}(\text{feasible}(\mathbf{x})) + \alpha \|\mathbf{x} - \text{feasible}(\mathbf{x})\|_2^2 \quad (5)$$

in the search process, where

$$\text{feasible}(\mathbf{x}) = (\min(\max(x_1, x_1^l), x_1^u), \dots, \min(\max(x_n, x_n^l), x_n^u))^T \quad (6)$$

and x_i^l and x_i^u are the lower and upper bound of the i th component of the search space. We set ad-hoc $\alpha = 10^{-6}$.

3 A New Covariance Matrix Update

The recombination of information provided by candidate solutions is a powerful variation operator that is present in most current single- and multi-objective evolutionary algorithms. Currently, the MO-CMA-ES as proposed in [1] lacks this feature. This section introduces a method for recombining neighbouring individuals to further speed up the strategy parameter adaptation in the MO-CMA-ES.

3.1 Incorporation of Information from Successful Offspring

The basic idea is that appropriate covariance matrices (i.e., appropriate coordinate systems) are similar for individuals that are in the same region of the decision space if we presume some notion of continuity of the objective function (more precisely, we presume that the principle of *strong causality* [11] is not too often violated). Thus, combining information about the topology of the search space gathered by neighbouring individuals is expected to speed up the learning of the covariance matrix. In the following, we realize this idea in the MO-CMA-ES.

Consider the set of parent individuals $P^{(g)}$ and the set of newly generated candidate solutions $Q^{(g)}$ in generation g . Let $Q'^{(g)} \subseteq Q^{(g)}$ be the set of successful offspring (i.e., $Q'^{(g)} \subseteq P^{(g+1)}$). The covariance matrix of each individual in $Q'^{(g)}$ is updated. The standard $\mu_{\text{MO}} \times (1+1)$ -MO-CMA-ES relies on the “isolated” rank-one-update with cumulative evolution path that solely exploits the step from the parent to its offspring.

Let $a_i^{(g)}$ and $a_i'^{(g+1)}$ be an individual and its offspring, respectively. Let us assume that $a_i'^{(g+1)}$ is successful and therefore $a_i'^{(g+1)} \in Q'^{(g)}$. The rank-one-update of the covariance matrix of $a_i'^{(g+1)}$ is given by

$$\mathbf{C}_i'^{(g+1)} = (1 - c_{\text{cov}})\mathbf{C}_i^{(g)} + c_{\text{cov}}\mathbf{p}_i'^{(g+1)} \left(\mathbf{p}_i'^{(g+1)}\right)^{\text{T}}, \quad (7)$$

where $\mathbf{p}_i'^{(g+1)}$ is the updated cumulative evolution path of $a_i'^{(g+1)}$. Our modification of the adaptation method reads:

$$\begin{aligned} \mathbf{C}_i'^{(g+1)} = (1 - c_{\text{cov}}) & \left[\left(1 - \sum_{j=1}^{\mu_{\text{MO}}} w_{ij}^{(g+1)} \right) \mathbf{C}_i^{(g)} \right. \\ & \left. + \underbrace{\sum_{j=1}^{\mu_{\text{MO}}} w_{ij}^{(g+1)} \frac{\mathbf{x}_j'^{(g+1)} - \mathbf{x}_j^{(g)}}{\sigma_j^{(g)}} \left(\frac{\mathbf{x}_j'^{(g+1)} - \mathbf{x}_j^{(g)}}{\sigma_j^{(g)}} \right)^{\text{T}}}_{=\mathbf{Z}^{(g+1)}} \right] + \underbrace{c_{\text{cov}}\mathbf{p}_i'^{(g+1)}\mathbf{p}_i'^{(g+1)\text{T}}}_{\text{rank-one-update}}, \end{aligned} \quad (8)$$

Here $w_{ij}^{(g+1)}$ is a weighting coefficient assigned to the j -th offspring individual $a_j'^{(g+1)}$. The weight is calculated anew in each generation. It is different for each

individual in the offspring population. If the individual is not selected for the next parent generation, it is assigned a weight $w_{ij}^{(g+1)} = 0$.

