Covariance Matrix Adaptation for Multi-objective Optimization

Christian Igel christian.igel@neuroinformatik.rub.de
Institut für Neuroinformatik, Ruhr-Universität Bochum, 44780 Bochum, Germany

Nikolaus Hansen nikolaus.hansen@inf.ethz.ch
Computational Laboratory (CoLab), Institute of Computational Science, ETH Zurich, 8092 Zurich, Switzerland

Stefan Roth stefan.roth@neuroinformatik.rub.de
Institut für Neuroinformatik, Ruhr-Universität Bochum, 44780 Bochum, Germany

Abstract
The covariance matrix adaptation evolution strategy (CMA-ES) is one of the most powerful evolutionary algorithms for real-valued single-objective optimization. In this paper, we develop a variant of the CMA-ES for multi-objective optimization (MOO). We first introduce a single-objective, elitist CMA-ES using plus-selection and step size control based on a success rule. This algorithm is compared to the standard CMA-ES. The elitist CMA-ES turns out to be slightly faster on unimodal functions, but is more prone to getting stuck in sub-optimal local minima. In the new multi-objective CMA-ES (MO-CMA-ES) a population of individuals that adapt their search strategy as in the elitist CMA-ES is maintained. These are subject to multi-objective selection. The selection is based on non-dominated sorting using either the crowding-distance or the contributing hypervolume as second sorting criterion. Both the elitist single-objective CMA-ES and the MO-CMA-ES inherit important invariance properties, in particular invariance against rotation of the search space, from the original CMA-ES. The benefits of the new MO-CMA-ES in comparison to the well-known NSGA-II and to NSDE, a multi-objective differential evolution algorithm, are experimentally shown.

1 Introduction

The covariance matrix adaptation evolution strategy (CMA-ES) is one of the most powerful evolutionary algorithms for real-valued optimization (Hansen and Ostermeier, 2001; Hansen et al., 2003; Hansen and Kern, 2004; Hansen, 2006b) with many successful applications (for an overview see Hansen, 2005). The main advantages of the CMA-ES lie in its invariance properties, which are achieved by carefully designed variation and selection operators, and in its efficient (self-)adaptation of the mutation distribution. The CMA-ES is invariant against order-preserving transformations of the fitness function value and in particular against rotation and translation of the search space—apart from the initialization. If either the strategy parameters are initialized accordingly or the time needed to adapt the strategy parameters is neglected, any affine transformation of the search space does not affect the performance of the CMA-ES. Rotation of the search space to test invariance and to generate non-separable functions was proposed

by Hansen et al. (1995), and the importance of such invariance properties for evolutionary algorithms is discussed in depth by Salomon (1996) and Hansen (2000). Note that an algorithm not being invariant against a certain group of transformations means that it is biased towards a certain class of problems defined w.r.t. those transformations, for example to tasks with separable fitness functions. Such a bias is only desirable if the applications for which the algorithm is designed fall into that special class. We think, for the transformations mentioned above this assumption is not attractive in general.

The interest in multi-objective optimization (MOO) is increasing rapidly. Several successful evolutionary MOO algorithms have been developed (Coello Coello et al., 2002; Deb, 2001), where the main focus of research has been put on the selection and archiving strategies. Multi-objective evolution strategies with (self-) adaptation of the search distribution have been proposed (Laumanns et al., 2001; Büche et al., 2003; Igel, 2005), but none of them achieves the invariance properties of the single-objective CMA-ES. Therefore in this study, we develop a variant of the CMA-ES for real-valued MOO.

In the CMA-ES a small population size is usually sufficient and only one set of strategy parameters is maintained. For MOO a large population is needed to evolve a diverse set of solutions, each ideally representing a (Pareto-) optimal compromise between the objectives. The optimal strategy parameters for the members of this population may differ considerably and should therefore be adapted individually. This suggests that it is reasonable to apply a MOO selection mechanism to a population of individuals each of which uses the strategy adaptation of the CMA-ES (for details about the covariance matrix adaptation we refer to Hansen and Ostermeier, 2001, and Hansen, 2006b). The standard single-objective CMA-ES relies on non-elitist (μ, λ)-selection, that is, the best μ of λ offspring form the next parent population and all former parents are discarded. For each set of strategy parameters to be adapted, several offspring have to be generated in each generation. If we want to maximize the number of different strategy parameter sets, given a fixed total number of offspring per iteration, the number of offspring per parent has to be as small as possible. Therefore, we first develop a single-objective, elitist CMA-ES with $(1+\lambda)$-selection, where λ can be chosen as small as one. In this elitist $(1+\lambda)$-CMA-ES the parent population consists of a single individual generating λ offspring and the best individual out of parent and offspring becomes the parent of the next generation. This $(1+\lambda)$-CMA-ES inherits all invariance properties from the original CMA-ES and is integrated into the MOO framework by considering, roughly speaking, a population of $(1+\lambda)$ evolution strategies, which are subject to multi-objective selection. Thus, the new MO-CMA-ES inherits important invariance properties from the original CMA-ES.

To summarize, the goal of this study is to augment evolutionary real-valued MOO with efficient adaptation of the search distribution and invariance against transformations of the search space. To achieve this, we develop an elitist variant of the single-objective CMA-ES. Its strategy adaptation mechanism can be combined with multi-objective selection using non-dominated sorting. To improve selection, we propose the contributing hypervolume as second sorting criterion. For better empirical evaluation, new biobjective benchmark functions are presented. The article is organized as follows. In the next section, the new single-objective elitist $(1+\lambda)$-CMA-ES is presented and empirically compared to the standard (μ, λ)-CMA-ES. Then, in Section 3, we introduce the MO-CMA-ES using either the original selection of the non-dominated sorting genetic algorithm II (NSGA-II; Deb et al., 2002) or a new modification thereof based on the contributing hypervolume of individuals (Emmerich et al., 2005). In Section 4, the two variants of the MO-CMA-ES are empirically compared with the NSGA-II and non-

dominated sorting differential evolution (NSDE; Iorio and Li, 2005). As far as we know, the latter is the only other evolutionary MOO algorithm invariant against rotation of the search space. The results substantiate our final conclusions.

2 A Single-objective Elitist CMA Evolution Strategy

In this section we combine the well known $(1+\lambda)$-selection scheme of evolution strategies (Rechenberg, 1973; Schwefel, 1995; Beyer and Schwefel, 2002) with the covariance matrix adaptation. The original update rule for the covariance matrix can be reasonably applied in the $(1+\lambda)$-selection. The cumulative step size adaptation (path length control) of the $(\mu/\mu, \lambda)$-CMA-ES is replaced by a success rule based step size control. The path length control cannot be easily applied, because the update of the evolution path stalls whenever no successful offspring is produced. If in this case the evolution path is long, the step size diverges.

Nomenclature In the $(1+\lambda)$-CMA-ES, each individual, a, is a 5-tuple $a = [\boldsymbol{x}, \overline{p}_{\mathrm{succ}}, \sigma, \boldsymbol{p}_c, \boldsymbol{C}]$ comprising its candidate solution vector $\boldsymbol{x} \in \mathbb{R}^n$, an averaged success rate $\overline{p}_{\mathrm{succ}} \in [0, 1]$, the global step size $\sigma \in \mathbb{R}_+$, an evolution path $\boldsymbol{p}_c \in \mathbb{R}^n$, and the covariance matrix $\boldsymbol{C} \in \mathbb{R}^{n \times n}$. Additionally, the following nomenclature is used:

$f : \mathbb{R}^n \to \mathbb{R}, \boldsymbol{x} \mapsto f(\boldsymbol{x})$ is the objective (fitness) function to be minimized.

$\lambda_{\mathrm{succ}}^{(g+1)} = \left| \left\{ i = 1, \dots, \lambda \left| f(\boldsymbol{x}_i^{(g+1)}) \le f(\boldsymbol{x}_{\mathrm{parent}}^{(g)}) \right. \right\} \right|$ is the number of successful new candidate solutions (successful offspring).

$\mathcal{N}(\boldsymbol{m}, \boldsymbol{C})$ is a multi-variate normal distribution with mean vector $\boldsymbol{m}$ and covariance matrix $\boldsymbol{C}$. The notation $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{C})$ denotes that random variable $\boldsymbol{x}$ is distributed according to the distribution $\mathcal{N}(\boldsymbol{m}, \boldsymbol{C})$.

$\boldsymbol{x}_{1:\lambda}^{(g)} \in \mathbb{R}^n$ is the best point from $\left\{ \boldsymbol{x}_1^{(g)}, \dots, \boldsymbol{x}_\lambda^{(g)} \right\}$, that is, $f(\boldsymbol{x}_{1:\lambda}^{(g)}) \le f(\boldsymbol{x}_i^{(g)})$ for all $i = 1, \dots, \lambda$.

2.1 The $(1+\lambda)$-CMA-ES

The algorithm is described within three routines. In the main routine, $(1+\lambda)$-CMA-ES, the λ new candidate solutions are sampled and the parent solution a_{parent} is updated depending on whether any of the new solutions is better than a_{parent}.

Algorithm 1: $(1+\lambda)$-CMA-ES

---

1 $g = 0$, initialize $a_{\text{parent}}^{(g)}$

2 **repeat**

3     $a_{\text{parent}}^{(g+1)} \leftarrow a_{\text{parent}}^{(g)}$

4   **for** $k = 1, \ldots, \lambda$ **do**

5        $\boldsymbol{x}_k^{(g+1)} \sim \mathcal{N}\left(\boldsymbol{x}_{\text{parent}}^{(g)}, {\sigma^{(g)}}^2 \boldsymbol{C}^{(g)}\right)$

6     $\texttt{updateStepSize}\left(a_{\text{parent}}^{(g+1)}, \dfrac{\lambda_{\text{succ}}^{(g+1)}}{\lambda}\right)$

7   **if** $f\left(\boldsymbol{x}_{1:\lambda}^{(g+1)}\right) \leq f\left(\boldsymbol{x}_{\text{parent}}^{(g)}\right)$ **then**

8        $\boldsymbol{x}_{\text{parent}}^{(g+1)} \leftarrow \boldsymbol{x}_{1:\lambda}^{(g+1)}$

9        $\texttt{updateCovariance}\left(a_{\text{parent}}^{(g+1)}, \dfrac{\boldsymbol{x}_{\text{parent}}^{(g+1)} - \boldsymbol{x}_{\text{parent}}^{(g)}}{\sigma_{\text{parent}}^{(g)}}\right)$

10    $g \leftarrow g + 1$

11 **until** stopping criterion is met

After sampling the new candidate solutions, the step size is updated based on the success rate $p_{\text{succ}} = \lambda_{\text{succ}}^{(g+1)}/\lambda$ with a learning rate c_p ($0 < c_p \leq 1$).

Procedure $\texttt{updateStepSize}(a = [\boldsymbol{x}, \overline{p}_{\text{succ}}, \sigma, \boldsymbol{p}_c, \boldsymbol{C}], p_{\text{succ}})$

---

1 $\overline{p}_{\text{succ}} \leftarrow (1 - c_p)\overline{p}_{\text{succ}} + c_p p_{\text{succ}}$

2 $\sigma \leftarrow \sigma \cdot \exp\left(\dfrac{1}{d} \dfrac{\overline{p}_{\text{succ}} - p_{\text{succ}}^{\text{target}}}{1 - p_{\text{succ}}^{\text{target}}}\right)$

This update rule is rooted in the $1/5$-success-rule proposed by Rechenberg (1973) and is an extension from the rule proposed by Kern et al. (2004). It implements the well-known heuristic that the step size should be increased if the success rate (i.e., the fraction of offspring better than the parent) is high, and the step size should be decreased if the success rate is low. The rule is reflected in the argument to the exponential function. For $\overline{p}_{\text{succ}} > p_{\text{succ}}^{\text{target}}$ the argument is greater than zero and the step size increases; for $\overline{p}_{\text{succ}} < p_{\text{succ}}^{\text{target}}$ the argument is smaller than zero and the step size decreases; for $\overline{p}_{\text{succ}} = p_{\text{succ}}^{\text{target}}$ the argument becomes zero and no change of σ takes place.

The argument to the exponential function is always smaller than $1/d$ and larger than $-1/d$ if $p_{\text{succ}}^{\text{target}} < 0.5$ (a necessary assumption). Therefore, the damping parameter d controls the rate of the step size adaptation. Using $\overline{p}_{\text{succ}}$ instead of the input argument $p_{\text{succ}} = \lambda_{\text{succ}}^{(g+1)}/\lambda$ primarily smoothes the single step size changes and has only a minor influence on the maximal possible step size changing rate.

If the best new candidate solution was better than the parent individual (see main routine), the covariance matrix is updated as in the $(1,\lambda)$-CMA-ES (see Hansen and Ostermeier, 2001).

Table 1: Default parameters for the $(1+\lambda)$-CMA Evolution Strategy.

Selection:

$\lambda = 1$

Step size control:

$$d = 1 + \frac{n}{2\,\lambda} \;,\quad p_{\text{succ}}^{\text{target}} = \frac{1}{5 + \sqrt{\lambda}/2} \;,\quad c_p = \frac{p_{\text{succ}}^{\text{target}}\lambda}{2 + p_{\text{succ}}^{\text{target}}\lambda}$$

Covariance matrix adaptation:

$$c_c = \frac{2}{n+2}, \quad c_{\text{cov}} = \frac{2}{n^2 + 6} \;,\quad p_{\text{thresh}} = 0.44$$

Procedure `updateCovariance`$(a = [\boldsymbol{x}, \overline{p}_{\text{succ}}, \sigma, \boldsymbol{p}_c, \boldsymbol{C}], \boldsymbol{x}_{\text{step}} \in \mathbb{R}^n)$

---

1 **if** $\overline{p}_{\text{succ}} < p_{\text{thresh}}$ **then**

2 $\quad \boldsymbol{p}_c \leftarrow (1 - c_c)\boldsymbol{p}_c + \sqrt{c_c(2 - c_c)}\, \boldsymbol{x}_{\text{step}}$

3 $\quad \boldsymbol{C} \leftarrow (1 - c_{\text{cov}})\boldsymbol{C} + c_{\text{cov}} \cdot \boldsymbol{p}_c\boldsymbol{p}_c^{\mathrm{T}}$

4 **else**

5 $\quad \boldsymbol{p}_c \leftarrow (1 - c_c)\boldsymbol{p}_c$

6 $\quad \boldsymbol{C} \leftarrow (1 - c_{\text{cov}})\boldsymbol{C} + c_{\text{cov}} \cdot \left(\boldsymbol{p}_c\boldsymbol{p}_c^{\mathrm{T}} + c_c(2 - c_c)\boldsymbol{C}\right)$

The update of the evolution path $\boldsymbol{p}_c$ depends on the value of $\overline{p}_{\text{succ}}$ (here the smoothing of $\lambda_{\text{succ}}/\lambda$ is of considerable relevance). If the smoothed success rate $\overline{p}_{\text{succ}}$ is high, that is, above $p_{\text{thresh}} < 0.5$, the update of the evolution path $\boldsymbol{p}_c$ is stalled. This prevents an overly fast increase of axes of $\boldsymbol{C}$ when the step size is far too small, for example, in a linear surrounding. If the smoothed success rate $\overline{p}_{\text{succ}}$ is low, the update of $\boldsymbol{p}_c$ is accomplished by obeying an exponential smoothing. The constants c_c and c_{cov} ($0 \leq c_{\text{cov}} < c_c \leq 1$) are learning rates for the evolution path and the covariance matrix, respectively. The factor $\sqrt{c_c(2 - c_c)}$ normalizes the variance of $\boldsymbol{p}_c$ viewed as a random variable (see Hansen and Ostermeier, 2001). The evolution path $\boldsymbol{p}_c$ is used to update the covariance matrix. The new covariance matrix is a weighted mean of the old covariance matrix and the outer product of $\boldsymbol{p}_c$. In the second case (line 5), the second summand in the update of $\boldsymbol{p}_c$ is missing and the length of $\boldsymbol{p}_c$ shrinks. Although of minor relevance, the term $c_c(2 - c_c)\boldsymbol{C}$ (line 6) compensates for this shrinking in $\boldsymbol{C}$.

Strategy Parameters The (external) strategy parameters are offspring number λ, 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 are given in Table 1. Most default values are derived from the precursor algorithms and validated by sketchy simulations on simple test functions: the target success rate is close to the well-known $1/5$ and depends on λ, because the optimal success rate in the $(1 + \lambda)$-ES certainly decreases with increasing λ (Beyer, 2001, p. 78f). The parameters for the covariance matrix adaptation are similar to those for the $(1,\lambda)$-CMA-ES.

Initialization The elements of the initial individual, $a_{\text{parent}}^{(0)}$ are set to $\overline{p}_{\text{succ}} = p_{\text{succ}}^{\text{target}}$, $\boldsymbol{p}_c = \boldsymbol{0}$, and $\boldsymbol{C} = \boldsymbol{I}$, where $p_{\text{succ}}^{\text{target}}$ is given in Table 1. The initial candidate solution $\boldsymbol{x} \in \mathbb{R}^n$ and the initial $\sigma \in \mathbb{R}_+$ must be chosen problem dependent. The optimum should presumably be within the cube $\boldsymbol{x} \pm 2\,\sigma\,(1, \ldots, 1)^{\mathrm{T}}$.

Table 2: Single-objective test functions to be minimized, where $y = Ox$ and O is an orthogonal matrix, implementing an angle-preserving linear transformation.

Name	Function	Initial region
Linear	$f_{\text{linear}}(x) = y_1$	$O^{-1}[6000, 6006]^n$
Sphere	$f_{\text{sphere}}(x) = \sum_{i=1}^{n} x_i^2$	$O^{-1}[-1, 5]^n$
Ellipsoid	$f_{\text{elli}}(x) = \sum_{i=1}^{n} \left(1000^{\frac{i-1}{n-1}} y_i\right)^2$	$O^{-1}[-1, 5]^n$
Rastrigin	$f_{\text{rastrigin}}(x) = 10n + \sum_{i=1}^{n} \left(y_i^2 - 10\cos(2\pi y_i)\right)$	$O^{-1}[-1, 5]^n$

2.2 Simulation of the (1+λ)-CMA-ES

Test functions To validate the essential properties of the search algorithm we use the single-objective test problems summarized in Table 2. The linear function f_{linear} tests the ability and the speed to increase the step size σ. On f_{sphere} basic convergence properties and the speed of step size decrease are tested. On f_{elli} the performance of the CMA procedure, that is, the ability to adapt the distribution shape to the function topography is examined. On $f_{\text{rastrigin}}$, the ability to circumvent local optima is examined. Apart from f_{linear}, the optimum function value is zero for all functions. The experimental results are independent of angle-preserving transformations, like translation and rotation of the search space, that is, they are in particular independent of the chosen orthogonal transformation matrix O.