The matrix $\mathbf{Z}^{(g+1)}$ aggregates information from the selected new candidate solutions and is of rank $\min\{\mu_{\text{MO,succ}}, n\}$ with probability one, where $\mu_{\text{MO,succ}}$ denotes the number of successful offspring. Thus, the new adaptation method is referred to as rank- $\mu_{\text{MO,succ}}$ -update.

In contrast to the combination of the rank- μ - and rank-one-update in the non-elitist CMA-ES (see [4]), no constant μ_{cov} is used to balance the impact of the two different update rules. The blending of old information $\mathbf{C}^{(g)}$ and new information $\mathbf{Z}^{(g+1)}$ is controlled by considering the sum of weighting coefficients.

3.2 Weighting of Neighbouring Individuals

The following assumptions underlie the calculation of the weighting coefficients:

- Non-successful offspring individuals do not represent promising sampling directions and are assigned a weight of zero.
- Information contributed by individuals that are closer to the individual to be updated is more important, as the chance for a similar topology of the search space is higher for individuals nearby.
- Distances between individuals should be measured in terms of the metric learnt by the individual to be updated.

Now we take the point of view of an offspring individual $a_i^{(g+1)}$, $i \in \{1, \dots, \mu_{\text{MO}}\}$, whose covariance matrix $\mathbf{C}'_i^{(g+1)}$ needs to be updated. A weighting coefficient $w_{ij}^{(g+1)}$ for each offspring $a_j^{(g+1)}$, $j \in \{1, \dots, \mu_{\text{MO}}\}$ is determined. The weight reflects the relevance of $a_j^{(g+1)}$ for the covariance matrix update of $a_i^{(g+1)}$. The importance of $a_j^{(g+1)}$ depends on its distance to $a_i^{(g+1)}$. The closer $a_j^{(g+1)}$ is situated to $a_i^{(g+1)}$, the higher the weight it is assigned. An individual $a_j^{(g+1)}$ is considered near in the search space if $a_i^{(g+1)}$ can reach $a_j^{(g+1)}$ by a small number of mutative steps. That is, the individual $a_j^{(g+1)}$ is close to $a_i^{(g+1)}$ if the probability to sample a point close to $a_j^{(g+1)}$ is high according to the search distribution

$$\mathcal{N}\left(\mathbf{x}'_i^{(g+1)}, \sigma_i'^{(g+1)2} \mathbf{C}'_i^{(g+1)}\right), \quad (9)$$

of the individual $a_i^{(g+1)}$.

Accordingly, the distance calculation is carried out w.r.t. the shape of the search distribution and the step-size of $a_i^{(g+1)}$ using the Mahalanobis distance based on the covariance matrix of the individual to be updated (9):

$$d_{\text{M}}\left(a_i^{(g+1)}, a_j^{(g+1)}\right) = \frac{\sqrt{\left(\mathbf{x}'_i^{(g+1)} - \mathbf{x}'_j^{(g+1)}\right)^T \mathbf{C}'_i^{(g+1)-1} \left(\mathbf{x}'_i^{(g+1)} - \mathbf{x}'_j^{(g+1)}\right)}}{\sigma_i'^{(g+1)}} \quad (10)$$

Note that $d_M \left(a_i^{(g+1)}, a_j^{(g+1)} \right)$ is not symmetric as the difference vector $\mathbf{x}_i^{(g+1)} - \mathbf{x}_j^{(g+1)}$ is transformed into the coordinate system of $a_i^{(g+1)}$ by multiplying with the inverse of $\sigma_i^{(g+1)2} \mathbf{C}_i^{(g+1)}$.

The (MO-)CMA-ES explores the search space by means of mutative steps. Therefore, we normalize the distance w.r.t. this “unit of measurement”. The scaling of the Euclidean norm of an $\mathcal{N}(\mathbf{0}, \mathbf{I})$ -distributed random vector with the dimension of the search space needs to be addressed to render the distance calculation independent of the search space dimension n . To this end, the expected length of an $\mathcal{N}(\mathbf{0}, \mathbf{I})$ -distributed is approximated by

$$\mathbb{E}(\|\mathcal{N}(\mathbf{0}, \mathbf{I})\|_2) = \sqrt{n} + \mathcal{O}(1/n) \approx \sqrt{n} . \quad (11)$$

Thus, the comparison of the distance $d_M \left(a_i^{(g+1)}, a_j^{(g+1)} \right)$ with the expected length $\mathbb{E}(\|\mathcal{N}(\mathbf{0}, \mathbf{I})\|_2)$ corresponds approximately to a division by \sqrt{n} . This can be viewed as a normalization by the unit of measurement “reachable in *one* mutative step”. Now this normalization is extended to d_{steps} mutative steps. Basically, the distance $d_M \left(a_i^{(g+1)}, a_j^{(g+1)} \right)$ is compared to the expected length of a random vector $\mathbf{x} = \sum_{k=1}^{d_{\text{steps}}} \mathbf{x}_k$, where $\mathbf{x}_1, \dots, \mathbf{x}_{d_{\text{steps}}}$ are independently $\mathcal{N}(\mathbf{0}, \mathbf{I})$ -distributed. The variance of \mathbf{x} is equal to the sum of variances of the independent steps \mathbf{x}_k , $k \in \{1, \dots, d_{\text{steps}}\}$,

$$\mathbf{x} = \sum_{k=1}^{d_{\text{steps}}} \mathbf{x}_k \sim \mathcal{N}(\mathbf{0}, d_{\text{steps}} \mathbf{I}) \sim \sqrt{d_{\text{steps}}} \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (12)$$

and therefore the expected length of \mathbf{x} distributed according to $\mathcal{N}(\mathbf{0}, d_{\text{steps}} \mathbf{I})$ is

$$\mathbb{E}(\|\mathcal{N}(\mathbf{0}, d_{\text{steps}} \mathbf{I})\|_2) = \sqrt{d_{\text{steps}}} \mathbb{E}(\|\mathcal{N}(\mathbf{0}, \mathbf{I})\|_2) \approx \sqrt{d_{\text{steps}}} n . \quad (13)$$

We determine the weights based on this distance measure. A weight $w_{ij}^{(g+1)}$ is computed by

$$w_{ij}^{(g+1)} = w'_{ij}{}^{(g+1)} \min \left\{ 1, \frac{2\mu_{\text{MO,eff}} - 1}{(n+2)^2 + \mu_{\text{MO,eff}}} \right\} \quad (14)$$

using the intermediate weights

$$w''_{ij}{}^{(g+1)} = \begin{cases} h \left(d_M \left(a_i^{(g+1)}, a_j^{(g+1)} \right) / \sqrt{d_{\text{steps}} n} \right) & \text{if } a_j^{(g+1)} \prec_{Q(g)} a_i^{(g+1)} \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

$$w'_{ij}{}^{(g+1)} = \frac{w''_{ij}{}^{(g+1)}}{\mu_{\text{MO}} - \mu_{\text{MO,succ}} + \sum_{k=1}^{\mu_{\text{MO}}} w''_{ik}{}^{(g+1)}} . \quad (16)$$

Only successful offspring individuals shall be considered in the update, and therefore an intermediate weight of zero is assigned to non-successful offspring. For

the others, the intermediate weight $w''_{ij}{}^{(g+1)}$ is calculated by applying a monotonically decreasing *distance weighting function* $h : \mathbb{R}^{\geq 0} \rightarrow \mathbb{R}$ to the distance $d_M(a'_i{}^{(g+1)}, a'_j{}^{(g+1)}) / \sqrt{d_{\text{steps}} n}$. Here, $h(\cdot)$ has been chosen as

$$h : \mathbb{R}^{\geq 0} \rightarrow \mathbb{R}, \quad x \mapsto e^{-x} . \quad (17)$$

Thus, the neighbourhood of $a'_i{}^{(g+1)}$ that is considered important for the covariance matrix update depends “smoothly” on the distance measured by d_M . Our goal is to fuse the information encoded in the covariance matrix $\mathbf{C}'_i{}^{(g+1)}$ and in the matrix $\mathbf{Z}^{(g+1)}$, which contains information about successful steps:

$$\mathbf{Z}^{(g+1)} = \sum_{j=1}^{\mu_{\text{MO}}} w'_{ij}{}^{(g+1)} \frac{\mathbf{x}'_j{}^{(g+1)} - \mathbf{x}_j^{(g)}}{\sigma_j^{(g)}} \left(\frac{\mathbf{x}'_j{}^{(g+1)} - \mathbf{x}_j^{(g)}}{\sigma_j^{(g)}} \right)^{\text{T}} \quad (18)$$