Methods We conducted 51 runs for each function and each dimension. The initial candidate solution x is chosen uniformly randomly in the initial region from Table 2, and the initial $\sigma = 3$ is half of the width of the initial interval. Excepting f_{linear}, the simulation is stopped when function value differences do not exceed 10^{-12} or when the function value becomes smaller than the target function value 10^{-9}. To conduct statistical testing the runs were ranked. Runs that reached the target function value were regarded as better and ranked according to their number of function evaluations. The remaining runs were ranked according to their final function value. To evaluate statistical significance the non-parametric Mann-Whitney U-test (Wilcoxon rank sum test) was conducted. If not stated otherwise discussed differences are significant with $p < 0.001$.

Results and Discussion The (1+λ)-CMA-ES is compared to the (μ/μ_{W},λ)-CMA-ES, the standard CMA-ES with weighted global intermediate (μ/μ_{W}) recombination as described by Hansen and Kern (2004). The former is elitist and has a success rule based step size adaptation. The latter is non-elitist, uses the cumulative step size adaptation (path length control), and conducts weighted recombination of all $\mu = \lfloor \lambda/2 \rfloor$ parents.

On f_{linear} the step size increases linearly on the log-scale in all strategy variants, a minimal necessary demand on step size control (Hansen, 2006a). The mean number of function evaluations to increase the step size by one order of magnitude is shown in Table 3 for two plus- and two comma-strategies. The success rule in the plus-strategy is up to five times faster than the path length control in the comma-strategies, but this difference should usually be irrelevant.

Runs on f_{sphere}, f_{elli}, and $f_{\text{rastrigin}}$ are shown in Figure 1. First, we discuss the comparison between (1+λ)- and (1,λ)-CMA-ES for the same λ ($\square$ and $\diamond$ in the figure). On f_{sphere} and f_{elli} the two strategies perform quite similarly. The slight differences on

Table 3: Mean number of function evaluations needed to increase the step size by a factor of ten on f_{linear}, divided by $n/5$.

n	λ	(1+1)	(1+λ)	(1,λ)	(μ/μ_{W},λ)
5	8	25	60	98	72
20	12	18	32	96	56

f_{sphere} are primarily the result of different step size change rates. The reason for the slight differences on f_{elli} is presumably the smaller target step size of the success based adaptation rule. A smaller step size can lead to a more pronounced evolution path that assists a faster adaptation of the covariance matrix. Both strategies perform identically on $f_{\mathrm{rastrigin}}$ in 5D, but in 20D non-elitist (1,λ) finds significantly better solutions. The reasons are probably the advantage of the comma-selection scheme in escaping local optima and the larger adapted step sizes.

More pronounced differences can be observed between the default variants (1+1) and (μ/μ_{W},λ). On f_{sphere} and f_{elli} elitist (1+1) is roughly 1.5 times faster than (μ/μ_{W},λ). On $f_{\mathrm{rastrigin}}$ the standard (μ/μ_{W},λ) finds the considerably (and significantly) better solutions. Here, the performance of the plus-strategy can be considerably improved if the step size change rate is slowed down by increasing the damping d, but nevertheless remains below that of the (μ/μ_{W},λ) strategy.

The empirical results give evidence that the plus-selection is effectively combined with the covariance matrix adaptation. On the one hand, the plus-selection together with the success rule based adaptation for the step size makes the evolution strategy faster by a factor of about 1.5 on unimodal functions. On the other hand, the comma-strategy is less susceptible to get trapped into sub-optimal local minima for two reasons. First, even a particularly well evaluated individual is abandoned in the next generation; second, the path length control adapts larger step lengths, in particular within the recombinant strategy variant (the default one).

3 Covariance Matrix Adaptation for Multi-objective Optimization

Based on the (1+λ)-CMA-ES we propose a multi-objective evolution strategy. After a brief introduction to evolutionary multi-objective optimization, we present the considered selection mechanisms, which are based on non-dominated sorting. We propose an alternative ranking of individuals that have the same level of non-dominance. The ranking relies on the contributing hypervolume and can be computed efficiently for two objectives. Then the (1+λ)-MO-CMA-ES is described.

3.1 Multi-objective Optimization

Consider an optimization problem with M objectives $f_1, \ldots, f_M : X \to \mathbb{R}$ to be minimized. The vector $f(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x}))$ is the objective vector of $\boldsymbol{x} \in X$ living in the objective space $\mathbb{R}^M$. The elements of X can be strictly partially ordered using the concept of Pareto dominance. A solution $\boldsymbol{x} \in X$ dominates a solution $\boldsymbol{x}'$ and we write $\boldsymbol{x} \prec \boldsymbol{x}'$ iff $\forall m \in \{1, \ldots, M\} : f_m(\boldsymbol{x}) \leq f_m(\boldsymbol{x}')$ and $\exists m \in \{1, \ldots, M\} : f_m(\boldsymbol{x}) < f_m(\boldsymbol{x}')$. The elements of the (Pareto) set $\{\boldsymbol{x} \mid \boldsymbol{x} \in X \wedge \nexists \boldsymbol{x}' \in X : \boldsymbol{x}' \prec \boldsymbol{x}\}$ are called Pareto optimal. The corresponding Pareto front is given by $\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in X \wedge \nexists \boldsymbol{x}' \in X : \boldsymbol{x}' \prec \boldsymbol{x}\} \subset \mathbb{R}^M$.

Without any further information no Pareto-optimal solution can be said to be superior to another. The goal of multi-objective optimization (MOO) is to find a diverse
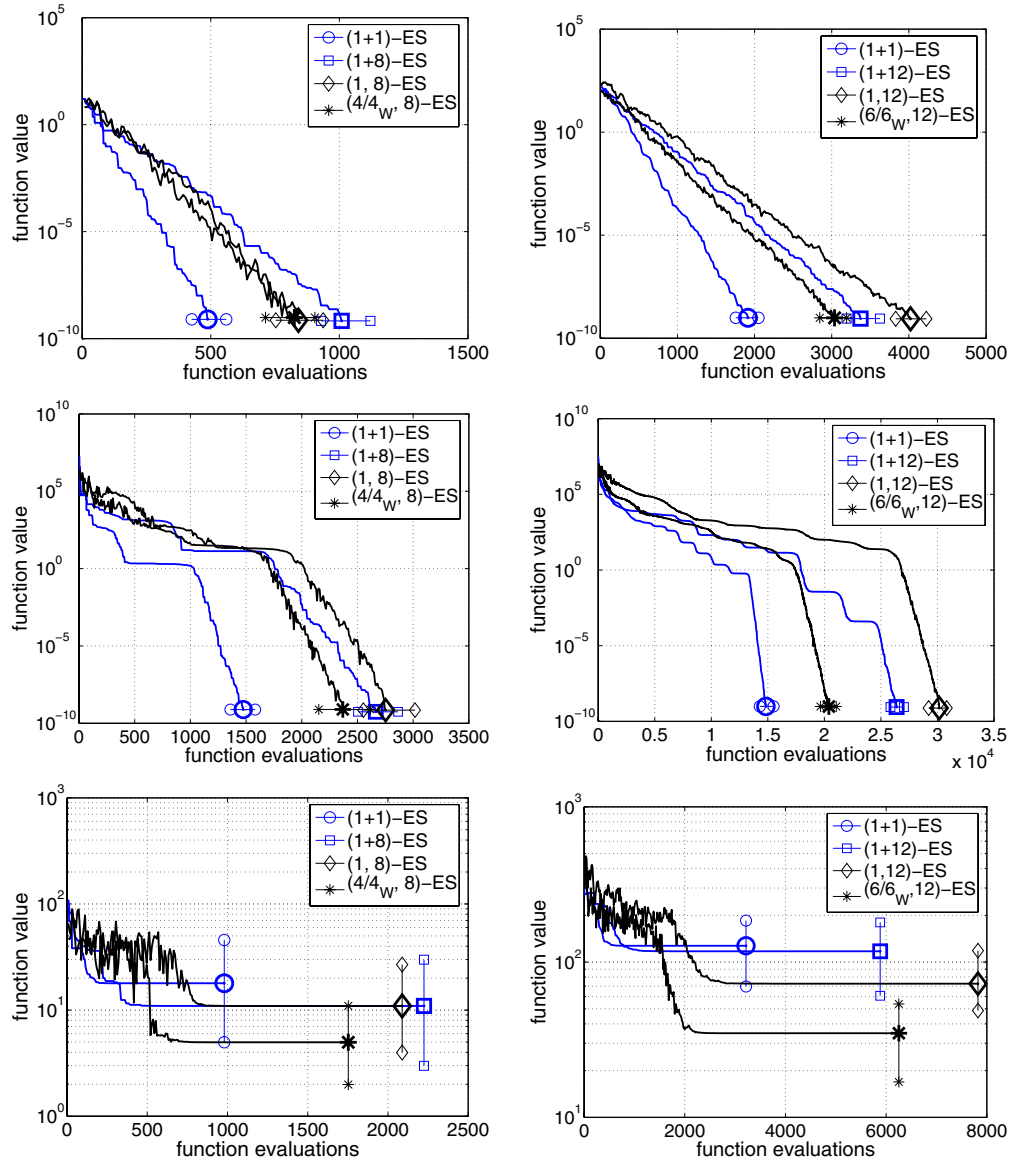
Figure 1: Simulations on Sphere (above), Ellipsoid (middle), and Rastrigin function (below), in 5D ($n = 5$, left) and 20D ($n = 20$, right). Shown is the median out of 51 runs for the (1+1)-, (1+λ)-, (1,λ)-, and (μ/μ_W,λ)-CMA-ES. The error bars denote final values for the 3rd and the 49th run (5%- and 95%-percentile).

set of Pareto-optimal solutions, which provide insights into the trade-offs between the objectives. When approaching a MOO problem by linearly aggregating all objectives into a scalar function, each weighting of the objectives yields only a subset of Pareto-optimal solutions (usually only a single solution). That is, various optimization trials with different aggregations become necessary to find a diverse set of Pareto-optimal solutions. Even worse, no linear aggregate exists such that concave parts of the Pareto

front become optimal. Therefore, various trials cannot help in case of partially concave Pareto fronts (cf. Das and Dennis, 1997). Consequently, evolutionary multi-objective algorithms have become more and more popular for MOO in the last years (Coello Coello et al., 2002; Deb, 2001).

In the following, we consider evolutionary real-valued MOO, where each individual $a_i^{(g)}$ at generation g represents a real-valued candidate solution $\boldsymbol{x}_i^{(g)} \in X \subseteq \mathbb{R}^n$ of an n-dimensional problem with M objectives. For simplicity, we do not distinguish between $f_m(a_i^{(g)})$ and $f_m(\boldsymbol{x}_i^{(g)})$.

3.2 Multi-objective Selection

Our multi-objective algorithm is based on the non-dominated sorting approach used in NSGA-II (Deb, 2001; Deb et al., 2002). The individuals are sorted according to their level of non-dominance. To rank individuals on the same level an additional sorting criterion is needed. We consider two criteria, the crowding-distance and the contributing hypervolume.

3.2.1 Non-dominated Sorting

First of all, the elements in a population A of candidate solutions are ranked according to their level of non-dominance. Let the non-dominated solutions in A be denoted by $\mathrm{ndom}(A) = \{a \in A \,|\, \nexists a' \in A : a' \prec a\}$. The Pareto front of A is then given by $\{(f_1(a), \ldots, f_M(a)) \,|\, a \in \mathrm{ndom}(A)\}$. The elements in $\mathrm{ndom}(A)$ get rank 1. The other ranks are defined recursively by considering the set without the solutions with lower ranks. Formally, let $\mathrm{dom}_0(A) = A$, $\mathrm{dom}_l(A) = \mathrm{dom}_{l-1}(A) \setminus \mathrm{ndom}_l(A)$, and $\mathrm{ndom}_l(A) = \mathrm{ndom}(\mathrm{dom}_{l-1}(A))$ for $l \in \{1, \ldots\}$. For $a \in A$ we define the level of non-dominance $\mathrm{r}(a, A)$ to be i iff $a \in \mathrm{ndom}_i(A)$. The time complexity of non-dominated sorting of N elements is $\mathcal{O}(MN^2)$ (Deb et al., 2002).

A second sorting criterion is needed to rank the solutions having the same level of non-dominance. This criterion is very important, as usually (in particular in real-valued optimization of continuous objective functions) after some generations there are more non-dominated solutions in the population than solutions to be selected. We consider two alternative second sorting criteria: the crowding-distance (Deb et al., 2002) and the contributing hypervolume (Emmerich et al., 2005).

3.2.2 Crowding-distance

In the NSGA-II, non-dominated solutions A' with the same level of non-dominance are ranked according to how much they contribute to the spread (or diversity) of objective function values in A'. This can be measured by the crowding-distance. For M objectives, the crowding-distance of $a \in A'$ is given by

$$c(a, A') = \sum_{m=1}^{M} c_m(a, A') / (f_m^{\max} - f_m^{\min}) \ ,$$

where $f_m^{\max}$ and $f_m^{\min}$ are (estimates of) the minimum and maximum value of the mth objective and

$c_m(a, A') :=$
$$\begin{cases} \infty \, , \text{ if } f_m(a) = \min\{f_m(a') \,|\, a' \in A'\} \text{ or } f_m(a) = \max\{f_m(a') \,|\, a' \in A'\} \\ \min\{f_m(a'') - f_m(a') \,|\, a', a'' \in A' \setminus \{a\} : f_m(a') \leq f_m(a) \leq f_m(a'')\} \, , \text{ otherwise.} \end{cases}$$

Based on the level of non-dominance and the crowding-distance we define the relation

$$a \prec_{c,A'} a' \Leftrightarrow \mathrm{r}(a, A') < \mathrm{r}(a', A') \text{ or}$$
$$\left[(\mathrm{r}(a, A') = \mathrm{r}(a', A')) \wedge (c(a, \mathrm{ndom}_{\mathrm{r}(a',A')}(A')) > c(a', \mathrm{ndom}_{\mathrm{r}(a',A')}(A'))) \right] ,$$

for $a, a' \in A'$. That is, a is better than a' when compared using $\prec_{c,A'}$ if either a has a better (lower) level of non-dominance or a and a' are on the same level but a is in a "lesser crowded region of the objective space" and therefore induces more diversity.

The crowding-distance of N non-dominated solutions can be computed efficiently in $\mathcal{O}(MN \log N)$ (Deb et al., 2002). However, the crowding-distance is related to the spread of solutions, which may be a desirable quantity and foster evolvability, but it is not directly related to progress in terms of selecting better solutions as we will discuss in section 4.1.

3.2.3 Contributing Hypervolume

The hypervolume measure or $\mathcal{S}$-metric was introduced by Zitzler and Thiele (1998) 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 (Coello Coello et al., 2002):

$$\mathcal{S}_{a_{\mathrm{ref}}}(A') = \Lambda \left(\bigcup_{a \in \mathrm{ndom}(A')} \{(f_1(a'), \ldots, f_M(a')) \,|\, a \prec a' \prec a_{\mathrm{ref}}\} \right) ,$$

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

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

Using binary indicators for selection was proposed by Zitzler and Künzli (2004), and they also used a binary indicator based on the hypervolume concept to assign fitness values to individuals. The contributing hypervolume was also used for selection in a steady-state evolutionary algorithm proposed by Emmerich et al. (2005). Here, we adopt the latter approach for the ranking of a whole population.

The rank $s(a, A')$ of an individual a can be defined recursively based on its contribution to the hypervolume, where ties are broken at random. The individual contributing least to the hypervolume of A' gets the worst rank. The individual contributing least to the hypervolume of A' without the individual with the worst rank is assigned the second worst rank and so on. We call $a \in A'$ a boundary element if $\Delta_{\mathcal{S}}(a, A')$ depends on the choice of the reference point a_{ref}. We choose a_{ref} such that all elements in A' dominate a_{ref} and that for any boundary element $a \in A'$ and any non boundary element $a' \in A'$ we have $\Delta_{\mathcal{S}}(a, A') > \Delta_{\mathcal{S}}(a', A')$. (For example, this can be achieved by an a_{ref} that is extremely bad in each objective. The exact choice of a_{ref} is irrelevant in the following experiments.) That is, the individuals at the "boundaries" of the Pareto front of A' are preferably selected. Let a lower rank be worse. Formally (assuming that argmin breaks ties randomly), for $a \in \mathrm{ndom}(A')$ we have $s(a, A') = 1$ if $a = \mathrm{argmin}_{a' \in A'}\{\Delta_{\mathcal{S}}(a', A')\}$ and $s(a, A') = n$ if $a = \mathrm{argmin}_{a' \in A'}\{\Delta_{\mathcal{S}}(a', A' \setminus \{a'' \,|\, s(a'', A') < n\})\}$.

For two objectives, this ranking can be calculated efficiently in log-linear time in the number of individuals using appropriate data structures and the equation for $\Delta_{\mathcal{S}}(a, A')$ given by Emmerich et al. (2005). Unfortunately, the scaling behavior in the number of objectives is likely to be bad (While, 2005). Still, we can show:

Lemma 1. *For two objectives, the ranks $s(a, A')$ of a set A' of non-dominated individuals $a \in A'$, $|A'| = N$, can be computed in $\mathcal{O}(N \log N)$ time.*

Proof. In the following, we describe an algorithm that computes the ranking in log-linear time by storing the relevant information in appropriate data structures. We consider sorted indexed lists F and S containing individuals sorted by first fitness value and by contributing hypervolume, respectively. Consider the list S containing an individual a. Then $S[l]$ returns the lth element of S, $\text{index}(S, a)$ gives the number of a in the list (i.e., $S[\text{index}(S, a)] = a$), and $\text{insert}(S, a)$ adds and $\text{delete}(S, a)$ removes a from S. We presume an appropriate data structure (say, an AVL-tree, e.g., Knuth, 1973) that allows these look-up, insertion, and deletion operations in $\mathcal{O}(\log N)$ time, where N is the number of elements in the list.

First, F is filled with the elements in A' sorted by their first fitness value. Then the elements in A' are inserted into S sorted by their contributing hypervolume. This can also be done in $\mathcal{O}(N \log N)$, because the contributing hypervolume of an individual a to the hypervolume of a set B can be computed by

$$\Delta_{\mathcal{S}}(a, B) =$$
$$\begin{cases} (f_1(a_{\text{ref}}) - f_1(a)) \cdot (f_2(F[\text{index}(F, a) - 1]) - f_2(a)) & \text{if } a = F[|B|] \\ (f_1(F[\text{index}(F, a) + 1]) - f_1(a)) \cdot (f_2(a_{\text{ref}}) - f_2(a)) & \text{if } a = F[1] \\ (f_1(F[\text{index}(F, a) + 1]) - f_1(a)) \cdot (f_2(F[\text{index}(F, a) - 1]) - f_2(a)) & \text{otherwise.} \end{cases}$$

Due to the choice of a_{ref}, the boundary elements have the largest contributing hypervolumes, $S[|A'| - 1]$ and $S[|A'|]$, and thus get the ranks $|A'| - 1$ and $|A'|$. Now, if $|A'| > 2$ we set $l \leftarrow 1$ and the following procedure is repeated $|A'| - 2$ times. We determine $a \leftarrow S[1]$, the element contributing least to the hypervolume, and its neighbors in F by looking up $i \leftarrow \text{index}(F, a)$, and $a_{-1} \leftarrow F[i - 1]$ and $a_{+1} \leftarrow F[i + 1]$. Note that a_{-1} and a_{+1} exist, because the elements with the extreme f_1 values have the largest contributing hypervolumes. The individual a is assigned the rank l, $s(a, A') \leftarrow l$, and is deleted from both lists, $\text{delete}(S, a)$ and $\text{delete}(F, a)$. We set $l \leftarrow l + 1$. The elements a_{+1} and a_{-1} are deleted from S, $\text{delete}(S, a_{-1})$ and $\text{delete}(S, a_{+1})$. The contributing hypervolumes (to the hypervolume covered by the remaining elements in F) are recomputed for a_{+1} and a_{-1} using the equation given above, and the elements are reinserted into S according to the new contributing hypervolumes, $\text{insert}(S, a_{-1})$ and $\text{insert}(S, a_{+1})$.