The sum of all final weights determines how much emphasis we put on $\mathbf{Z}^{(g+1)}$ in the covariance matrix update (8). The “more information” is contained in $\mathbf{Z}^{(g+1)}$ the larger the sum of the final weights can be. To account for that, we first normalize the intermediate weights $w''_{ij}{}^{(g+1)}$ by the number of successful offspring individuals $\mu_{\text{MO},\text{succ}}$ according to Eq. (16).

Consider the case of all offspring individuals being selected for the next generation. Then the sum $\sum_{j=1}^{\mu_{\text{MO}}} w'_{ij}{}^{(g+1)}$ evaluates to one. If these weights were used in Eq. (8), the covariance matrix $\mathbf{C}'_i{}^{(g+1)}$ would be replaced by the newly estimated covariance matrix $\mathbf{Z}^{(g+1)}$. This shows that we have to be careful not to put too much emphasis on $\mathbf{Z}^{(g+1)}$. The matrix $\mathbf{Z}^{(g+1)}$ has a rank of at most $\mu_{\text{MO},\text{succ}}$, which is likely to be less than n , and therefore just considering $\mathbf{Z}^{(g+1)}$ would lead to a degenerate covariance matrix.

The amount of “information” contained in $\mathbf{Z}^{(g+1)}$ clearly depends on the number of successful offspring $\mu_{\text{MO},\text{succ}}$. But due to the weighting, one can observe a “loss of variance” in $\mathbf{Z}^{(g+1)}$ we want to account for. To this end, we rely on the *variance effective selection mass*

$$\mu_{\text{MO},\text{eff}} = \frac{\left(\sum_{j=1}^{\mu_{\text{MO}}} w'_{ij}{}^{(g+1)} \right)^2}{\sum_{j=1}^{\mu_{\text{MO}}} \left(w'_{ij}{}^{(g+1)} \right)^2} \quad (19)$$

as a measure for the “amount of information” contained in $\mathbf{Z}^{(g+1)}$ [6]. The dependence of $\mu_{\text{MO},\text{eff}}$ on i is not indicated to keep the notation uncluttered.

To get an idea of this measure, let us assume that successful steps are distributed independently according to $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Then the weighted sum of successful steps is distributed according to

$$\sum_{j=1}^{\mu_{\text{MO}}} w'_{ij}{}^{(g+1)} \mathcal{N}(\mathbf{0}, \mathbf{I}) , \quad (20)$$

with variance $\sum_{j=1}^{\mu_{\text{MO}}} \left(w'_{ij}{}^{(g+1)} \right)^2$. As $\sum_{j=1}^{\mu_{\text{MO}}} \left(w'_{ij}{}^{(g+1)} \right)^2 \leq \left(\sum_{j=1}^{\mu_{\text{MO}}} w'_{ij}{}^{(g+1)} \right)^2$ we in general loose variance due to the weighting, and this is captured by $\mu_{\text{MO},\text{eff}}$.

The value of $\mu_{\text{MO,eff}}$ is always greater than one and less than or equal to $\mu_{\text{MO,succ}}$. It is equal to μ_{MO} for $w'_{i1} = \dots = w'_{i\mu_{\text{MO}}} = 1/\mu_{\text{MO}}$ and goes to one if all but one weights go to zero.

Finally, the relation between the information contributed by all selected steps and the information required to prevent from a degenerated covariance matrix is evaluated. A covariance matrix is a symmetric matrix with $n(n+1)/2$ degrees of freedom. With $\mu_{\text{MO,eff}}$ providing a measure of information within the offspring population, the term

$$\frac{\mu_{\text{MO,eff}}}{n(n+1)/2} = \frac{2\mu_{\text{MO,eff}}}{n(n+1)}, \quad (21)$$

gives an estimate of the relation between “contributed” and “required” knowledge. For constant n and $\mu_{\text{MO,eff}}$ large enough, the term evaluates to a value greater than one, thus indicating that a re-estimate of the covariance matrix based on the offspring population is possible without degenerating. If the value is less than one, the newly generated offspring do not exhibit enough information. In [4] a slightly different expression based on the same idea has been found and validated empirically. It reads