All operations in this loop can be done in constant or logarithmic time, which proves the lemma. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\Box$

Based on this ranking and the level of non-dominance we define the relation

$$a \prec_{s,A} a' \Leftrightarrow \text{r}(a, A) < \text{r}(a', A) \text{ or}$$
$$\left[(\text{r}(a, A) = \text{r}(a', A)) \wedge (s(a, \text{ndom}_{\text{r}(a',A)}(A)) > s(a', \text{ndom}_{\text{r}(a',A)}(A))) \right] \ ,$$

for $a, a' \in A$. That is, a is better than a' when compared using $\prec_{s,A}$ if either a has a better level of non-dominance or a and a' are on the same level but a contributes more to the hypervolume when considering the points at that level of non-dominance.

3.3 MO-CMA-ES

Now we have all the ingredients for a multi-objective CMA-ES. In the $\lambda_{\text{MO}} \times (1+\lambda)$-MO-CMA-ES, we maintain a population of λ_{MO} elitist $(1+\lambda)$-CMA-ES. The kth individual

in generation g is denoted by $a_k^{(g)} = [\boldsymbol{x}_k^{(g)}, \overline{p}_{\text{succ},k}^{(g)}, \sigma_k^{(g)}, \boldsymbol{p}_{c,k}^{(g)}, \boldsymbol{C}_k^{(g)}]$. For simplicity, we consider only the standard case $\lambda = 1$. The extension to $\lambda > 1$ is straightforward.

In every generation g each of the λ_{MO} parents generates $\lambda = 1$ offspring. Parents and offspring form the set $Q^{(g)}$. The step sizes of a parent and its offspring are updated depending on whether the mutations were successful, that is, whether the offspring is better than the parent according to the relation $\prec_{Q^{(g)}}$. The covariance matrix of the offspring is updated taking into account the mutation that has led to its genotype. Both the step size and the covariance matrix update are the same as in the single-objective $(1+\lambda)$-CMA-ES. The best λ_{MO} individuals in $Q^{(g)}$ sorted by $\prec_{Q^{(g)}}$ form the next parent generation.

Putting all together, the $\lambda_{\text{MO}} \times (1+1)$-MO-CMA reads:

Algorithm 4: $\lambda_{\text{MO}} \times (1+1)$-MO-CMA

---

1 $g = 0$, initialize $a_k^{(g)}$ for $k = 1, \ldots, \lambda_{\text{MO}}$

2   **repeat**

3   **for** $k = 1, \ldots, \lambda_{\text{MO}}$ **do**

4 $a'^{(g+1)}_k \leftarrow a_k^{(g)}$

5 $\boldsymbol{x}'^{(g+1)}_k \sim \mathcal{N}\left(\boldsymbol{x}_k^{(g)}, \sigma_k^{(g)^2} \boldsymbol{C}_k^{(g)}\right)$

6 $Q^{(g)} = \left\{ a'^{(g+1)}_k, a_k^{(g)} \mid 1 \le k \le \lambda_{\text{MO}} \right\}$

7   **for** $k = 1, \ldots, \lambda_{\text{MO}}$ **do**

8 $\texttt{updateStepSize}\left(a_k^{(g)}, \lambda_{\text{succ},Q^{(g)},k}^{(g+1)}\right)$

9 $\texttt{updateStepSize}\left(a'^{(g+1)}_k, \lambda_{\text{succ},Q^{(g)},k}^{(g+1)}\right)$

10 $\texttt{updateCovariance}\left(a'^{(g+1)}_k, \dfrac{\boldsymbol{x}'^{(g+1)}_k - \boldsymbol{x}_k^{(g)}}{\sigma_k^{(g)}}\right)$

11   **for** $i = 1, \ldots, \lambda_{\text{MO}}$ **do**

12 $a_i^{(g+1)} \leftarrow Q_{\prec:i}^{(g)}$

13 $g \leftarrow g + 1$

14 **until** stopping criterion is met

Here

$$\lambda_{\text{succ},Q^{(g)},k}^{(g+1)} = \begin{cases} 1 & \text{, if } a'^{(g+1)}_k \prec_{Q^{(g)}} a_k^{(g)} \\ 0 & \text{, otherwise} \end{cases} \quad \text{is the number of successful offspring from}$$

parent $a_k^{(g)}$ for $\lambda = 1$ and

$Q_{\prec:i}^{(g)}$ is the ith best offspring in $Q^{(g)}$ w.r.t. $\prec_{Q^{(g)}}$.

We consider two variants of the MO-CMA-ES, the c-MO-CMA and the s-MO-CMA, which use the crowding-distance and the contributing hypervolume as second level sorting criterion, respectively. That is, $\prec_{Q^{(g)}} := \prec_{c,Q^{(g)}}$ in the c-MO-CMA and $\prec_{Q^{(g)}} := \prec_{s,Q^{(g)}}$ in the s-MO-CMA, see Section 3.2.

Handling Box constraints Consider an optimization problem with M objectives $f_1, \ldots, f_M : X \rightarrow \mathbb{R}$ with $X = [x_1^l, x_1^u] \times \cdots \times [x_n^l, x_n^u] \subset \mathbb{R}^n$. For $\boldsymbol{x} \in \mathbb{R}^n$ let $\text{feasible}(\boldsymbol{x}) = (\min(\max(x_1, x_1^l), x_1^u), \ldots, \min(\max(x_n, x_n^l), x_n^u))^{\text{T}}$. We define the penalized fitness

$$f_m^{\text{penalty}}(\boldsymbol{x}) = f_m(\text{feasible}(\boldsymbol{x})) + \alpha \|\boldsymbol{x} - \text{feasible}(\boldsymbol{x})\|_2^2$$

where $\alpha > 0$ is a penalty parameter.

When in this study the MO-CMA-ES is applied to problems with box constraints the penalized fitness functions f_m^{penalty} with $\alpha = 10^{-6}$ are used in the evolutionary process.

4 Empirical Evaluation of the MO-CMA-ES

In this section, we demonstrate how the MO-CMA behaves on test functions. First, we discuss performance assessment of stochastic multi-objective algorithms in general and introduce the performance indicators. Then we empirically compare the c-MO-CMA, the s-MO-CMA, the NSGA-II, and the differential evolution method NSDE on common and new benchmark problems.

4.1 Evaluating the Performance of MOO Algorithms

The performance assessment of stochastic multi-objective algorithms is more difficult than evaluating single-objective algorithms: In empirical investigations, sets of sets, the non-dominated solutions evolved in multiple trials of different algorithms, have to be compared. Many ways of measuring the performance of MOO algorithms have been proposed. In this study, we follow recommendations by Fonseca et al.(2005, see also Knowles, Thiele, and Zitzler, 2005). We concisely define the performance measures used, for a detailed description of the methods we refer to the literature (Knowles and Corne, 2002; Zitzler et al., 2003; Knowles et al., 2005).

Given two sets of solutions $A, B \subseteq X$ there is a common sense definition of one set being better than the other. Set A is better than B and we write $A \triangleleft B$ if for every element $a \in B$ there exists an element $a' \in A$ that is not worse than a in each objective, $\forall m \in \{1, \ldots, M\}, \forall a \in B, \exists a' \in A : f_m(a') \leq f_m(a)$, and $\text{ndom}(A) \neq \text{ndom}(B)$. Otherwise we have $A \ntriangleleft B$. Regularly for two sets, A and B, neither $A \triangleleft B$ nor $B \triangleleft A$ holds. Therefore, quality indicators are introduced.

A unary quality indicator assigns a real valued quality to a set of solutions. Here, the hypervolume indicator (Zitzler and Thiele, 1998) and the ϵ-indicator (Zitzler et al., 2003) are measured. We use the performance assessment tools contributed to the PISA (Bleuler et al., 2003) software package with standard parameters.

The hypervolume indicator w.r.t. reference set A_{ref} is defined as

$$\mathcal{I}_{\mathcal{S}, A_{\text{ref}}}(A) = \mathcal{S}_{a_{\text{ref}}}(A_{\text{ref}}) - \mathcal{S}_{a_{\text{ref}}}(A) \ ,$$

where a_{ref} denotes a (hypothetical) reference point having in each objective an objective function value worse than all considered individuals. A smaller $\mathcal{I}_{\mathcal{S}}$ is preferable. The additive unary ϵ-indicator w.r.t. reference set A_{ref} is defined as

$$\mathcal{I}_{\epsilon, A_{\text{ref}}}(A) = \inf \left\{ \epsilon \in \mathbb{R} \, | \, \forall a \in A_{\text{ref}} \, \exists a' \in A \, \forall m \in \{1, \ldots, M\} : f_m(a) + \epsilon \geq f_m(a') \right\} \ .$$

The ϵ-indicator determines the smallest offset by which the fitness values of the elements in A have to be shifted such that the resulting Pareto front covers the Pareto front of A_{ref} in the objective space. A smaller $\mathcal{I}_{\epsilon, A_{\text{ref}}}$ is preferable.

Before the performance indicators are computed, the data are normalized. We consider two slightly different ways of defining the reference sets. Assume we want to compare k algorithms on a particular optimization problem after a predefined number g of generations (this is the standard scenario when using the PISA software package). For each algorithm we have conducted t trials. We consider the non-dominated individuals of the union of all kt populations after g generations. Their objective vectors

are normalized such that for every objective the smallest and largest objective function values are mapped to 1 and 2, respectively, by an affine transformation. These individuals make up the reference set A_{ref}. The mapping to $[1, 2]^M$ is fixed and applied to all objective vectors under consideration. The reference point a_{ref} is chosen to have an objective value of 2.1 in each objective. Otherwise, if we want to compare the evolution of an indicator value over all generations, we consider the union of all populations over all algorithms, all trials, *and all generations* (i.e., $(G + 1)kt$ populations if G is the number of the final generation) for normalization and computation of A_{ref} and proceed analogously.

Knowles and Corne (2002) and Zitzler et al. (2003) studied various properties of quality indicators. Of particular interest is the relation to the "being better" definition given above. An unary quality indicator is $\not\lhd$-compatible, if a better indicator value for A than for B implies $B \not\lhd A$. An indicator is $\lhd$-complete, if $A \lhd B$ implies a better indicator value for A than for B. Both the ϵ-indicator as well as the hypervolume indicator are $\not\lhd$-compatible, the hypervolume indicator is also $\lhd$-complete. The crowding-distance measure described in section 3.2.2, which is related to the spread of solutions and not directly to the being better relation defined above, is neither $\not\lhd$-compatible nor $\lhd$-complete.

Table 4: Standard box constrained benchmark problems to be minimized.

Problem	n	Variable bounds	Objective functions	Optimal solution
FON	3	$[-4, 4]$	$f_1(\boldsymbol{x}) = 1 - \exp\left(-\sum_{i=1}^{3}\left(x_i - \frac{1}{\sqrt{3}}\right)^2\right)$	$x_1 = x_2 = x_3$
			$f_2(\boldsymbol{x}) = 1 - \exp\left(-\sum_{i=1}^{3}\left(x_i + \frac{1}{\sqrt{3}}\right)^2\right)$	$x_i \in [-1/\sqrt{3}, 1/\sqrt{3}]$
ZDT1	30	$[0, 1]$	$f_1(\boldsymbol{x}) = x_1$	$x_1 \in [0, 1]$
			$f_2(\boldsymbol{x}) = g(\boldsymbol{x})\left[1 - \sqrt{x_1/g(\boldsymbol{x})}\right]$	$x_i = 0$
			$g(\boldsymbol{x}) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	$i = 2, \ldots n$
ZDT2	30	$[0, 1]$	$f_1(\boldsymbol{x}) = x_1$	$x_1 \in [0, 1]$
			$f_2(\boldsymbol{x}) = g(\boldsymbol{x})\left[1 - (x_1/g(\boldsymbol{x}))^2\right]$	$x_i = 0$
			$g(\boldsymbol{x}) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	$i = 2, \ldots n$
ZDT3	30	$[0, 1]$	$f_1(\boldsymbol{x}) = x_1$	$x_1 \in [0, 1]$
			$f_2(\boldsymbol{x}) = g(\boldsymbol{x})\left[1 - \sqrt{x_1/g(\boldsymbol{x})} - \frac{x_1}{g(\boldsymbol{x})}\sin(10\pi x_1)\right]$	$x_i = 0$
			$g(\boldsymbol{x}) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	$i = 2, \ldots n$
ZDT4	10	$x_1 \in [0, 1]$	$f_1(\boldsymbol{x}) = x_1$	$x_1 \in [0, 1]$
		$x_i \in [-5, 5]$	$f_2(\boldsymbol{x}) = g(\boldsymbol{x})\left[1 - \sqrt{x_1/g(\boldsymbol{x})}\right]$	$x_i = 0$
		$i = 2, \ldots n$	$g(\boldsymbol{x}) = 1 + 10(n-1) + \sum_{i=2}^{n}\left[x_i^2 - 10\cos(4\pi x_i)\right]$	$i = 2, \ldots n$
ZDT6	10	$[0, 1]$	$f_1(\boldsymbol{x}) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$	$x_1 \in [0, 1]$
			$f_2(\boldsymbol{x}) = g(\boldsymbol{x})\left[1 - (f_1(\boldsymbol{x})/g(\boldsymbol{x}))^2\right]$	$x_i = 0$
			$g(\boldsymbol{x}) = 1 + 9\left[\left(\sum_{i=2}^{n} x_i\right)/(n-1)\right]^{0.25}$	$i = 2, \ldots n$

4.2 Experiments

Standard Benchmark Functions We consider three groups of test functions. The first group comprises six common benchmark problems taken from the literature, namely the function FON proposed by Fonseca and Fleming (1998) and the test functions

Table 5: Unconstrained benchmark problems to be minimized, with $a = 1000$, $b = 100$, $\boldsymbol{y} = \boldsymbol{O}_1 \boldsymbol{x}$, and $\boldsymbol{z} = \boldsymbol{O}_2 \boldsymbol{x}$, where $\boldsymbol{O}_1$ and $\boldsymbol{O}_2$ are orthogonal matrices.

Problem	n	Initial region	Objective functions	Optimal solution
ELLI_1	10	$[-10, 10]$	$f_1(\boldsymbol{y}) = \frac{1}{a^2 n} \sum_{i=1}^{n} a^{2\frac{i-1}{n-1}} y_i^2$	$y_1 = \cdots = y_n$
			$f_2(\boldsymbol{y}) = \frac{1}{a^2 n} \sum_{i=1}^{n} a^{2\frac{i-1}{n-1}} (y_i - 2)^2$	$y_1 \in [0, 2]$
ELLI_2	10	$[-10, 10]$	$f_1(\boldsymbol{y}) = \frac{1}{a^2 n} \sum_{i=1}^{n} a^{2\frac{i-1}{n-1}} y_i^2$	
			$f_2(\boldsymbol{z}) = \frac{1}{a^2 n} \sum_{i=1}^{n} a^{2\frac{i-1}{n-1}} (z_i - 2)^2$	
CIGTAB_1	10	$[-10, 10]$	$f_1(\boldsymbol{y}) = \frac{1}{a^2 n} \left[y_1^2 + \sum_{i=2}^{n-1} a y_i^2 + a^2 y_n^2 \right]$	$y_1 = \cdots = y_n$
			$f_2(\boldsymbol{y}) = \frac{1}{a^2 n} \left[(y_1 - 2)^2 + \sum_{i=2}^{n-1} a (y_i - 2)^2 + a^2 (y_n - 2)^2 \right]$	$y_1 \in [0, 2]$
CIGTAB_2	10	$[-10, 10]$	$f_1(\boldsymbol{y}) = \frac{1}{a^2 n} \left[y_1^2 + \sum_{i=2}^{n-1} a y_i^2 + a^2 y_n^2 \right]$	
			$f_2(\boldsymbol{z}) = \frac{1}{a^2 n} \left[(z_1 - 2)^2 + \sum_{i=2}^{n-1} a (z_i - 2)^2 + a^2 (z_n - 2)^2 \right]$	

ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6 proposed by Zitzler et al. (2000), see Table 4. All functions have box constraints also given in the table. In the five ZDT problems, most components of the optimal solution lie on the boundary of box constraints (this might favor the NSGA-II operators for box constraints, see Appendix A). For this reason we question the relevance of these test functions and we consider them only because they were frequently used in former studies.

Unconstrained Test Functions with Quadratic Objectives The second group of benchmarks are functions where for each objective the objective function is quadratic (a quadratic approximation close to a local optimum is reasonable for any smooth enough fitness function), see Table 5. They are of the general form $f_m(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} = \boldsymbol{x}^T \boldsymbol{O}_m^T \boldsymbol{A} \boldsymbol{O}_m \boldsymbol{x}$, where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{Q}, \boldsymbol{O}_m, \boldsymbol{A} \in \mathbb{R}^{n \times n}$ with $\boldsymbol{O}_m$ orthogonal and $\boldsymbol{A}$ diagonal and positive definite. There are two types of functions, ELLI and CIGTAB, which differ in the eigenspectrum of $\boldsymbol{Q}$. In each optimization run the coordinate system of the objective functions is changed by a random choice of $\boldsymbol{O}_m$. The *Gram-Schmidt orthogonalization procedure* can be used to construct random orthonormal basis vectors, the columns of $\boldsymbol{O}_m$, from $(\boldsymbol{0}, \boldsymbol{I})$-normally distributed random vectors. In the case of the test functions ELLI_1 and CIGTAB_1 the same rotation $\boldsymbol{O}$ is used for both objective functions (i.e., $\boldsymbol{O}_1 = \boldsymbol{O}_2$). In the more general case of ELLI_2 and CIGTAB_2 two independent rotation matrices $\boldsymbol{O}_1$ and $\boldsymbol{O}_2$ are generated, which are applied to the first and second objective function, respectively.