$$\frac{2\mu_{\text{MO,eff}} - 1}{(n+2)^2 + \mu_{\text{MO,eff}}}. \quad (22)$$

The calculation of the weights is now completed by incorporating this “estimate of information”. It is given by:

$$w_{ij}^{(g+1)} = w'_{ij}{}^{(g+1)} \min \left\{ 1, \frac{2\mu_{\text{MO,eff}} - 1}{(n+2)^2 + \mu_{\text{MO,eff}}} \right\}. \quad (23)$$

The weight $w'_{ij}{}^{(g+1)}$ is rescaled if the information contained within the successful offspring is not sufficient to prevent the covariance matrix from degenerating.³

If none of the offspring individuals is successful, all weights are equal to zero and no rank- $\mu_{\text{MO,succ}}$ -update of the covariance matrix is carried out. If all individuals of the offspring population are successful and enough information is provided by the selected steps, the old covariance matrix is discarded and re-estimated from scratch. The standard rank-one-update is always applied, see Eq. (8).

Now we have all ingredients for the modified MO-CMA-ES with “recombination” (in the sense that information from several offspring are combined) for learning strategy parameters. It is referred to as $(\mu_{\text{MO}} + \mu_{\text{MO}})$ -MO-CMA-ES. Only a small modification of the original $\mu_{\text{MO}} \times (1+1)$ -MO-CMA-ES (see Algorithm 1) is necessary. Before the **rank-one-update** is carried out (Algorithm 1, line 9), the covariance matrices of the individuals are updated according to the Procedure **rank- $\mu_{\text{MO,succ}}$ -update**.

³ The expression $\min \left\{ 1, \frac{2\mu_{\text{MO,eff}} - 1}{(n+2)^2 + \mu_{\text{MO,eff}}} \right\}$ evaluates to one only if $\mu_{\text{MO,eff}}$, which is bounded from above by μ_{MO} , is larger than $n^2 + 4n + 5$.

Procedure rank- $\mu_{\text{MO},\text{succ}}$ -update ($a = [\mathbf{x}, \bar{\mathbf{p}}_{\text{succ}}, \sigma, \mathbf{p}_c, \mathbf{C}], Q^{(g)}$)

```

1  $\mu_{\text{MO},\text{succ}} \leftarrow 0$ 
2 for  $k \leftarrow 1, \dots, \mu_{\text{MO}}$  do
3    $d_k \leftarrow \frac{\left\| \left[ \mathbf{x}'_k{}^{(g+1)} - \mathbf{x} \right]^T \mathbf{C}^{-1} \left[ \mathbf{x}'_k{}^{(g+1)} - \mathbf{x} \right] \right\|_2}{\sigma}$ 
4    $w_i \leftarrow \mathbb{I} \left( a'_k{}^{(g+1)} \prec_{Q^{(g)}} a_k^{(g)} \right) h \left( \frac{d_k}{\sqrt{n d_{\text{steps}}}} \right)$ 
5    $\mu_{\text{MO},\text{succ}} \leftarrow \mu_{\text{MO},\text{succ}} + \mathbb{I} \left( a'_k{}^{(g+1)} \prec_{Q^{(g)}} a_k^{(g)} \right)$ 
6    $\mu_{\text{MO},\text{eff}} \leftarrow \frac{\left( \sum_{i=1}^{\mu_{\text{MO}}} w_i \right)^2}{\sum_i^{\mu_{\text{MO}}} w_i^2}$ 
7   for  $k \leftarrow 1, \dots, \mu_{\text{MO}}$  do
8      $w_k \leftarrow \frac{w_k}{\mu_{\text{MO}} - \mu_{\text{MO},\text{succ}} + \sum_{i=1}^{\mu_{\text{MO}}} w_i} \min \left\{ 1, \frac{2\mu_{\text{MO},\text{eff}} - 1}{(n+2)^2 + \mu_{\text{MO},\text{eff}}} \right\}$ 
9    $\mathbf{C} \leftarrow \left( 1 - \sum_{i=1}^{\mu_{\text{MO}}} w_i \right) \mathbf{C} + \sum_{k=1}^{\mu_{\text{MO}}} w_k \frac{\mathbf{x}'_k{}^{(g+1)} - \mathbf{x}_k^{(g)}}{\sigma} \left( \frac{\mathbf{x}'_k{}^{(g+1)} - \mathbf{x}_k^{(g)}}{\sigma} \right)^T$ 