Generalized ZDT Problems The third group of problems shown in Table 6 are new benchmarks that generalize the ZDT problems to allow a rotation of the search space as in the second group. Already Deb (1999) proposed to use a rotation matrix together with multi-objective test functions, but his suggestion was, to the best of our knowledge, never realized for the ZDT set of benchmark problems. The boundaries of the variables make it difficult to combine rotation and the ZDT functions, because the box constraints must be satisfied *after* rotation has taken place, for example to ensure that the objectives are actually conflicting. To address this difficulty, we propose three non-linear auxiliary functions, see Table 7.

In the first function ZDT4' the rotation is applied to all but the first coordinates.

Table 6: New benchmark problems to be minimized, $\boldsymbol{y} = \boldsymbol{Ox}$, where $\boldsymbol{O} \in \mathbb{R}^{n \times n}$ is an orthogonal matrix, and $y_{\max} = 1/\max_j(|o_{1j}|)$. In the case of ZDT4', $o_{1j} = o_{j1} = 0$ for $1 < j \leq n$ and $o_{11} = 1$. For the definition of h, h_f, and h_g see Table 7.

Problem	n	Variable bounds	Objective function	Optimal solution
ZDT4'	10	$x_1 \in [0,1]$ $x_i \in [-5,5]$ $i = 2,\ldots n$	$f_1(\boldsymbol{x}) = x_1$ $f_2(\boldsymbol{x}) = g(\boldsymbol{y}) \left[1 - \sqrt{x_1/g(\boldsymbol{y})}\right]$ $g(\boldsymbol{y}) = 1 + 10(n-1) + \sum_{i=2}^n \left[y_i^2 - 10\cos(4\pi y_i)\right]$	$x_1 \in [0,1]$ $y_i = 0$ $i = 2,\ldots n$
IHR1	10	$[-1,1]$	$f_1(\boldsymbol{x}) = \|y_1\|$ $f_2(\boldsymbol{x}) = g(\boldsymbol{y}) h_f\left(1 - \sqrt{h(y_1)/g(\boldsymbol{y})}\right)$ $g(\boldsymbol{y}) = 1 + 9 \left(\sum_{i=2}^n h_g(y_i)\right)/(n-1)$	$y_1 \in [0, y_{\max}]$ $y_i = 0$ $i = 2,\ldots n$
IHR2	10	$[-1,1]$	$f_1(\boldsymbol{x}) = \|y_1\|$ $f_2(\boldsymbol{x}) = g(\boldsymbol{y}) h_f\left(1 - (y_1/g(\boldsymbol{y}))^2\right)$ $g(\boldsymbol{y}) = 1 + 9 \left(\sum_{i=2}^n h_g(y_i)\right)/(n-1)$	$y_1 \in [-y_{\max}, y_{\max}]$ $y_i = 0$ $i = 2,\ldots n$
IHR3	10	$[-1,1]$	$f_1(\boldsymbol{x}) = \|y_1\|$ $f_2(\boldsymbol{x}) = g(\boldsymbol{y}) h_f\left(1 - \sqrt{h(y_1)/g(\boldsymbol{y})} - \frac{h(y_1)}{g(\boldsymbol{y})}\sin(10\pi y_1)\right)$ $g(\boldsymbol{y}) = 1 + 9 \left(\sum_{i=2}^n h_g(y_i)\right)/(n-1)$	$y_1 \in [0, y_{\max}]$ $y_i = 0$ $i = 2,\ldots n$
IHR4	10	$[-5,5]$	$f_1(\boldsymbol{x}) = \|y_1\|$ $f_2(\boldsymbol{x}) = g(\boldsymbol{y}) h_f\left(1 - \sqrt{h(y_1)/g(\boldsymbol{y})}\right)$ $g(\boldsymbol{y}) = 1 + 10(n-1) + \sum_{i=2}^n \left[y_i^2 - 10\cos(4\pi y_i)\right]$	$y_1 \in [0, y_{\max}]$ $y_i = 0$ $i = 2,\ldots n$
IHR6	10	$[-1,1]$	$f_1(\boldsymbol{x}) = 1 - \exp(-4\|y_1\|)\sin^6(6\pi y_1)$ $f_2(\boldsymbol{x}) = g(\boldsymbol{y}) h_f\left(1 - (f_1(\boldsymbol{x})/g(\boldsymbol{y}))^2\right)$ $g(\boldsymbol{y}) = 1 + 9 \left[\left(\sum_{i=2}^n h_g(y_i)\right)/(n-1)\right]^{0.25}$	$y_1 \in [-y_{\max}, y_{\max}]$ $y_i = 0$ $i = 2,\ldots n$

Table 7: Auxiliary functions for Table 6.

$$h: \quad \mathbb{R} \rightarrow [0,1], \quad x \mapsto \left(1 + \exp\left(\frac{-x}{\sqrt{n}}\right)\right)^{-1}$$

$$h_f: \quad \mathbb{R} \rightarrow \mathbb{R}, \quad x \mapsto \begin{cases} x & \text{if } |y_1| \leq y_{\max} \\ 1 + |y_1| & \text{otherwise} \end{cases}$$

$$h_g: \quad \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}, \quad x \mapsto \frac{x^2}{|x|+0.1}$$

That is, we consider $\boldsymbol{y} = \boldsymbol{Ox}$, where $\boldsymbol{O} \in \mathbb{R}^{n \times n}$ is an orthogonal matrix with $o_{1j} = o_{j1} = 0$ for $1 < j \leq n$ and $o_{11} = 1$. In the other functions the rotation matrices are not restricted. Compared to the ZDT functions, the search space is expanded and the Pareto front is not completely located on the boundaries anymore. The lower end $y_1 = 0$ of the Pareto front is induced by the absolute value in the definition of f_1. The ends $y_1 = \pm y_{\max}$ of the Pareto front are determined by h_f, see Table 7. The value $y_{\max}$ can be chosen between 1 and $1/\max_j(|o_{1j}|)$, and in the latter case the Pareto optimal solution $y_1 = y_{\max}$ lies on the search space boundary. If $y_{\max}$ is chosen larger, up to $\sum_j |o_{1j}|$ or $5\sum_j |o_{1j}|$, respectively, parts of the Pareto front would lie on the boundary again and the Pareto front would not be linear in search space anymore. The function $h: \mathbb{R} \rightarrow [0,1]$, see Table 7, is monotonic and emulates the original variable boundary

$x_1 \in [0, 1]$. Similarly, the function $h_g : \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$ emulates the original lower variable boundary of $x_i \geq 0$ for $i = 2, \ldots, n$.

NSGA-II We compare the c-MO-CMA and the s-MO-CMA with the real-coded non-dominated sorting genetic algorithm II (NSGA-II). The real-coded NSGA-II (Deb, 2001; Deb et al., 2002) uses non-dominated sorting and the crowding-distance for selection, and real-valued genetic algorithm (GA) operators, namely polynomial mutation and simulated binary crossover (SBX). A detailed description of how these operators work is given in Appendix A based on Deb and Agrawal (1999) and Deb et al. (2003). These operators have their roots in GAs and are tailored for box constraints. Note that they are particularly well-suited when the optimal solutions lie on the boundary of a box constraint. They operate component-wise and therefore implicitly favor separability. Thus, the NSGA-II is perfectly customized for the benchmark problems in Table 4.

NSDE To our knowledge, the only other evolutionary MOO approach that is invariant against rotation and rescaling of the search space is non-dominated sorting differential evolution (NSDE). Hence, we compare our methods to NSDE as described by Iorio and Li (2005).

In every generation g of NSDE, each parent $\boldsymbol{x}_i^{(g)}$ generates one offspring $\boldsymbol{x'}_i^{(g)}$ according to

$$\boldsymbol{x'}_i^{(g)} = \boldsymbol{x}_i^{(g)} + K \left(\boldsymbol{x}_{r_{i,3}^{(g)}}^{(g)} - \boldsymbol{x}_i^{(g)} \right) + F \left(\boldsymbol{x}_{r_{i,1}^{(g)}}^{(g)} - \boldsymbol{x}_{r_{i,2}^{(g)}}^{(g)} \right) \;,$$

where $r_{i,1}^{(g)}, r_{i,2}^{(g)}, r_{i,3}^{(g)} \in \{1, \ldots, \mu\}$ are randomly chosen indices obeying $\left| \{r_{i,1}^{(g)}, r_{i,2}^{(g)}, r_{i,3}^{(g)}, i\} \right| = 4$ and K and F are real-valued parameters. The new parents are selected from the former parents and their offspring by non-dominated sorting using the crowding-distance.

The described variation rule is known as *DE/current-to-rand/1* in single-objective differential evolution (Price, 1999). The individuals in the population span the subspace of the search space reachable by the algorithm. All offspring are linearly dependent on the parents. This bears the risk that selection may lead to a degenerated population that is restricted to some subspace not containing the desired solutions. The risk depends on the relation between the dimension n of the search space and the population size. The higher the dimension and the smaller the population size the higher is this risk.

Parameter Setting and Initialization For the real-coded NSGA-II we used the same parameter setting as Deb et al. (2002). We set the mutation probability to the inverse of the genotype space dimension, $p_m = n^{-1}$, and the crossover probability to $p_c = 0.9$. The distribution indices of the crossover and mutation operator were set to $\eta_c = \eta_m = 20$. In the case of the unconstrained benchmark functions in Table 5 the boundaries of the mutation and crossover operator were set to the boundaries of the initial regions. See Appendix A for a description of the real-coded NSGA-II variation operators and their parameters.

The parameters of the NSDE were set to $K = 0.4$ and $F = 0.8$ as done by Iorio and Li (2005). Constraints are handled as in the MO-CMA-ES, see section 3.3.

We used the standard parameters of the (1+1)-CMA-ES in the MO-CMA-ES. For the functions FON, ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6 we set $\sigma^{(0)}$ equal to 60 % of $x_2^u - x_2^l$ (we rescaled the first component of ZDT4 to $[-5, 5]$). In the unconstrained problems, Table 5, we set $\sigma^{(0)}$ equal to 60 % of the initialization range of one component.

In all algorithms the population size (λ_{MO}) was set to 100 as in the study by Deb et al. (2002) to allow for a better comparison.

Table 8: Results on common benchmark problems. The upper and lower table show the median of 100 trials after 50000 evaluations of the hypervolume-indicator and the ϵ-indicator, respectively. The smallest value in each column is printed in bold, the largest in italics. The superscripts I, II, III, and IV indicate whether an algorithm is statistically significantly better than the s-MO-CMA, c-MO-CMA, NSGA-II, and NSDE, respectively (two-sided Wilcoxon rank sum test, $p < 0.001$, slanted superscripts refer to a significance level of $p < 0.01$).

algorithm	Hypervolume indicator					
	FON	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
s-MO-CMA	**0.00467**[II,III,IV]	**0.00217**[II,III,IV]	**0.00247**[II,III,IV]	**0.00105**[II,III,IV]	0.22792[IV]	**0.00051**[II,III,IV]
c-MO-CMA	0.00643[III,IV]	0.00375[IV]	0.00416[IV]	0.00186[IV]	0.22286[IV]	0.00064[IV]
NSGA-II	*0.00855*	0.00264[II,IV]	0.00316[IV]	0.00140[II,IV]	**0.00016**[I,II,IV]	0.00062[II,IV]
NSDE	0.00719[III]	*0.10872*	*0.09133*	*0.09326*	*0.80156*	*0.00121*

algorithm	ϵ-indicator					
	FON	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
s-MO-CMA	**0.00684**[II,III,IV]	**0.00459**[II,III,IV]	**0.00502**[II,III,IV]	**0.00317**[II,III,IV]	0.21138[IV]	**0.00148**[II,III,IV]
c-MO-CMA	0.01414	0.01124[IV]	0.01280[IV]	0.00870[IV]	0.20985[IV]	0.00305
NSGA-II	0.01388	0.00818[II,IV]	0.01033	0.00711[II,IV]	**0.00186**[I,II,IV]	0.00256[II,IV]
NSDE	*0.01436*	*0.08017*	*0.08533*	*0.09936*	*0.73511*	*0.00328*

Table 9: Results on new unconstrained, rotated benchmark problems. The upper and lower tables show the median of 100 trials after 50000 evaluations of the hypervolume-indicator and the ϵ-indicator, respectively.

algorithm	Hypervolume indicator			
	ELLI$_1$	ELLI$_2$	CIGTAB$_1$	CIGTAB$_2$
s-MO-CMA	**0.00345**[II,III,IV]	0.00003[III]	**0.00314**[II,III,IV]	0.00001[III,IV]
c-MO-CMA	0.00624[III,IV]	0.00003[III]	0.00545[III,IV]	**0.00000**[I,III,IV]
NSGA-II	*0.00750*	*0.00023*	0.00584[IV]	*0.00005*
NSDE	0.00687[III]	**0.00002**[I,II,III]	*0.00694*	0.00001[III]

algorithm	ϵ-indicator			
	ELLI$_1$	ELLI$_2$	CIGTAB$_1$	CIGTAB$_2$
s-MO-CMA	**0.00577**[II,III,IV]	0.00011[II,III]	**0.00561**[II,III,IV]	0.00019[II,III]
c-MO-CMA	0.01378	0.00013[III]	0.01357	0.00022[III]
NSGA-II	0.01305[IV]	*0.00049*	*0.01418*	*0.00033*
NSDE	*0.01405*	**0.00009**[I,II,III]	0.01405	**0.00018**[I,II,III]

Methods For each pair of test function and optimization algorithm 100 trials with different initial populations were conducted. For each test problem, the 100 initial populations and the randomly generated rotation matrices for the rotated problems were the same for each algorithm.

Results The characteristics of the Pareto fronts after 500 generations (50000 fitness evaluations) are shown in Tables 8, 9, and 10 for the three groups of benchmark problems. The superscripts I, II, III, and IV indicate whether a value is statistically significant compared to the s-MO-CMA, c-MO-CMA, NSGA-II, and NSDE, respectively (paired Wilcoxon rank sum test, $p < 0.001$, slanted superscripts refer to a significance level of

Table 10: Results on new, rotated, constrained benchmark problems. The upper and lower tables show the median of 100 trials after 50000 evaluations of the hypervolume-indicator and the ϵ-indicator, respectively.

algorithm	ZDT4'	IHR1	Hypervolume indicator IHR2	IHR3	IHR4	IHR6
s-MO-CMA	**0.16774**^{IV}	0.00323^{III,IV}	**0.04140**^{II,III,IV}	**0.02401**^{II,III,IV}	**0.00683**^{III,IV}	0.01093^{III,IV}
c-MO-CMA	0.18962^{IV}	**0.00284**^{III,IV}	0.04323^{III}	0.02402^{III,IV}	0.00759^{III,IV}	**0.01076**^{III,IV}
NSGA-II	0.18282^{IV}	0.01939^{IV}	*0.06383*	0.02409^{IV}	0.01725^{IV}	*0.04053*
NSDE	*0.75090*	*0.02012*	0.04289^{II,III}	*0.02415*	*0.03600*	0.02391^{III}

algorithm	ZDT4'	IHR1	ϵ-indicator IHR2	IHR3	IHR4	IHR6
s-MO-CMA	0.16626^{IV}	0.01053^{III,IV}	**0.16396**^{III,IV}	**0.03996**^{III,IV}	**0.00669**^{III,IV}	**0.02123**^{III}
c-MO-CMA	0.18465^{IV}	**0.00937**^{III,IV}	0.16428^{III,IV}	0.03996^{III,IV}	0.00746^{III,IV}	0.02170^{III}
NSGA-II	**0.16531**^{IV}	0.03147^{IV}	*0.21648*	0.04003^{IV}	0.01777^{IV}	*0.05727*
NSDE	*0.69407*	*0.03214*	0.16497^{III}	*0.04008*	*0.03321*	0.02899^{III}
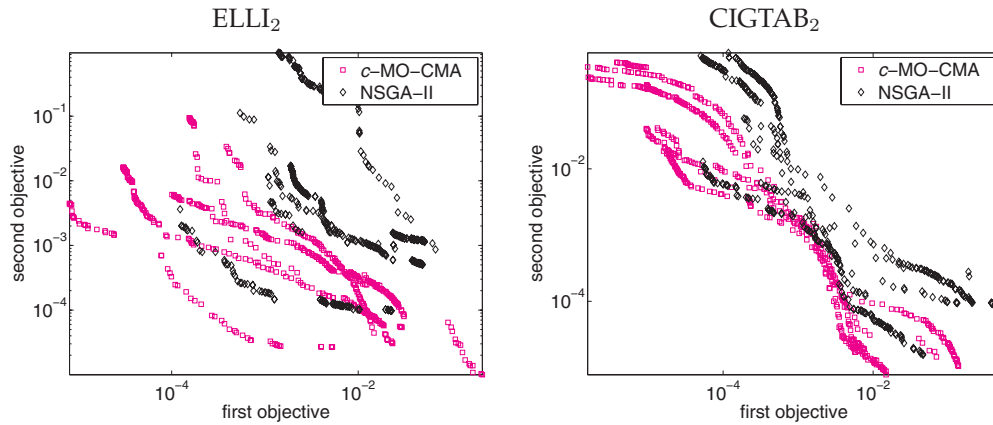
Figure 2: Population plots in objective space for c-MO-CMA and NSGA-II on the test functions ELLI$_2$ and CIGTAB$_2$ on logarithmic scale. The populations after 500 generations of the first 5 trials are shown. Note the different shapes of the Pareto fronts due to the different coordinate transformations and that s-MO-CMA, which is not shown in this figure, has a significantly better performance on these functions even compared to c-MO-CMA.

$p < 0.01$). Figure 2 shows population plots of the first five trials after 500 generations of the c-MO-CMA and NSGA-II for ELLI$_2$ and CIGTAB$_2$ in the objective space. Performance indicators for ELLI$_2$, CIGTAB$_2$, and ZDT4' after 1000 generations are given in Table 11. A corresponding population plot for c-MO-CMA and NSGA-II on ZDT4' after 1000 is shown in Figure 3. In Figure 4, the evolution of the median of the ϵ-Indicator is shown for the four test problems with quadratic objective functions. As described in Section 4.1, the reference sets and therefore the absolute values in Figure 4 are different from those in Table 9, although they are computed from the same trials.

Table 11: Results on the new benchmark problems $ELLI_2$, $CIGTAB_2$, and ZDT4' after 100000 evaluations. The upper and lower tables show the median of 100 trials of the hypervolume-indicator and the ϵ-indicator, respectively. In the corresponding figure, the populations after 1000 generations of the first 5 trials on ZDT4' are plotted.

	Hypervolume indicator		
algorithm	$ELLI_2$	$CIGTAB_2$	ZDT4'
s-MO-CMA	$0.00001^{III,IV}$	$\mathbf{0.00000}^{II,III,IV}$	$\mathbf{0.15583}^{III,IV}$
c-MO-CMA	$\mathbf{0.00001}^{I,III,IV}$	$0.00000^{III,IV}$	$0.18808^{III,IV}$
NSGA-II	*0.00018*	*0.00005*	0.22316^{IV}
NSDE	0.00002^{III}	0.00001^{III}	*0.80157*

	ϵ-indicator		
algorithm	$ELLI_2$	$CIGTAB_2$	ZDT4'
s-MO-CMA	$\mathbf{0.00009}^{II,III,IV}$	$\mathbf{0.00033}^{II,III,IV}$	$\mathbf{0.14434}^{III,IV}$
c-MO-CMA	0.00014^{III}	0.00042^{III}	$0.17247^{III,IV}$
NSGA-II	*0.00044*	*0.00073*	0.20273^{IV}
NSDE	$0.00010^{II,III}$	$0.00034^{II,III}$	*0.73926*
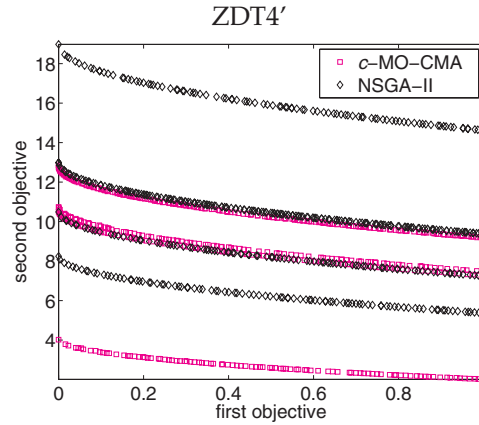
Figure 3: Populations generated by c-MO-CMA and NSGA-II after 1000 generations on ZDT4'. The first 5 of the 100 trials described in Table 11 are plotted.

Discussion   The three methods NSGA-II, NSDE, and c-MO-CMA rely on the same multi-objective selection criteria. Comparing their performance in terms of the ϵ-indicator and the hypervolume indicator allows for a fair comparison of the different variation strategies. Because the selection in s-MO-CMA almost directly aims at optimizing hypervolume, using the latter for comparisons of s-MO-CMA with the other algorithms is biased.

When looking at both the ϵ-indicator and the hypervolume indicator in the tables, s-MO-CMA is statistically significantly better than NSGA-II in all benchmark problems except ZDT4, where NSGA-II is significantly better, and ZDT4', where after 500 generations the lower values reached by the evolution strategies are not statistically significant (see below).

The multi-modal ZDT4 is separable, in the sense that the optima form a regular axis-parallel grid. The recombination operator in NSGA-II exploits this kind of sepa-
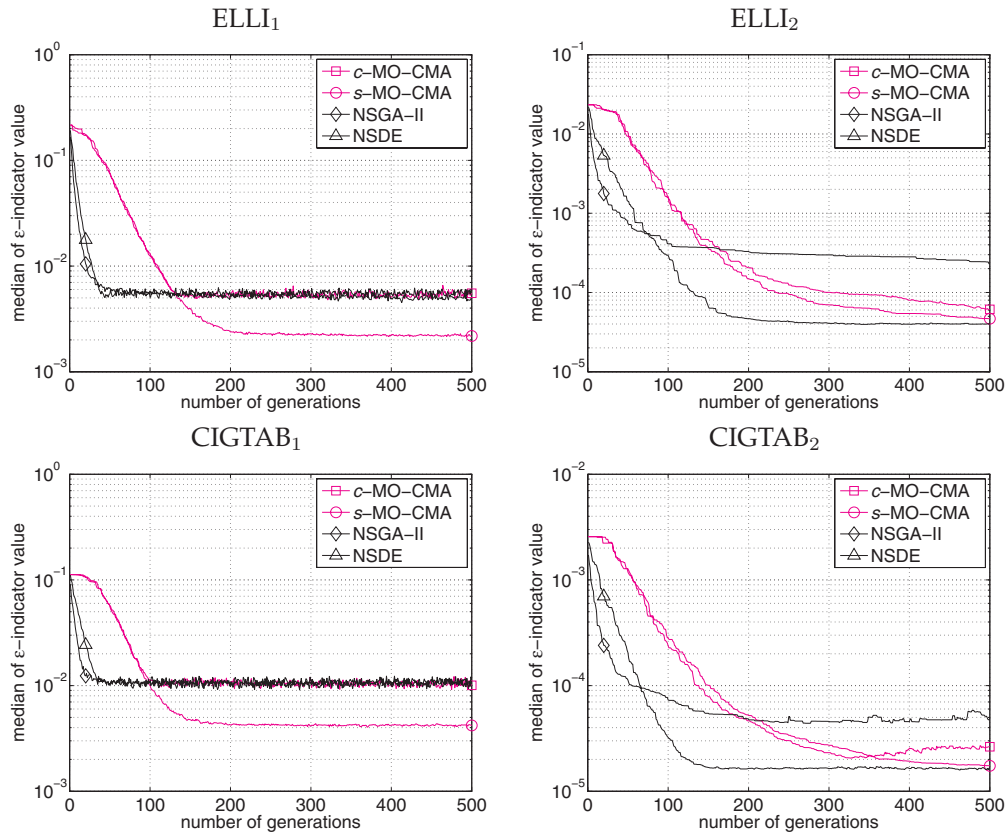
Figure 4: Simulations of the multi-objective algorithms on four rotated test functions. Shown is the median of the ϵ-indicator of the first 20 trials. The reference sets and therefore the absolute values are different compared to Table 9, see Section 4.1. In the first approximately 100 and 120 generations, the plots for c-MO-CMA and s-MO-CMA overlap in case of ELLI$_1$ and CIGTAB$_1$, respectively. After that, the plots for c-MO-CMA, NSGA-II, and NSDE can hardly be distinguished. On CIGTAB$_2$ and in particular on ELLI$_2$, the plots for c-MO-CMA and s-MO-CMA are very close. Note that ELLI$_2$ and CIGTAB$_2$ are the only benchmark problems considered in this study where the NSDE outperforms the other methods.

rability by combining (locally) optimal settings of different variables: the crossover of local optima always delivers another optimum being better in almost half of the cases and eventually the global optimum. MO-CMA-ES does not exploit this kind of separability and cannot find close to optimal solutions. When the search space is rotated such that the optima do not necessarily lie on a regular axis-parallel grid, NSGA-II is no longer superior as can be seen from the ZDT4′ results. On the contrary, the performance of the evolution strategies is not impaired. After 500 generations the differences are not significant at $p < 0.01$, because the median absolute deviation (and the variance) for the single algorithms is quite high due to the huge number of local optima of the ZDT4′ function. However, after 1000 generations the evolution strategies are significantly better than NSGA-II w.r.t. both performance indicators, see Table 11. Fig-

ure 3 shows that on ZDT4' NSGA-II suffers more under premature convergence than the evolution strategies.

The s-MO-CMA differs in two main aspects from the NSGA-II. First, the adaptation of the individual Gaussian mutation distributions uses the CMA instead of real-valued GA operators. Second, the sorting is based on the contributing hypervolume instead of the crowding-distance. To investigate the impact of these two differences, we compare c-MO-CMA and s-MO-CMA, which differ only in the selection scheme, and c-MO-CMA and NSGA-II. The latter two algorithms differ in the variation operators, but have the same selection mechanism.

Selection based on the hypervolume seems to be superior. It often leads to significantly better results in terms of the measured indicators for s-MO-CMA compared to c-MO-CMA, whereas only in two cases c-MO-CMA outperforms s-MO-CMA significantly.

On the ZDT benchmark problems, Table 8, the NSGA-II is superior to the c-MO-CMA. The indicator values are significantly better on ZDT1, ZDT3, ZDT4, and ZDT6. On these functions, NSGA-II can take advantage of the separability as described above. On FON, the c-MO-CMA is significantly better than the NSGA-II in terms of the hypervolume indicator.

On the rotated benchmark problems with one global coordinate system, $ELLI_1$ and $CIGTAB_1$, the c-MO-CMA and the NSGA-II do not differ significantly w.r.t ϵ-indicator but only w.r.t. the hypervolume indicator. The reason why the c-MO-CMA does not reach better ϵ-indicator values than the NSGA-II—despite the fact that the evolution strategy can adapt its mutation distribution to the rotated coordinate systems—lies in the selection mechanism. After an initial phase, NSGA-II and c-MO-CMA as well as NSDE suffer from the fact that the crowding-distance is not related to the being better relation defined in Section 4.1. Depending on the population size, this limits the progress of the algorithms in terms of the ϵ-indicator (and, although not obvious from all presented results, also in terms of the hypervolume indicator). This can be observed in Figure 4, left. After approximately 120 and 100 generations on $ELLI_1$ and $CIGTAB_1$, respectively, the three methods relying on the crowding-distance fluctuate around a sub-optimal level without any progress w.r.t. the ϵ-indicator. Their final performance is determined by the second sorting criterion, the corresponding plots can not be distinguished.

When looking at the problems $ELLI_2$ and $CIGTAB_2$, where, roughly speaking, the appropriate coordinate system varies along the Pareto front, both MO-CMA-ES variants clearly outperform NSGA-II. In this general case, the adaptation of arbitrary normal mutation distributions, individually along the Pareto front, seems to be of great importance. The resulting Pareto fronts are visualized in Figure 2 (note that the s-MO-CMA leads to even better results on these functions). On the IHR problems, c-MO-CMA is significantly better than NSGA-II w.r.t. both indicators. These results confirm that the invariance properties of the MO-CMA really matter, see also Figure 4, right.

Figure 4 reveals that the evolution strategies are slower in the early generations on the four test problems with quadratic objective functions compared to the other methods. It takes some time for the CMA to adapt the strategy parameters during which NSDE and NSGA-II make significantly more progress (at least for the initial CMA strategy parameters and learning rates used in this study).

Overall the NSDE performed worst of all methods. However, on $ELLI_2$ and $CIGTAB_2$, until 500 generations most indicator values are better than the corresponding values of the evolution strategies. This changes after more evaluations, see Table 11.

This result does not carry over to the other benchmarks with rotated search spaces. On some of the IHR tasks, IHR1, IHR3, and IHR4, NSDE is even significantly worse than the NSGA-II. The differential evolution algorithm in the considered form seems to have problems with higher dimensionality, as can be conjectured based on the results on ZDT1, ZDT2, and ZDT3, as well as multi-modality, as reflected by the performance on ZDT4 and ZDT4′.

The question arises whether tuning the external parameters of the MO-CMA-ES, NSDE, or NSGA-II would qualitatively affect their performance. We conducted a parameter study for the NSGA-II with different values for p_m, p_c, η_m, and η_c on the new benchmark problems with quadratic objective functions without observing remarkably improved behavior. We think that the experiments in this section well reflect the principal conceptual advantages and limitations of the algorithms.

5 Summary and Conclusions

We presented the single-objective $(1+\lambda)$-CMA-ES, an elitist evolution strategy (ES) with covariance matrix adaptation (CMA). It combines plus-selection and success rule based step size control with the powerful covariance matrix adaptation. The empirical evaluation reveals that the (1+1)-CMA-ES works reliably and that it is faster by a factor of about 1.5 on unimodal functions compared to the standard CMA-ES with comma-selection.

While the new (1+1)-CMA-ES is slightly faster than the default $(\mu/\mu, \lambda)$-CMA-ES, it is more susceptible to get trapped into sub-optimal local minima. In particular for this reason we stick to the comma-variant as default recommendation for single-objective optimization.

Based on the $(1+\lambda)$-CMA-ES we developed the $\lambda_{\mathrm{MO}} \times (1+\lambda)$-MO-CMA-ES, a multi-objective CMA-ES, which combines the strategy parameter adaptation of λ_{MO} elitist $(1+\lambda)$ strategies with multi-objective selection based on non-dominated sorting. Two variants were considered, c-MO-CMA and s-MO-CMA, using the crowding-distance and the contributing hypervolume as second sorting criterion, respectively.

The MO-CMA strategies are independent of the chosen coordinate system. Apart from the respective initializations, their behavior does not change if the search space is translated, rotated, and/or rescaled. The single-objective CMA-ES with plus-selection is additionally invariant against order-preserving transformations of the fitness function value, the MO-CMA-ES is not, because of the second level sorting criterion for selection. However, in comparison to other multi-objective evolutionary algorithms, the invariance properties of the MO-CMA-ES are an important feature, because they allow for a generalization from results obtained on benchmark functions and real world problems.

In experiments we compared c-MO-CMA, s-MO-CMA, NSGA-II, and the differential evolution approach NSDE. The s-MO-CMA algorithm appears to be the superior method. It significantly outperforms the NSGA-II on all but one of the considered test problems: the NSGA-II is faster than the s-MO-CMA only on the ZDT4 problem where the optima form a regular axis-parallel grid, because NSGA-II heavily exploits this kind of separability. However, otherwise s-MO-CMA is superior. Our results clearly show that both the new selection mechanism and in particular the covariance matrix adaptation improve the search behavior in case of the s-MO-CMA. The rotation-invariant NSDE showed the worst performance of all methods on the ZDT problems, but gave good results on the two test problems where the appropriate coordinate system varies along the Pareto front.

The ranking in the s-MO-CMA, based on the contributing hypervolume, can be computed in log-linear time in the number of individuals for two objectives, but the algorithm scales badly for an increasing number of objectives. We do not regard the bad scaling behavior as a severe drawback, in particular because in multi-objective optimization applications usually less than five objectives are considered. This is not only because the applications do not give rise to more objectives, but also because otherwise the results would be too hard to interpret (e.g., to visualize). Further, in real-world applications the costs for generating offspring and selection can often be neglected compared to the time needed for the fitness evaluation. If the contributing hypervolume cannot be used for selection because of a high number of objectives, the c-MO-CMA provides an alternative. However, one could replace the second sorting criterion with still another indicator-based measure. The underlying performance indicator should have the desired properties discussed in section 4.1 and should be efficient to compute for a higher number of objectives. For example, the ϵ-indicator is a promising candidate (Zitzler and Künzli, 2004).

In conclusion, with the caveat of the so far limited empirical data basis, the s-MO-CMA is a promising candidate to become the method of choice for real-valued non-separable optimization problems with multiple criteria given that the maximum number of fitness evaluations is not too small to allow for an adaptation of the strategy parameters.

Acknowledgments

A NSGA-II Operators

The NSGA-II uses the polynomial mutation operator for optimization problems with box constrains (Deb and Agrawal, 1999; Deb et al., 2003). Let $c = (c_1, \ldots, c_n)$ with $c_i \in [x_i^l, x_i^u], 1 \le i \le n$. The parameter $\eta_m > 0$ is called the distribution index of the mutation.

Procedure `mutatePolynomial`($c \in [x_1^l, x_1^u] \times \cdots \times [x_n^l, x_n^u]$)

---

1 **foreach** $1 \le i \le n$ **do**

2 $\quad$ $u \backsim U[0,1]$

3 $\quad$ **if** $u \le p_m$ **then**

4 $\quad\quad$ $\alpha \leftarrow \dfrac{\min\{c_i - x_i^l, x_i^u - c_i\}}{(x_i^u - x_i^l)}$

5 $\quad\quad$ $z \backsim U[0,1]$

6 $\quad\quad$ $\delta \leftarrow \begin{cases} [(2z) + (1-2z)(1-\alpha)^{\eta_m+1}]^{\frac{1}{\eta_m+1}} - 1 & \text{, if } z \le 0.5 \\ 1 - [2 \cdot (1-z) + 2 \cdot (z-0.5)(1-\alpha)^{\eta_m+1}]^{\frac{1}{\eta_m+1}} & \text{, otherwise} \end{cases}$

7 $\quad\quad$ $c_i \leftarrow c_i + \delta \cdot (x_i^u - x_i^l)$

The simulated binary crossover operator (SBX) for constrained problems (Deb and Agrawal, 1999; Deb et al., 2003) with distribution index $\eta_c > 0$ is defined as follows.

Procedure SBX $(c_1, c_2 \in [x_1^l, x_1^u] \times \cdots \times [x_n^l, x_n^u])$

---

1 **foreach** $1 \leq i \leq n$ **do**

2 $u \curvearrowleft U[0, 1[$

3   **if** $u \geq 0.5$ **then**

4 $y_1 \leftarrow \min(c_{1i}, c_{2i})$

5 $y_2 \leftarrow \max(c_{1i}, c_{2i})$

6   **if** $(y_2 - y_1) > \epsilon$ **then**

7 $\beta \leftarrow 1 + \dfrac{2}{y_2 - y_1} \cdot \min\{(y_1 - x_i^l), (x_i^u - y_2)\}$

8 $\alpha \leftarrow 2 - \beta^{-(\eta_c + 1)}$

9 $z \curvearrowleft U[0, 1]$

10 $\gamma \leftarrow \begin{cases} (z\alpha)^{\frac{1}{\eta_c + 1}} & \text{, if } z \leq \frac{1}{\alpha} \\ \left(\frac{1}{2 - z\alpha}\right)^{\frac{1}{\eta_c + 1}} & \text{, otherwise} \end{cases}$

11   **else**

12 $\gamma \leftarrow 1$

13 $[c_1]_i \leftarrow 0.5 \cdot [(y_1 + y_2) - \gamma \cdot (y_2 - y_1)]$

14 $[c_2]_i \leftarrow 0.5 \cdot [(y_1 + y_2) + \gamma \cdot (y_2 - y_1)]$

The parameter ϵ, which determines when two values are regarded as too close, is set to $\epsilon = 10^{-12}$. Due to numerical problems, these operators rather frequently hit the upper and lower bounds. In these cases, the mutation operator sets the corresponding variable x_i to some value chosen from $[x_i^l, x_i^u]$ uniformly at random.

References

Beyer, H.-G. (2001). *The Theory of Evolution Strategies*. Springer-Verlag.

Beyer, H.-G. and H.-P. Schwefel (2002). Evolution strategies: A comprehensive introduction. *Natural Computing 1*(1), 3–52.

Bleuler, S., M. Laumanns, L. Thiele, and E. Zitzler (2003). PISA – A platform and programming language independent interface for search algorithms. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele (Eds.), *Evolutionary Multi-Criterion Optimization (EMO 2003)*, Volume 2632 of *LNCS*, pp. 494 – 508. Springer-Verlag.

Büche, D., S. D. Müller, and P. Koumoutsakos (2003). Self-adaptation for multi-objective evolutionary algorithms. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele (Eds.), *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, Volume 2632 of *LNCS*, pp. 267–281. Springer-Verlag.

Coello Coello, C. A., D. A. Van Veldhuizen, and G. B. Lamont (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers.

Das, I. and J. E. Dennis (1997). A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Structural Optimization 14*(1), 63–69.

Deb, K. (1999). Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation 7*(3), 205–230.

Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley.

Deb, K. et al. (2003).
`http://www.iitk.ac.in/kangal/code/new_nsga/nsga2code.tar`.

Deb, K. and S. Agrawal (1999). A niched-penalty approach for constraint handling in genetic algorithms. In R. Albrecht, A. Dobnikar, D. Pearson, and N. Steele (Eds.), *International Conference on Artifical Neural Networks and Genetic Algorithms*, pp. 235–243. Springer-Verlag.

Deb, K., S. Agrawal, A. Pratap, and T. Meyarivan (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation 6*(2), 182–197.