```

Choosing the right neighbourhood size by setting the parameter d_{steps} is crucial for the performance of the $(\mu_{\text{MO}} + \mu_{\text{MO}})$ -MO-CMA-ES. A value that works reliably across different fitness functions is desired, but there is no obvious heuristic for the selection of d_{steps} . For this reason, an empirical investigation of the performance of the $(\mu_{\text{MO}} + \mu_{\text{MO}})$ -MO-CMA-ES with different values for d_{steps} has been conducted in the context of this study. The bi-criteria benchmark function ELL1, CIGTAB1, ELL2, CIGTAB2 [1] and different search space dimensions n have been considered. Based on the results we derived the preliminary rule $d_{\text{steps}} = n + 3$.

4 Empirical Evaluation

This section presents a performance evaluation of the new $(\mu_{\text{MO}} + \mu_{\text{MO}})$ -MO-CMA-ES on a set of common multi-objective benchmark functions. Our goal is to answer the question whether the “recombination” of strategy parameters improves the performance of the MO-CMA-ES on a broad range of bi-objective fitness functions. Therefore we compare the $(\mu_{\text{MO}} + \mu_{\text{MO}})$ -MO-CMA-ES to the results of the original $\mu_{\text{MO}} \times (1+1)$ -MO-CMA-ES. For comparisons of this baseline algorithm with alternative multi-objective optimization methods we refer to previous studies [1,2,3]. The experiments have been conducted using the Shark machine learning library [12].

4.1 Experimental Setup

We compare the $(\mu_{\text{MO}}+\mu_{\text{MO}})$ - and the original $\mu_{\text{MO}}\times(1+1)$ -MO-CMA-ES presented in [1] on three classes of two-objective benchmark functions. Both algorithms rely on the hypervolume-indicator as second-level sorting criterion. The constrained benchmark functions ZDT1, ZDT2, ZDT3 and ZDT6 (see [13]) and their rotated variants IHR1, IHR2, IHR3 and IHR6 (see [1]) have been chosen for the performance evaluation. Moreover, the set of test problems has been augmented by the unconstrained and rotated functions ELLI1, ELLI2, CIGTAB1 and CIGTAB2 (see [1]), with the distance of the optima of the single objectives set to the default value two. The default search space dimension for constrained and non-rotated benchmark functions has been chosen to be 30. In case of rotated benchmark functions, the search space dimensions has been chosen to be 10.

The value of the parameter d_{steps} of the $(\mu_{\text{MO}}+\mu_{\text{MO}})$ -MO-CMA-ES has been set to the empirically validated choice of $n + 3$. The number of parent and offspring individuals has been set $\mu_{\text{MO}} = 100$. We conducted 50 independent trials and evaluated the algorithms after 250 and 500 generations.

The evaluation procedure adheres to the suggestions given in [14]. We briefly outline the process and refer to [9,15] for a detailed description of the methods. We consider the unary hypervolume-indicator and the unary additive ϵ_+ -indicator as performance measures. Before indicator values are computed, the data are normalized. We want to compare $k = 2$ algorithms on a particular optimization problem \mathbf{f} after g fitness evaluations and we assume that we have conducted t trials. We consider the non-dominated individuals of the union of all $k \cdot t$ populations after g evaluations. These individuals make up the reference set \mathcal{R} . Their objective vectors are normalized by an affine linear transformation such that for every objective the smallest and largest objective function values are mapped to 1 and 2, respectively. The value for the unary hypervolume-indicator is compared to the hypervolume of the reference set \mathcal{R} , which is also used to compute the ϵ_+ -indicator value. Therefore lower indicator values indicate better performance.

We used different reference sets for the evaluation after 250 and 500 generations. Therefore, the absolute values of the results of these two lines of experiments can not be compared.