Emmerich, M., N. Beume, and B. Naujoks (2005). An EMO algorithm using the hypervolume measure as selection criterion. In C. A. C. Coello, E. Zitzler, and A. H. Aguirre (Eds.), *Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, Volume 3410 of *LNCS*, pp. 62–76. Springer-Verlag.

Fonseca, C. M. and P. J. Fleming (1998). Multiobjective optimization and multiple constraint handling with evolutionary algorithms—Part II: Application example. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans 28*(1), 38–47.

Fonseca, C. M., J. D. Knowles, L. Thiele, and E. Zitzler (2005). A tutorial on the performance assessment of stochastic multiobjective optimizers. Presented at the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005).

Hansen, N. (2000). Invariance, self-adaptation and correlated mutations in evolution strategies. In *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature (PPSN VI)*, Volume 1917 of *LNCS*, pp. 355–364. Springer-Verlag.

Hansen, N. (2005). References to CMA-ES applications.
`http://www.bionik.tu-berlin.de/user/niko/cmaapplications.pdf`.

Hansen, N. (2006a). An analysis of mutative σ-self-adaptation on linear fitness functions. *Evolutionary Computation 14*(3), accepted.

Hansen, N. (2006b). The CMA evolution strategy: A comparing review. In I. I. J. A. Lozano, P. Larraãga and E. Bengoetxea (Eds.), *Towards a new evolutionary computation. Advances on estimation of distribution algorithms.*, Volume 192 of *Studies in Fuzziness and Soft Computing*. Springer-Verlag.

Hansen, N. and S. Kern (2004). Evaluating the CMA evolution strategy on multimodal test functions. In X. Yao et al. (Eds.), *Parallel Problem Solving from Nature (PPSN VIII)*, Volume 3242 of *LNCS*, pp. 282–291. Springer-Verlag.

Hansen, N., S. D. Müller, and P. Koumoutsakos (2003). Reducing the time complexity of the derandomized evolution strategy with covariance matrix adapt ation (CMA-ES). *Evolutionary Computation 11*(1), 1–18.

Hansen, N. and A. Ostermeier (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation 9*(2), 159–195.

Hansen, N., A. Ostermeier, and A. Gawelczyk (1995). On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. In L. Eshelman (Ed.), *Proceedings of the Sixth International Conference on Genetic Algorithms*, Pittsburgh, pp. 57–64. Morgan Kaufmann.

Igel, C. (2005). Multi-objective model selection for support vector machines. In C. A. C. Coello, E. Zitzler, and A. H. Aguirre (Eds.), *Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, Volume 3410 of *LNAI*, pp. 534–546. Springer-Verlag.

Iorio, A. and X. Li (2005). Solving rotated multi-objective optimization problems using differential evolution. In G. I. Webb and X. Yu (Eds.), *Proceedings of the 17th Joint Australian Conference on Artificial Intelligence*, Volume 3339 of *LNCS*, pp. 861–872. Springer-Verlag.

Kern, S., S. D. Müller, N. Hansen, D. Büche, J. Ocenasek, and P. Koumoutsakos (2004). Learning probability distributions in continuous evolutionary algorithms – a comparative review. *Natural Computing 3*, 77–112.

Knowles, J., L. Thiele, and E. Zitzler (2005, July). A tutorial on the performance assessment of stochastic multiobjective optimizers. 214, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich.

Knowles, J. D. and D. W. Corne (2002). On metrics for comparing non-dominated sets. In *Congress on Evolutionary Computation Conference (CEC 2002)*, pp. 711–716. IEEE Press.

Knuth, D. E. (1973). *The art of computer programming* (1st ed.), Volume 3: Sorting and searching, Chapter 6, pp. 451–471. Addison-Wesley.

Laumanns, M., G. Rudolph, and H.-P. Schwefel (2001). Mutation control and convergence in evolutionary multi-objective optimization. In R. Matousek and P. Osmera (Eds.), *Proceedings of the 7th International Mendel Conference on Soft Computing (MENDEL 2001)*, pp. 24–29. Brno, Czech Republic: University of Technology.

Price, K. V. (1999). An introduction to differential evolution. In D. Corne, M. Dorigo, and F. Glover (Eds.), *New Ideas in Optimization*, London, pp. 79–108. McGraw-Hill.

Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution*. Werkstatt Bionik und Evolutionstechnik. Stuttgart: Frommann-Holzboog.

Salomon, R. (1996). Reevaluating genetic algorithm performance under coordinate rotation of benchmark functions. *BioSystems 39*(3), 263–278.

Schwefel, H.-P. (1995). *Evolution and Optimum Seeking*. Sixth-Generation Computer Technology Series. John Wiley & Sons.

While, L. (2005). A new analysis of the LebMeasure algorithm for calculating hypervolume. In C. A. C. Coello, E. Zitzler, and A. H. Aguirre (Eds.), *Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, Volume 3410 of *LNCS*, pp. 326–340. Springer-Verlag.

Zitzler, E., K. Deb, and L. Thiele (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation 8*(2), 173–195.

Zitzler, E. and S. Künzli (2004). Indicator-based selection in multiobjective search. In X. Yao et al. (Eds.), *Parallel Problem Solving from Nature (PPSN VIII)*, Volume 3242 of *LNCS*, pp. 832–842. Springer-Verlag.

Zitzler, E. and L. Thiele (1998). Multiobjective optimization using evolutionary algorithms — a comparative case study. In A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel (Eds.), *Fifth International Conference on Parallel Problem Solving from Nature (PPSN-V)*, pp. 292–301. Springer-Verlag.

Zitzler, E., L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca (2003). Performance assesment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation 7*(2), 117–132.

This article has been cited by:

1. Ran Cheng, Cheng He, Yaochu Jin, Xin Yao. 2018. Model-based evolutionary algorithms: a short survey. *Complex & Intelligent Systems* **3**. . [Crossref]

2. Shahroz Khan, Erkan Gunpinar. 2018. Sampling CAD models via an extended teaching–learning-based optimization technique. *Computer-Aided Design* **100**, 52-67. [Crossref]

3. Ricardo J. Gomes, Andreia P. Guerreiro, Tobias Kuhn, Luís Paquete. 2018. Implicit enumeration strategies for the hypervolume subset selection problem. *Computers & Operations Research* . [Crossref]

4. Kimia Bazargan Lari, Ali Hamzeh. 2018. An evolutionary based framework for many-objective optimization problems. *Engineering Computations* **35**:4, 1805-1828. [Crossref]

5. Apoorva S. Shastri, Anand J. Kulkarni. 2018. Multi-Cohort Intelligence algorithm: an intra- and inter-group learning behaviour based socio-inspired optimisation methodology. *International Journal of Parallel, Emergent and Distributed Systems* **36**, 1-41. [Crossref]

6. Saúl Zapotecas-Martínez, Antonio López-Jaimes, Abel García-Nájera. 2018. LIBEA: A Lebesgue Indicator-Based Evolutionary Algorithm for multi-objective optimization. *Swarm and Evolutionary Computation* . [Crossref]

7. Rizk M. Rizk-Allah. 2018. Hybridizing sine cosine algorithm with multi-orthogonal search strategy for engineering design problems. *Journal of Computational Design and Engineering* **5**:2, 249-273. [Crossref]

8. Wenting Zhao, Lijin Wang, Yilong Yin, Bingqing Wang, Yuchun Tang. 2018. Sequential quadratic programming enhanced backtracking search algorithm. *Frontiers of Computer Science* **12**:2, 316-330. [Crossref]

9. Kwonhee Suh, Hiseak Yoon. 2018. Design Optimization of a Rear Independent Suspension for the Korean Light Tactical Vehicle. *International Journal of Automotive Technology* **19**:2, 245-252. [Crossref]

10. Meeta Kumar, Anand J. Kulkarni, Suresh Chandra Satapathy. 2018. Socio evolution & learning optimization algorithm: A socio-inspired optimization methodology. *Future Generation Computer Systems* **81**, 252-272. [Crossref]

11. J. Peeters, E. Louarroudi, B. Bogaerts, S. Sels, J. J. J. Dirckx, G. Steenackers. 2018. Active thermography setup updating for NDE: a comparative study of regression techniques and optimisation routines with high contrast parameter influences for thermal problems. *Optimization and Engineering* **19**:1, 163-185. [Crossref]

12. Teerawat Kumrai, Kaoru Ota, Mianxiong Dong, Kazuhiko Sato, Jay Kishigami. 2018. Optimising operation management for multi-micro-grids control. *IET Cyber-Physical Systems: Theory & Applications* **3**:1, 24-33. [Crossref]

13. A. L. Custódio, J. F. A. Madeira. 2018. MultiGLODS: global and local multiobjective optimization using direct search. *Journal of Global Optimization* **41**. . [Crossref]

14. Ruochen Liu, Jianxia Li, Wen Feng, Xin Yu, Licheng Jiao. 2018. A new angle-based preference selection mechanism for solving many-objective optimization problems. *Soft Computing* **32**. . [Crossref]

15. Madalina M. Drugan. Efficient Real-Parameter Single Objective Optimizer Using Hierarchical CMA-ES Solvers 131-145. [Crossref]

16. Hailong Wang, Zhongbo Hu, Yuqiu Sun, Qinghua Su, Xuewen Xia. 2018. Modified Backtracking Search Optimization Algorithm Inspired by Simulated Annealing for Constrained Engineering Optimization Problems. *Computational Intelligence and Neuroscience* **2018**, 1-27. [Crossref]

17. Michael Emmerich, Ofer M. Shir, Hao Wang. Evolution Strategies 1-31. [Crossref]

18. Fei Li, Jianchang Liu, Peiqiu Huang, Huaitao Shi. 2018. An R 2 Indicator and Decomposition Based Steady-State Evolutionary Algorithm for Many-Objective Optimization. *Mathematical Problems in Engineering* **2018**, 1-18. [Crossref]

19. Noureddine Boukhari, Fatima Debbat, Nicolas Monmarché, Mohamed Slimane. A Study on Self-adaptation in the Evolutionary Strategy Algorithm 150-160. [Crossref]

20. Michiel Haemers, Stijn Derammelaere, Clara-Mihaela Ionescu, Kurt Stockman, Jasper De Viaene, Florian Verbelen. 2018. Proportional-Integral State-Feedback Controller Optimization for a Full-Car Active Suspension Setup using a Genetic Algorithm. *IFAC-PapersOnLine* **51**:4, 1-6. [Crossref]

21. Julian Keupp, Rochus Schmid. 2018. TopoFF: MOF structure prediction using specifically optimized blueprints. *Faraday Discussions* **38**. . [Crossref]

22. Ryan Tasseff, Holly A. Jensen, Johanna Congleton, David Dai, Katharine V. Rogers, Adithya Sagar, Rodica P. Bunaciu, Andrew Yen, Jeffrey D. Varner. 2017. An Effective Model of the Retinoic Acid Induced HL-60 Differentiation Program. *Scientific Reports* **7**:1. . [Crossref]

23. Teo Ting Huan, Anand J. Kulkarni, Jeevan Kanesan, Chuah Joon Huang, Ajith Abraham. 2017. Ideology algorithm: a socio-inspired optimization methodology. *Neural Computing and Applications* **28**:S1, 845-876. [Crossref]

24. Kwonhee Suh, Hiseak Yoon. 2017. Lightweight design of airlift provision for Korean light tactical vehicle using approximate optimization. *Journal of Mechanical Science and Technology* **31**:12, 5929-5936. [Crossref]

25. Maowei He, Yabao Hu, Hanning Chen, Liling Sun, Xiaoxuan Wang, Weixing Su, Fang Liu, Xiaodan Liang, Lianbo Ma. 2017. Lifecycle Coevolution Framework for many evolutionary and swarm intelligence algorithms fusion in solving complex optimization problems. *Swarm and Evolutionary Computation* . [Crossref]

26. Leonardo C. T. Bezerra, Manuel López-Ibáñez, Thomas Stützle. A Large-Scale Experimental Evaluation of High-Performing Multi- and Many-Objective Evolutionary Algorithms. *Evolutionary Computation*, ahead of print1-36. [Abstract] [PDF] [PDF Plus]

27. S. Parisi, M. Pirotta, J. Peters. 2017. Manifold-based multi-objective policy search with sample reuse. *Neurocomputing* **263**, 3-14. [Crossref]

28. Changgu Kang, Sung-Hee Lee. 2017. Scene Reconstruction and Analysis from Motion. *Graphical Models* . [Crossref]

29. Ye Tian, Ran Cheng, Xingyi Zhang, Yaochu Jin. 2017. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum]. *IEEE Computational Intelligence Magazine* **12**:4, 73-87. [Crossref]

30. F. Capitanescu, A. Marvuglia, E. Benetto, A. Ahmadi, L. Tiruta-Barna. 2017. Linear programming-based directed local search for expensive multi-objective optimization problems: Application to drinking water production plants. *European Journal of Operational Research* **262**:1, 322-334. [Crossref]

31. Qinqin Fan, Weili Wang, Xuefeng Yan. 2017. Multi-objective differential evolution with performance-metric-based self-adaptive mutation operator for chemical and biochemical dynamic optimization problems. *Applied Soft Computing* **59**, 33-44. [Crossref]

32. U Fasel, D Keidel, G Molinari, P Ermanni. 2017. Aerostructural optimization of a morphing wing for airborne wind energy applications. *Smart Materials and Structures* **26**:9, 095043. [Crossref]

33. Shahin Rostami, Alex Shenfield. 2017. A multi-tier adaptive grid algorithm for the evolutionary multi-objective optimisation of complex problems. *Soft Computing* **21**:17, 4963-4979. [Crossref]

34. K. Schmidt, O. Wyszyński. 2017. An evolutionary strategy for $\Delta E - E$ identification. *Journal of Instrumentation* **12**:09, P09007-P09007. [Crossref]

35. Srisatja Vitayasak, Pupong Pongcharoen, Chris Hicks. 2017. A tool for solving stochastic dynamic facility layout problems with stochastic demand using either a Genetic Algorithm or modified Backtracking Search Algorithm. *International Journal of Production Economics* **190**, 146-157. [Crossref]

36. Jutta Toscano, Atreju Tauschinsky, Katrin Dulitz, Christopher J Rennick, Brianna R Heazlewood, Timothy P Softley. 2017. Zeeman deceleration beyond periodic phase space stability. *New Journal of Physics* **19**:8, 083016. [Crossref]

37. Senliang Ying, Li Li, Zheng Wang, Weikun Li, Wanliang Wang. 2017. An improved decomposition-based multiobjective evolutionary algorithm with a better balance of convergence and diversity. *Applied Soft Computing* **57**, 627-641. [Crossref]

38. Tarek Kormi, Nizar Bel Hadj Ali, Tarek Abichou, Roger Green. 2017. Estimation of landfill methane emissions using stochastic search methods. *Atmospheric Pollution Research* **8**:4, 597-605. [Crossref]

39. Jiawei Zhang, Lining Xing. A Survey of Multiobjective Evolutionary Algorithms 93-100. [Crossref]

40. Olacir R. Castro, Aurora Pozo, Jose A. Lozano, Roberto Santana. 2017. Transfer weight functions for injecting problem information in the multi-objective CMA-ES. *Memetic Computing* **9**:2, 153-180. [Crossref]

41. Shahin Rostami, Ferrante Neri. 2017. A fast hypervolume driven selection mechanism for many-objective optimisation problems. *Swarm and Evolutionary Computation* **34**, 50-67. [Crossref]

42. Siddhartha Verma, Panagiotis Hadjidoukas, Philipp Wirth, Petros Koumoutsakos. Multi-objective optimization of artificial swimmers 1037-1046. [Crossref]

43. Hui Li, Kalyanmoy Deb. Challenges for evolutionary multiobjective optimization algorithms in solving variable-length problems 2217-2224. [Crossref]

44. Yanan Sun, Gary G. Yen, Zhang Yi. Global view-based selection mechanism for many-objective evolutionary algorithms 427-434. [Crossref]

45. Olacir R. Castro, Roberto Santana, Jose A. Lozano, Aurora Pozo. Combining CMA-ES and MOEA/DD for many-objective optimization 1451-1458. [Crossref]

46. Ryoma Sano, Hernan Aguirre, Kiyoshi Tanaka. A closer look to elitism in ε-dominance many-objective optimization 2722-2729. [Crossref]

47. Matthew Sheckells, Gowtham Garimella, Marin Kobilarov. Robust policy search with applications to safe vehicle navigation 2343-2349. [Crossref]

48. Qiaoyong Jiang, Lei Wang, Jiatang Cheng, Xiaoshu Zhu, Wei Li, Yanyan Lin, Guolin Yu, Xinhong Hei, Jinwei Zhao, Xiaofeng Lu. 2017. Multi-objective differential evolution with dynamic covariance matrix learning for multi-objective optimization problems with variable linkages. *Knowledge-Based Systems* **121**, 111-128. [Crossref]

49. B. Sangamithra, P. Neelima, M. Sunil Kumar. A Memetic Algorithm for Multi Objective Vehicle Routing Problem with Time Windows 1-8. [Crossref]

50. Ilya Loshchilov. 2017. LM-CMA: An Alternative to L-BFGS for Large-Scale Black Box Optimization. *Evolutionary Computation* **25**:1, 143-171. [Abstract] [Full Text] [PDF] [PDF Plus]

51. Jogender Nagar, Douglas H. Werner. 2017. A Comparison of Three Uniquely Different State of the Art and Two Classical Multiobjective Optimization Algorithms as Applied to Electromagnetics. *IEEE Transactions on Antennas and Propagation* **65**:3, 1267-1280. [Crossref]

52. Xiang Yu, Xueqing Zhang. 2017. Multiswarm comprehensive learning particle swarm optimization for solving multiobjective optimization problems. *PLOS ONE* **12**:2, e0172033. [Crossref]

53. Hong-Yan Sang, Quan-Ke Pan, Pei-yong Duan. 2017. Self-adaptive fruit fly optimizer for global optimization. *Natural Computing* . [Crossref]

54. Simon Wessing, Rosa Pink, Kai Brandenbusch, Günter Rudolph. Toward Step-Size Adaptation in Evolutionary Multiobjective Optimization 670-684. [Crossref]

55. Daniel P. Loucks, Eelco van Beek. Data-Fitting, Evolutionary, and Qualitative Modeling 179-211. [Crossref]

56. Mateusz Spis, Adam Matecki, Patryk Maik, Adam Kurzawa, Marek Kopicki, Dominik Belter. Optimized and Reconfigurable Environment for Simulation of Legged Robots 290-299. [Crossref]

57. Meng-Dar Shieh, Yongfeng Li, Chih-Chieh Yang. 2017. Product Form Design Model Based on Multiobjective Optimization and Multicriteria Decision-Making. *Mathematical Problems in Engineering* **2017**, 1-15. [Crossref]

58. Hui Li, Qingfu Zhang, Jingda Deng. 2017. Biased Multiobjective Optimization and Decomposition Algorithm. *IEEE Transactions on Cybernetics* **47**:1, 52-66. [Crossref]

59. M. Astolfi, E. Martelli, L. Pierobon. Thermodynamic and technoeconomic optimization of Organic Rankine Cycle systems 173-249. [Crossref]

60. Richard Allmendinger, Michael T. M. Emmerich, Jussi Hakanen, Yaochu Jin, Enrico Rigoni. 2017. Surrogate-assisted multicriteria optimization: Complexities, prospective solutions, and business case. *Journal of Multi-Criteria Decision Analysis* **24**:1-2, 5-24. [Crossref]

61. Hao Wang, Yiyi Ren, André Deutz, Michael Emmerich. On Steering Dominated Points in Hypervolume Indicator Gradient Ascent for Bi-Objective Optimization 175-203. [Crossref]

62. Tobias Glasmachers. A Fast Incremental BSP Tree Archive for Non-dominated Points 252-266. [Crossref]

63. Danping Wang, Kunyuan Hu, Lianbo Ma, Maowei He, Hanning Chen. 2017. Multispecies Coevolution Particle Swarm Optimization Based on Previous Search History. *Discrete Dynamics in Nature and Society* **2017**, 1-22. [Crossref]

64. Reza Aghmasheh, Vahid Rashtchi, Ebrahim Rahimpour. 2017. Gray Box Modeling of Power Transformer Windings for Transient Studies. *IEEE Transactions on Power Delivery* 1-1. [Crossref]

65. Carlos A. Coello Coello. Recent Results and Open Problems in Evolutionary Multiobjective Optimization 3-21. [Crossref]

66. Tinkle Chugh, Karthik Sindhya, Jussi Hakanen, Kaisa Miettinen. 2017. A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms. *Soft Computing* . [Crossref]

67. Qiuzhen Lin, Songbai Liu, Qingling Zhu, Chaoyu Tang, Ruizhen Song, Jianyong Chen, Carlos Artemio Coello Coello, Ka-Chun Wong, Jun Zhang. 2017. Particle Swarm Optimization with a Balanceable Fitness Estimation for Many-objective Optimization Problems. *IEEE Transactions on Evolutionary Computation* 1-1. [Crossref]

68. Iris Kriest, Volkmar Sauerland, Samar Khatiwala, Anand Srivastav, Andreas Oschlies. 2017. Calibrating a global three-dimensional biogeochemical ocean model (MOPS-1.0). *Geoscientific Model Development* **10**:1, 127-154. [Crossref]

69. Xinye Cai, Zhiwei Mei, Zhun Fan, Qingfu Zhang. 2017. A Constrained Decomposition Approach with Grids for Evolutionary Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* 1-1. [Crossref]

70. Maryam Varmazyar, Nicholas Haritos, Michael Kirley. 2016. A wavelet-based Bayesian damage identification technique using an evolutionary algorithm. *Australian Journal of Structural Engineering* **130**, 1-17. [Crossref]

71. . Multiobjective Optimization of Green Sand Mold Product 159-180. [Crossref]

72. M. Prauzek, P. Krömer, J. Rodway, P. Musilek. 2016. Differential evolution of fuzzy controller for environmentally-powered wireless sensors. *Applied Soft Computing* **48**, 193-206. [Crossref]

73. S. Rodrigues, P. Bauer, Peter A.N. Bosman. 2016. Multi-objective optimization of wind farm layouts – Complexity, constraint handling and scalability. *Renewable and Sustainable Energy Reviews* **65**, 587-609. [Crossref]

74. Markus Wagner. 2016. Nested Multi- and Many-Objective Optimization of Team Track Pursuit Cycling. *Frontiers in Applied Mathematics and Statistics* **2**. . [Crossref]

75. Adham Salih, Amiram Moshaiov. Multi-objective neuro-evolution: Should the main reproduction mechanism be crossover or mutation? 004585-004590. [Crossref]

76. Miqing Li, Shengxiang Yang, Xiaohui Liu. 2016. Pareto or Non-Pareto: Bi-Criterion Evolution in Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* **20**:5, 645-665. [Crossref]

77. Mamatha Venugopal, Ram Mohan Vasu, Debasish Roy. 2016. A stochastically evolving non-local search and solutions to inverse problems with sparse data. *Probabilistic Engineering Mechanics* **46**, 37-47. [Crossref]

78. Shahin Rostami, Ferrante Neri. 2016. Covariance matrix adaptation pareto archived evolution strategy with hypervolume-sorted adaptive grid algorithm. *Integrated Computer-Aided Engineering* **23**:4, 313-329. [Crossref]

79. Tobias Kuhn, Carlos M. Fonseca, Luís Paquete, Stefan Ruzika, Miguel M. Duarte, José Rui Figueira. 2016. Hypervolume Subset Selection in Two Dimensions: Formulations and Algorithms. *Evolutionary Computation* **24**:3, 411-425. [Abstract] [Full Text] [PDF] [PDF Plus]

80. Ruben Müller, Niels Schütze. 2016. Multi-objective optimization of multi-purpose multi-reservoir systems under high reliability constraints. *Environmental Earth Sciences* **75**:18. . [Crossref]

81. Omer Abramovich, Amiram Moshaiov. Multi-objective topology and weight evolution of neuro-controllers 670-677. [Crossref]

82. Luis Marti, Harold D. de Mello, Nayat Sanchez-Pi, Marley Vellasco. SMS-EDA-MEC: Extending Copula-based EDAs to multi-objective optimization 3726-3733. [Crossref]

83. Steffen Limmer, Dietmar Fey. Investigation of strategies for an increasing population size in multi-objective CMA-ES 476-483. [Crossref]

84. Aras Ahmadi, Ligia Tiruta-Barna, Florin Capitanescu, Enrico Benetto, Antonino Marvuglia. 2016. An archive-based multi-objective evolutionary algorithm with adaptive search space partitioning to deal with expensive optimization problems: Application to process eco-design. *Computers & Chemical Engineering* **87**, 95-110. [Crossref]

85. Sawyer D Campbell, Jogender Nagar, Donovan E Brocker, Douglas H Werner. 2016. On the use of surrogate models in the analytical decompositions of refractive index gradients obtained through quasiconformal transformation optics. *Journal of Optics* **18**:4, 044019. [Crossref]

86. Yuan Yuan, Hua Xu, Bo Wang, Bo Zhang, Xin Yao. 2016. Balancing Convergence and Diversity in Decomposition-Based Many-Objective Optimizers. *IEEE Transactions on Evolutionary Computation* **20**:2, 180-198. [Crossref]

87. Aras Ahmadi. 2016. Memory-based adaptive partitioning (MAP) of search space for the enhancement of convergence in Pareto-based multi-objective evolutionary algorithms. *Applied Soft Computing* **41**, 400-417. [Crossref]

88. Adriana Menchaca-Mendez, Carlos A. Coello Coello. 2016. Selection mechanisms based on the maximin fitness function to solve multi-objective optimization problems. *Information Sciences* **332**, 131-152. [Crossref]

89. Yuan Yuan, Hua Xu, Bo Wang, Xin Yao. 2016. A New Dominance Relation-Based Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation* **20**:1, 16-37. [Crossref]

90. Aimin Zhou, Qingfu Zhang. 2016. Are All the Subproblems Equally Important? Resource Allocation in Decomposition-Based Multiobjective Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation* **20**:1, 52-64. [Crossref]

91. Feng Zhao, Xianghua Xie. 2016. Energy minimization in medical image analysis: Methodologies and applications. *International Journal for Numerical Methods in Biomedical Engineering* **32**:2, n/a-n/a. [Crossref]

92. Mudong Li, Hui Zhao, Xingwei Weng, Tong Han. 2016. Cognitive behavior optimization algorithm for solving optimization problems. *Applied Soft Computing* **39**, 199-222. [Crossref]

93. Lei Wu, Wensheng Xiao, Liang Zhang, Qi Liu, Jingli Wang. 2016. An improved fruit fly optimization algorithm based on selecting evolutionary direction intelligently. *International Journal of Computational Intelligence Systems* **9**:1, 80-90. [Crossref]

94. Dídac Rodríguez Arbonès, Boyin Ding, Nataliia Y. Sergiienko, Markus Wagner. Fast and Effective Multi-objective Optimisation of Submerged Wave Energy Converters 675-685. [Crossref]

95. Alan Díaz-Manríquez, Gregorio Toscano, Jose Hugo Barron-Zambrano, Edgar Tello-Leal. 2016. R 2 -Based Multi/Many-Objective Particle Swarm Optimization. *Computational Intelligence and Neuroscience* **2016**, 1-10. [Crossref]

96. Yuanxia Shen, Linna Wei, Chuanhua Zeng, Jian Chen. 2016. Particle Swarm Optimization with Double Learning Patterns. *Computational Intelligence and Neuroscience* **2016**, 1-19. [Crossref]

97. Ruben Müller, Henok Y. Gebretsadik, Niels Schütze. 2016. Towards an optimal integrated reservoir system management for the Awash River Basin, Ethiopia. *Proceedings of the International Association of Hydrological Sciences* **373**, 215-219. [Crossref]

98. Ali Ahrari, Masoud Shariat-Panahi. 2015. An improved evolution strategy with adaptive population size. *Optimization* **64**:12, 2567-2586. [Crossref]

99. Takafumi Moriya, Tomohiro Tanaka, Takahiro Shinozaki, Shinji Watanabe, Kevin Duh. Automation of system building for state-of-the-art large vocabulary speech recognition using evolution strategy 610-616. [Crossref]

100. Chunjiang Zhang, Qun Lin, Liang Gao, Xinyu Li. 2015. Backtracking Search Algorithm with three constraint handling methods for constrained optimization problems. *Expert Systems with Applications* **42**:21, 7831-7845. [Crossref]

101. Hussain Shareef, Ammar Hussein Mutlag, Azah Mohamed. 2015. A novel approach for fuzzy logic PV inverter controller optimization using lightning search algorithm. *Neurocomputing* **168**, 435-453. [Crossref]

102. Shouheng Tuo, Junying Zhang, Longquan Yong, Xiguo Yuan, Baobao Liu, Xiaoyang Xu, Fang'an Deng. 2015. A harmony search algorithm for high-dimensional multimodal optimization problems. *Digital Signal Processing* **46**, 151-163. [Crossref]

103. Miqing Li, Shengxiang Yang, Xiaohui Liu. 2015. Bi-goal evolution for many-objective optimization problems. *Artificial Intelligence* **228**, 45-65. [Crossref]

104. Martin Pilat, Roman Neruda. Hypervolume-Based Surrogate Model for MO-CMA-ES 604-611. [Crossref]

105. Hussain Shareef, Ahmad Asrul Ibrahim, Ammar Hussein Mutlag. 2015. Lightning search algorithm. *Applied Soft Computing* **36**, 315-333. [Crossref]

106. Chun-an Liu, Huamin Jia. 2015. Dynamic Multiobjective Evolutionary Algorithm With Two Stages Evolution Operation. *Intelligent Automation & Soft Computing* **21**:4, 575-588. [Crossref]

107. Xiaofang Yuan, Ting Zhang, Yongzhong Xiang, Xiangshan Dai. 2015. Parallel chaos optimization algorithm with migration and merging operation. *Applied Soft Computing* **35**, 591-604. [Crossref]

108. Lei Cai, Shiru Qu, Yuan Yuan, Xin Yao. 2015. A clustering-ranking method for many-objective optimization. *Applied Soft Computing* **35**, 681-694. [Crossref]

109. Ke Li, Kalyanmoy Deb, Qingfu Zhang, Sam Kwong. 2015. An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition. *IEEE Transactions on Evolutionary Computation* **19**:5, 694-716. [Crossref]

110. Zhengping Liang, Ruizhen Song, Qiuzhen Lin, Zhihua Du, Jianyong Chen, Zhong Ming, Jianping Yu. 2015. A double-module immune algorithm for multi-objective optimization problems. *Applied Soft Computing* **35**, 161-174. [Crossref]

111. Dimo Brockhoff, Tobias Wagner, Heike Trautmann. 2015. 2 Indicator-Based Multiobjective Search. *Evolutionary Computation* **23**:3, 369-395. [Abstract] [Full Text] [PDF] [PDF Plus]

112. Daniele Prada, Marco Bellini, Ivica Stevanovic, Laurent Lemaitre, James Victory, Jan Vobecky, Riccardo Sacco, Peter O. Lauritzen. 2015. On the Performance of Multiobjective Evolutionary Algorithms in Automatic Parameter Extraction of Power Diodes. *IEEE Transactions on Power Electronics* **30**:9, 4986-4997. [Crossref]

113. Adriana Menchaca-Mendez, Carlos A. Coello Coello. 2015. An alternative hypervolume-based selection mechanism for multi-objective evolutionary algorithms. *Soft Computing* . [Crossref]

114. Sen Bong Gee, Kay Chen Tan, Vui Ann Shim, Nikhil R. Pal. 2015. Online Diversity Assessment in Evolutionary Multiobjective Optimization: A Geometrical Perspective. *IEEE Transactions on Evolutionary Computation* **19**:4, 542-559. [Crossref]

115. Hojjat Emami, Farnaz Derakhshan. 2015. Election algorithm: A new socio-politically inspired strategy. *AI Communications* **28**:3, 591-603. [Crossref]

116. Jogender Nagar, Sawyer D. Campbell, Douglas H. Werner. Multi-objective optimization for GRIN lens design 1326-1327. [Crossref]

117. Jogender Nagar, Douglas H. Werner. A comparison of multi-objective optimizers using a microstrip patch antenna test problem 1332-1333. [Crossref]

118. T. Ganesan, I. Elamvazuthi, P. Vasant. 2015. Multiobjective design optimization of a nano-CMOS voltage-controlled oscillator using game theoretic-differential evolution. *Applied Soft Computing* **32**, 293-299. [Crossref]

119. Markus Wagner, Karl Bringmann, Tobias Friedrich, Frank Neumann. 2015. Efficient optimization of many objectives by approximation-guided evolution. *European Journal of Operational Research* **243**:2, 465-479. [Crossref]

120. Tirtharaj Dash, Prabhat K. Sahu. 2015. Gradient gravitational search: An efficient metaheuristic algorithm for global optimization. *Journal of Computational Chemistry* **36**:14, 1060-1068. [Crossref]

121. Rutuparna Panda, Manoj Kumar Naik. 2015. A novel adaptive crossover bacterial foraging optimization algorithm for linear discriminant analysis based face recognition. *Applied Soft Computing* **30**, 722-736. [Crossref]

122. Attia El-Fergany. 2015. Study impact of various load models on DG placement and sizing using backtracking search algorithm. *Applied Soft Computing* **30**, 803-811. [Crossref]

123. Yuki Yazawa, Hernan Aguirre, Akira Oyama, Kiyoshi Tanaka. Evolutionary many-objective optimization using dynamic ε-Hoods and Chebyshev function 1861-1868. [Crossref]

124. Xiaofang Yuan, Xiangshan Dai, Lianghong Wu. 2015. A mutative-scale pseudo-parallel chaos optimization algorithm. *Soft Computing* **19**:5, 1215-1227. [Crossref]

125. Giorgos Karafotias, Mark Hoogendoorn, A. E. Eiben. 2015. Parameter Control in Evolutionary Algorithms: Trends and Challenges. *IEEE Transactions on Evolutionary Computation* **19**:2, 167-187. [Crossref]

126. Pu Wang, Michael Emmerich, Rui Li, Ke Tang, Thomas Back, Xin Yao. 2015. Convex Hull-Based Multiobjective Genetic Programming for Maximizing Receiver Operating Characteristic Performance. *IEEE Transactions on Evolutionary Computation* **19**:2, 188-200. [Crossref]

127. Tobias Friedrich, Frank Neumann, Christian Thyssen. 2015. Multiplicative Approximations, Optimal Hypervolume Distributions, and the Choice of the Reference Point. *Evolutionary Computation* **23**:1, 131-159. [Abstract] [Full Text] [PDF] [PDF Plus]

128. Maxim V. Ivanov, Marat R. Talipov, Qadir K. Timerghazin. 2015. Genetic Algorithm Optimization of Point Charges in Force Field Development: Challenges and Insights. *The Journal of Physical Chemistry A* **119**:8, 1422-1434. [Crossref]

129. Lothar Thiele. Indicator-Based Selection 983-994. [Crossref]

130. Oswin Krause, Christian Igel. A More Efficient Rank-one Covariance Matrix Update for Evolution Strategies 129-136. [Crossref]

131. Lijin Wang, Yiwen Zhong, Yilong Yin, Wenting Zhao, Binqing Wang, Yulong Xu. 2015. A Hybrid Backtracking Search Optimization Algorithm with Differential Evolution. *Mathematical Problems in Engineering* **2015**, 1-16. [Crossref]

132. Nikolaus Hansen, Dirk V. Arnold, Anne Auger. Evolution Strategies 871-898. [Crossref]

133. S.F. Rodrigues, R. Teixeira Pinto, M. Soleimanzadeh, Peter A.N. Bosman, P. Bauer. 2015. Wake losses optimization of offshore wind farms with moveable floating wind turbines. *Energy Conversion and Management* **89**, 933-941. [Crossref]

134. Leonardo Bezerra, Manuel Lopez-Ibanez, Thomas Stuetzle. 2015. Automatic Component-Wise Design of Multi-Objective Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation* 1-1. [Crossref]

135. Saúl Zapotecas-Martínez, Bilel Derbel, Arnaud Liefooghe, Dimo Brockhoff, Hernán E. Aguirre, Kiyoshi Tanaka. Injecting CMA-ES into MOEA/D 783-790. [Crossref]

136. Wensheng Xiao, Lei Wu, Xue Tian, Jingli Wang. 2015. Applying a New Adaptive Genetic Algorithm to Study the Layout of Drilling Equipment in Semisubmersible Drilling Platforms. *Mathematical Problems in Engineering* **2015**, 1-9. [Crossref]

137. Siwei Jiang, Jie Zhang, Yew-Soon Ong. 2014. Multiobjective optimization based on reputation. *Information Sciences* **286**, 125-146. [Crossref]

138. A. Mereuta, S. Aupetit, N. Monmarché, M. Slimane. 2014. Web Page Textual Color Contrast Compensation for CVD Users Using Optimization Methods. *Journal of Mathematical Modelling and Algorithms in Operations Research* **13**:4, 447-470. [Crossref]

139. Haibin Duan, Qinan Luo. 2014. Adaptive Backtracking Search Algorithm for Induction Magnetometer Optimization. *IEEE Transactions on Magnetics* **50**:12, 1-6. [Crossref]

140. Karl Bringmann, Tobias Friedrich. 2014. Convergence of Hypervolume-Based Archiving Algorithms. *IEEE Transactions on Evolutionary Computation* **18**:5, 643-657. [Crossref]

141. George Piliounis, Nikos D. Lagaros. 2014. Reliability analysis of geostructures based on metaheuristic optimization. *Applied Soft Computing* **22**, 544-565. [Crossref]

142. Aris Lanaridis, Andreas Stafylopatis. 2014. An artificial immune network for multiobjective optimization problems. *Engineering Optimization* **46**:8, 1008-1031. [Crossref]