4.2 Results

The results of the performance evaluation after 25,000 and 50,000 evaluations are presented in Tables 1 and 2. Although only small differences between the two algorithms can be observed, the $(\mu_{\text{MO}}+\mu_{\text{MO}})$ -MO-CMA-ES performed statistically significantly better than the $\mu_{\text{MO}}\times(1+1)$ -MO-CMA-ES in all our experiments. This shows that the strategy parameter adaptation of the MO-CMA-ES is clearly improved by considering the information contributed by selected offspring individuals. The results suggest that the choice of the parameter d_{steps} linearly dependent on the search space dimension n indeed results in a robust

behaviour of the $(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES across different classes and types of benchmark problems.

Table 1: Results of the performance comparison of the $(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES and the original $\mu_{MO}\times(1+1)$ -MO-CMA-ES, respectively. The table shows the median of 50 trials after 250 and 500 generations, respectively, of the hypervolume-indicator (the lower the better). The better value in each row is printed in bold. The superscripts indicate whether the $(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES_{*n*+3} performs significantly better than the $\mu_{MO}\times(1+1)$ -MO-CMA-ES, respectively (two-sided Wilcoxon rank sum test, ** indicates a significance level of 0.001 and * a significance level of 0.01). It is important to note that different reference sets were used for computing the values after 250 and 500 generations, respectively, and that therefore the absolute values after 250 and 500 generations can not be compared directly.

	$(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES _{<i>n</i>+3}	$\mu_{MO}\times(1+1)$ -MO-CMA-ES
250 Generations		
ZDT1	0.000564 **	0.000592
ZDT2	0.000304 **	0.000462
ZDT3	0.000279 **	0.000621
ZDT6	0.000006 *	0.000017
IHR1	0.000242 *	0.000443
IHR2	0.000783 **	0.000922
IHR3	0.000047 **	0.000066
IHR6	0.000009 **	0.000045
ELLI1	0.017792 *	0.018896
ELLI2	0.007773 *	0.008844
CIGTAB1	0.006947 **	0.007956
CIGTAB2	0.000549 **	0.000599
500 Generations		
ZDT1	0.000134 **	0.000201
ZDT2	0.000314 *	0.000429
ZDT3	0.000025 **	0.000172
ZDT6	0.000379 *	0.000421
IHR1	0.000006 **	0.000023
IHR2	0.005776 *	0.005111
IHR3	0.000067 **	0.000193
IHR6	0.000572 **	0.000977
ELLI1	0.000791 *	0.000844
ELLI2	0.004592 **	0.006392
CIGTAB1	0.000253 **	0.000542
CIGTAB2	0.003194 *	0.003978

Table 2: Results of the performance comparison of the $(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES and the original $\mu_{MO}\times(1+1)$ -MO-CMA-ES. The table shows the median of 50 trials after 250 and 500 generations of the ε_+ -indicator (the lower the better). The superscripts indicate whether the $(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES $_{n+3}$ performs significantly better than the $\mu_{MO}\times(1+1)$ -MO-CMA-ES, respectively (two-sided Wilcoxon rank sum test, ** indicates a significance level of 0.001 and * a significance level of 0.01). Different reference sets were used for computing the values after 250 and 500 generations, respectively, and therefore the absolute values after 250 and 500 generations can not be compared directly.

	$(\mu_{MO}+\mu_{MO})$ -MO-CMA-ES $_{n+3}$	$\mu_{MO}\times(1+1)$ -MO-CMA-ES
250 Generations		
ZDT1	0.013756**	<i>0.015349</i>
ZDT2	0.222876*	<i>0.400001</i>
ZDT3	0.076653*	<i>0.199655</i>
ZDT6	0.000112**	<i>0.000231</i>
IHR1	0.004432**	<i>0.005654</i>
IHR2	0.002005**	<i>0.010001</i>
IHR3	0.000003*	<i>0.000339</i>
IHR6	0.002134**	<i>0.002667</i>
ELLI1	0.027492*	<i>0.039816</i>
ELLI2	0.000573**	<i>0.000742</i>
CIGTAB1	0.001947**	<i>0.004753</i>
CIGTAB2	0.002239**	<i>0.003333</i>
500 Generations		
ZDT1	0.013756**	<i>0.015349</i>
ZDT2	0.222876*	<i>0.400001</i>
ZDT3	0.076653*	<i>0.199655</i>
ZDT6	0.000112**	<i>0.000231</i>
IHR1	0.004432**	<i>0.005654</i>
IHR2	0.002005**	<i>0.010001</i>
IHR3	0.000003*	<i>0.000339</i>
IHR6	0.002134**	<i>0.002667</i>
ELLI1	0.027492*	<i>0.039816</i>
ELLI2	0.000573**	<i>0.000742</i>
CIGTAB1	0.001947*	<i>0.004753</i>
CIGTAB2	0.002239*	<i>0.003333</i>

5 Conclusions and Future Work

We presented a new, more elaborate covariance matrix update scheme for the multi-objective covariance matrix adaptation evolution strategy (MO-CMA-ES). The difference from the original update method is that each individual additionally considers successful mutations of neighboring individuals. Such a lateral information transfer was not considered in the MO-CMA-ES so far, and it allows for faster adaptation of the covariance matrix. Our preliminary empirical evaluation on common bi-criteria benchmark functions shows that the new update scheme significantly improves the performance in all cases.

There is the need for further investigation and room for improvements of the proposed algorithm. For example, the choice of the parameter d_{steps} should be studied in more detail and different choices for the distance weighting function $h(\cdot)$, see Eq. (17), could be considered. The empirical evaluation should include additional benchmark functions, for instance, with a larger number of objectives. In particular, we are searching for functions where the covariance matrix update scheme presented here is outperformed by the original rank-one-update procedure.

The evaluation of the algorithms after a fixed number of evaluations – although standard in the empirical analysis of evolutionary multi-objective algorithms – may be misleading. In future work we will study the evolution of the absolute hypervolume over the whole optimization process.

References

1. Igel, C., Hansen, N., Roth, S.: Covariance matrix adaptation for multi-objective optimization. *Evolutionary Computation* **15**(1) (2006) 1–28
2. Igel, C., Suttorp, T., Hansen, N.: Steady-state selection and efficient covariance matrix update in the multi-objective cma-es. In: Fourth International Conference on Evolutionary Multi-Criterion Optimization (EMO 2007). Volume 4403 of LNCS., Springer-Verlag (2007)
3. Voß, T., Beume, N., Rudolph, G., Igel, C.: Scalarization versus indicator-based selection in multi-objective CMA evolution strategies. In: IEEE Congress on Evolutionary Computation 2008 (CEC 2008), IEEE Press (2008) 3041–3048
4. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation* **9**(2) (2001) 159–195
5. Hansen, N., Müller, S.D., Koumoutsakos, P.: Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation* **11**(1) (2003) 1–18
6. Hansen, N.: The CMA Evolution Strategy: A Tutorial. <http://www.bionik.tu-berlin.de/user/niko/cmatutorial.pdf>
7. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* **6** (2002) 182–197
8. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research* **181**(3) (2007) 1653–1669

9. Zitzler, E., Thiele, L.: Multiobjective optimization using evolutionary algorithms — a comparative case study. In Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.P., eds.: Fifth International Conference on Parallel Problem Solving from Nature (PPSN-V), Berlin, Germany, Springer-Verlag (1998) 292–301
10. Coello Coello, C.A., Van Veldhuizen, D.A., Lamont, G.B.: Evolutionary Algorithms for Solving Multi-Objective Problems. Kluwer Academic Publishers (2002)
11. Rechenberg, I.: Evolutionsstrategie '94. Werkstatt Bionik und Evolutionstechnik. Frommann-Holzboog, Stuttgart (1994)
12. Igel, C., Glasmachers, T., Heidrich-Meisner, V.: Shark. *Journal of Machine Learning Research* **9** (2008) 993–996
13. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation* **8**(2) (2000) 173–195
14. Bleuler, S., Laumanns, M., Thiele, L., Zitzler, E.: PISA – A platform and programming language independent interface for search algorithms. In Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L., eds.: *Evolutionary Multi-Criterion Optimization (EMO 2003)*. Volume 2632 of LNCS., Springer-Verlag (2003) 494 – 508
15. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Grunert da Fonseca, V.: Performance assesment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation* **7**(2) (2003) 117–132