143. Miqing Li, Shengxiang Yang, Ke Li, Xiaohui Liu. 2014. Evolutionary Algorithms With Segment-Based Search for Multiobjective Optimization Problems. *IEEE Transactions on Cybernetics* **44**:8, 1295-1313. [Crossref]

144. Amiram Moshaiov, Omer Abramovich. Is MO-CMA-ES superior to NSGA-II for the evolution of multi-objective neuro-controllers? 2809-2816. [Crossref]

145. Diego Perez, Edward Powley, Daniel Whitehouse, Spyridon Samothrakis, Simon Lucas, Peter I. Cowling. The 2013 Multi-objective Physical Travelling Salesman Problem Competition 2314-2321. [Crossref]

146. Hao Liu, Guiyan Ding, Bing Wang. 2014. Bare-bones particle swarm optimization with disruption operator. *Applied Mathematics and Computation* **238**, 106-122. [Crossref]

147. Marcio Weck Pereira, Guenther Schwedersky Neto, Mauro Roisenberg. A topological niching covariance matrix adaptation for multimodal optimization 2562-2569. [Crossref]

148. Miqing Li, Shengxiang Yang, Jinhua Zheng, Xiaohui Liu. 2014. ETEA: A Euclidean Minimum Spanning Tree-Based Evolutionary Algorithm for Multi-Objective Optimization. *Evolutionary Computation* **22**:2, 189-230. [Abstract] [Full Text] [PDF] [PDF Plus]

149. Nikos D. Lagaros. 2014. A general purpose real-world structural design optimization computing platform. *Structural and Multidisciplinary Optimization* **49**:6, 1047-1066. [Crossref]

150. Constantin Brif, Matthew D Grace, Mohan Sarovar, Kevin C Young. 2014. Exploring adiabatic quantum trajectories via optimal control. *New Journal of Physics* **16**:6, 065013. [Crossref]

151. Hyoungjin Kim, Meng-Sing Liou. 2014. Adaptive directional local search strategy for hybrid evolutionary multiobjective optimization. *Applied Soft Computing* **19**, 290-311. [Crossref]

152. Miqing Li, Shengxiang Yang, Xiaohui Liu. 2014. Shift-Based Density Estimation for Pareto-Based Algorithms in Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation* **18**:3, 348-365. [Crossref]

153. Ofer M. Shir, Jonathan Roslund, Darrell Whitley, Herschel Rabitz. 2014. Efficient retrieval of landscape Hessian: Forced optimal covariance adaptive learning. *Physical Review E* **89**:6. . [Crossref]

154. Xiaofang Yuan, Xiangshan Dai, Jingyi Zhao, Qian He. 2014. On a novel multi-swarm fruit fly optimization algorithm and its application. *Applied Mathematics and Computation* **233**, 260-271. [Crossref]

155. Xiaofang Yuan, Jingyi Zhao, Yimin Yang, Yaonan Wang. 2014. Hybrid parallel chaos optimization algorithm with harmony search algorithm. *Applied Soft Computing* **17**, 12-22. [Crossref]

156. Yu Lei, Maoguo Gong, Jun Zhang, Wei Li, Licheng Jiao. 2014. Resource allocation model and double-sphere crowding distance for evolutionary multi-objective optimization. *European Journal of Operational Research* **234**:1, 197-208. [Crossref]

157. Sowmya Sree Bhagavatula, Sriram G. Sanjeevi, Divya Kumar, Chitranjan Kumar Yadav. Multi-objective indicator based evolutionary algorithm for portfolio optimization 1206-1210. [Crossref]

158. Andrea Maesani, Pradeep Ruben Fernando, Dario Floreano. 2014. Artificial Evolution by Viability Rather than Competition. *PLoS ONE* **9**:1, e86831. [Crossref]

159. Alex Shenfield, Peter J. Fleming. 2014. Multi-objective evolutionary design of robust controllers on the grid. *Engineering Applications of Artificial Intelligence* **27**, 17-27. [Crossref]

160. D.T. Rwasoka, C.E. Madamombe, W. Gumindoga, A.T. Kabobah. 2014. Calibration, validation, parameter indentifiability and uncertainty analysis of a 2 – parameter parsimonious monthly rainfall-runoff model in two catchments in Zimbabwe. *Physics and Chemistry of the Earth, Parts A/B/C* **67-69**, 36-46. [Crossref]

161. Alexander Kout, Heinrich Müller. 2014. Quantitative improvement of tool impact paths defined by isolines of scalar functions on triangular mesh workpiece surfaces. *The International Journal of Advanced Manufacturing Technology* **70**:1-4, 237-255. [Crossref]

162. Aleksandra P. Dabkowska, David J. Barlow, Luke A. Clifton, Arwel V. Hughes, John R. P. Webster, Rebecca J. Green, Peter J. Quinn, M. Jayne Lawrence. 2014. Calcium-mediated binding of DNA to 1,2-distearoyl-sn-glycero-3-phosphocholine-containing mixed lipid monolayers. *Soft Matter* **10**:11, 1685. [Crossref]

163. Ruben Müller. 2014. überprüfung der zukünftigen Leistung eines Talsperrensystems mittels Mehrzieloptimierung. *WASSERWIRTSCHAFT* **104**:1-2, 50-53. [Crossref]

164. Masoud Asadzadeh, Bryan Tolson. 2013. Pareto archived dynamically dimensioned search with hypervolume-based selection for multi-objective optimization. *Engineering Optimization* **45**:12, 1489-1509. [Crossref]

165. Silvio Rodrigues, Rodrigo Teixeira Pinto, Pavol Bauer, Jan Pierik. 2013. Optimal Power Flow Control of VSC-Based Multiterminal DC Network for Offshore Wind Integration in the North Sea. *IEEE Journal of Emerging and Selected Topics in Power Electronics* **1**:4, 260-268. [Crossref]

166. Yanping Liu, Erik D. Reichle, Ding-Guo Gao. 2013. Using Reinforcement Learning to Examine Dynamic Attention Allocation During Reading. *Cognitive Science* **37**:8, 1507-1540. [Crossref]

167. So-Youn Park, Ju-Jang Lee. A multi-objective covariance matrix adaptation evolutionary strategy based on decomposition for analog circuit design 6365-6370. [Crossref]

168. Karl Bringmann, Tobias Friedrich, Christian Igel, Thomas Voß. 2013. Speeding up many-objective optimization by Monte Carlo approximations. *Artificial Intelligence* **204**, 22-29. [Crossref]

169. Ali Osman Kusakci, Mehmet Can. 2013. An adaptive penalty based covariance matrix adaptation–evolution strategy. *Computers & Operations Research* **40**:10, 2398-2417. [Crossref]

170. Christoph Krimpmann, Jan Braun, Frank Hoffmann, Torsten Bertram. Active covariance matrix adaptation for multi-objective CMA-ES 189-194. [Crossref]

171. Leonard Judt, Olaf Mersmann, Boris Naujoks. 2013. Non-monotonicity of Observed Hypervolume in 1-Greedy S -Metric Selection. *Journal of Multi-Criteria Decision Analysis* **20**:5-6, 277-290. [Crossref]

172. Mansour Sheikhan, S. Amir Ghoreishi. 2013. Application of covariance matrix adaptation–evolution strategy to optimal control of hepatitis B infection. *Neural Computing and Applications* **23**:3-4, 881-894. [Crossref]

173. Dimo Brockhoff, Johannes Bader, Lothar Thiele, Eckart Zitzler. 2013. Directed Multiobjective Optimization Based on the Weighted Hypervolume Indicator. *Journal of Multi-Criteria Decision Analysis* **20**:5-6, 291-317. [Crossref]

174. Amiram Moshaiov, Mor Elias. Variable-based ε — PAES with adaptive fertility rate 159-166. [Crossref]

175. Yourim Yoon, Yong-Hyuk Kim. 2013. Geometricity of genetic operators for real-coded representation. *Applied Mathematics and Computation* **219**:23, 10915-10927. [Crossref]

176. Edward J. Powley, Daniel Whitehouse, Peter I. Cowling. Monte Carlo Tree Search with macro-actions and heuristic route planning for the Multiobjective Physical Travelling Salesman Problem 1-8. [Crossref]

177. Adriana Menchaca-Mendez, Carlos A. Coello Coello. A new selection mechanism based on hypervolume and its locality property 924-931. [Crossref]

178. Markus Wagner, Tobias Friedrich. Efficient parent selection for Approximation-Guided Evolutionary multi-objective optimization 1846-1853. [Crossref]

179. Alexander W. Churchill, Phil Husbands, Andrew Philippides. Tool sequence optimization using synchronous and asynchronous parallel multi-objective evolutionary algorithms with heterogeneous evaluations 2924-2931. [Crossref]

180. Gaetan Marceau Caron, Pierre Saveant, Marc Schoenauer. Multiobjective tactical planning under uncertainty for air traffic flow and capacity management 1548-1555. [Crossref]

181. Alexander W. Churchill, Phil Husbands, Andrew Philippides. Multi-objective tool sequence and parameter optimization for rough milling applications 1475-1482. [Crossref]

182. Alan Diaz-Manriquez, Gregorio Toscano-Pulido, Carlos A. Coello Coello, Ricardo Landa-Becerra. A ranking method based on the R2 indicator for many-objective optimization 1523-1530. [Crossref]

183. Raquel Hernandez Gomez, Carlos A. Coello Coello. MOMBI: A new metaheuristic for many-objective optimization based on the R2 indicator 2488-2495. [Crossref]

184. Alfredo Mendoza-Gonzalez, Eunice Ponce-de-Leon, Elva Diaz-Diaz. Classification scheme of multi-objective Estimation of Distribution Algorithms 3051-3057. [Crossref]

185. Tobias Friedrich, Trent Kroeger, Frank Neumann. 2013. Weighted preferences in evolutionary multi-objective optimization. *International Journal of Machine Learning and Cybernetics* **4**:2, 139-148. [Crossref]

186. Pinar Civicioglu. 2013. Backtracking Search Optimization Algorithm for numerical optimization problems. *Applied Mathematics and Computation* **219**:15, 8121-8144. [Crossref]

187. Ruochen Liu, Xiao Wang, Jing Liu, Lingfen Fang, Licheng Jiao. 2013. A preference multi-objective optimization based on adaptive rank clone and differential evolution. *Natural Computing* **12**:1, 109-132. [Crossref]

188. Daniele Muraro, Rui Dilão. 2013. A parallel multi-objective optimization algorithm for the calibration of mathematical models. *Swarm and Evolutionary Computation* **8**, 13-25. [Crossref]

189. Karl Bringmann, Tobias Friedrich. 2013. Approximation quality of the hypervolume indicator. *Artificial Intelligence* **195**, 265-290. [Crossref]

190. Chara Ch. Mitropoulou, Yiannis Fourkiotis, Nikos D. Lagaros, Matthew G. Karlaftis. Evolution Strategies-Based Metaheuristics in Structural Design Optimization 79-102. [Crossref]

191. T. Ganesan, I. Elamvazuthi, Ku Zilati Ku Shaari, P. Vasant. 2013. An Algorithmic Framework for Multiobjective Optimization. *The Scientific World Journal* **2013**, 1-11. [Crossref]

192. Ofer M. Shir, Jonathan Roslund, Zaki Leghtas, Herschel Rabitz. 2012. Quantum control experiments as a testbed for evolutionary multi-objective algorithms. *Genetic Programming and Evolvable Machines* **13**:4, 445-491. [Crossref]

193. Yong Wang, Jian Xiang, Zixing Cai. 2012. A regularity model-based multiobjective estimation of distribution algorithm with reducing redundant cluster operator. *Applied Soft Computing* **12**:11, 3526-3538. [Crossref]

194. Rudolf Berghammer, Tobias Friedrich, Frank Neumann. 2012. Convergence of set-based multi-objective optimization, indicators and deteriorative cycles. *Theoretical Computer Science* **456**, 2-17. [Crossref]

195. Shahin Rostami, Alex Shenfield. CMA-PAES: Pareto archived evolution strategy using covariance matrix adaptation for Multi-Objective Optimisation 1-8. [Crossref]

196. A. P. Dabkowska, D. J. Barlow, R. A. Campbell, A. V. Hughes, P. J. Quinn, M. J. Lawrence. 2012. Effect of Helper Lipids on the Interaction of DNA with Cationic Lipid Monolayers Studied by Specular Neutron Reflection. *Biomacromolecules* **13**:8, 2391-2401. [Crossref]

197. G. Tokic, D. K. P. Yue. 2012. Optimal shape and motion of undulatory swimming organisms. *Proceedings of the Royal Society B: Biological Sciences* **279**:1740, 3065-3074. [Crossref]

198. Hans-Georg Beyer, Steffen Finck. 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. [Crossref]

199. Martin Pilat, Roman Neruda. An Evolutionary Strategy for Surrogate-Based Multiobjective Optimization 1-7. [Crossref]

200. Aimin Zhou, Qingfu Zhang, Guixu Zhang. A multiobjective evolutionary algorithm based on decomposition and probability model 1-8. [Crossref]

201. Thiago Santos, Ricardo H. C. Takahashi, Gladston J. P. Moreira. A CMA stochastic differential equation approach for many-objective optimization 1-6. [Crossref]

202. Dimitrios V. Vavoulis, Volko A. Straub, John A. D. Aston, Jianfeng Feng. 2012. A Self-Organizing State-Space-Model Approach for Parameter Estimation in Hodgkin-Huxley-Type Models of Single Neurons. *PLoS Computational Biology* **8**:3, e1002401. [Crossref]

203. Anne Auger, Johannes Bader, Dimo Brockhoff, Eckart Zitzler. 2012. Hypervolume-based multiobjective optimization: Theoretical foundations and practical implications. *Theoretical Computer Science* **425**, 75-103. [Crossref]

204. Ling Wang, Xiang Zhong, Min Liu. 2012. A novel group search optimizer for multi-objective optimization. *Expert Systems with Applications* **39**:3, 2939-2946. [Crossref]

205. Oleg Opanasyuk, Lennart B.-Å. Johansson. 2012. On the analyses of fluorescence depolarisation data in the presence of electronic energy migration. Part I: Theory and general description. *Phys. Chem. Chem. Phys.* **14**:6, 1907-1916. [Crossref]

206. Jianyong Sun, J. M. Garibaldi, C. Hodgman. 2012. Parameter Estimation Using Metaheuristics in Systems Biology: A Comprehensive Review. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **9**:1, 185-202. [Crossref]

207. Oleg Opanasyuk, Therese Mikaelsson, Linus Ryderfors, Emad Mukhtar, Lennart B.-Å. Johansson. 2012. On the analyses of fluorescence depolarisation data in the presence of electronic energy migration. Part II: Applying and evaluating two-photon excited fluorescence. *Physical Chemistry Chemical Physics* **14**:6, 1917. [Crossref]

208. Yourim Yoon, Yong-Hyuk Kim, Alberto Moraglio, Byung-Ro Moon. 2012. A theoretical and empirical study on unbiased boundary-extended crossover for real-valued representation. *Information Sciences* **183**:1, 48-65. [Crossref]

209. B Naujoks, H Trautmann, S Wessing, C Weihs. 2011. Advanced concepts for multi-objective evolutionary optimization in aircraft industry. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **225**:10, 1081-1096. [Crossref]

210. Maciej Stachura, Eric W. Frew. 2011. Cooperative Target Localization with a Communication-Aware Unmanned Aircraft System. *Journal of Guidance, Control, and Dynamics* **34**:5, 1352-1362. [Crossref]

211. Nikos D. Lagaros, Matthew G. Karlaftis. 2011. A critical assessment of metaheuristics for scheduling emergency infrastructure inspections. *Swarm and Evolutionary Computation* **1**:3, 147-163. [Crossref]

212. Hoël Langouët, Ludovic Métivier, Delphine Sinoquet, Quang-Huy Tran. 2011. Engine calibration: multi-objective constrained optimization of engine maps. *Optimization and Engineering* **12**:3, 407-424. [Crossref]

213. Leif E. Peterson. 2011. Covariance matrix self-adaptation evolution strategies and other metaheuristic techniques for neural adaptive learning. *Soft Computing* **15**:8, 1483-1495. [Crossref]

214. François O. Laforge, Jonathan Roslund, Ofer M. Shir, Herschel Rabitz. 2011. Multiobjective adaptive feedback control of two-photon absorption coupled with propagation through a dispersive medium. *Physical Review A* **84**:1. . [Crossref]

215. Michael T. M. Emmerich, Andre H. Deutz, Jan Willem Klinkenberg. Hypervolume-based expected improvement: Monotonicity properties and exact computation 2147-2154. [Crossref]

216. Mouadh Yagoubi, Ludovic Thobois, Marc Schoenauer. Asynchronous Evolutionary Multi-Objective Algorithms with heterogeneous evaluation costs 21-28. [Crossref]

217. Gunter Rudolph. On geometrically fast convergence to optimal dominated hypervolume of set-based multiobjective evolutionary algorithms 1719-1723. [Crossref]

218. Renan Cabrera, Ofer M Shir, Rebing Wu, Herschel Rabitz. 2011. Fidelity between unitary operators and the generation of robust gates against off-resonance perturbations. *Journal of Physics A: Mathematical and Theoretical* **44**:9, 095302. [Crossref]

219. Johannes Bader, Eckart Zitzler. 2011. HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization. *Evolutionary Computation* **19**:1, 45-76. [Abstract] [PDF] [PDF Plus]

220. Luis Martí, Jesús García, Antonio Berlanga, Carlos A. Coello Coello, José M. Molina. 2011. MB-GNG: Addressing drawbacks in multi-objective optimization estimation of distribution algorithms. *Operations Research Letters* **39**:2, 150-154. [Crossref]

221. Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, Qingfu Zhang. 2011. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation* **1**:1, 32-49. [Crossref]

222. Tobias Friedrich, Trent Kroeger, Frank Neumann. Weighted Preferences in Evolutionary Multi-objective Optimization 291-300. [Crossref]

223. Ilya Loshchilov, Marc Schoenauer, Michèle Sebag. Not All Parents Are Equal for MO-CMA-ES 31-45. [Crossref]

224. Michael T. M. Emmerich, Carlos M. Fonseca. Computing Hypervolume Contributions in Low Dimensions: Asymptotically Optimal Algorithm and Complexity Results 121-135. [Crossref]

225. Jan Salmen, Lukas Caup, Christian Igel. Real-Time Estimation of Optical Flow Based on Optimized Haar Wavelet Features 448-461. [Crossref]

226. Rui Dilão, Daniele Muraro. 2010. Calibration and validation of a genetic regulatory network model describing the production of the protein Hunchback in Drosophila early development. *Comptes Rendus Biologies* **333**:11-12, 779-788. [Crossref]

227. Tim Hohm, Eckart Zitzler. 2010. A hierarchical approach to model parameter optimization for developmental systems. *Biosystems* **102**:2-3, 157-167. [Crossref]

228. B. Roeschies, C. Igel. 2010. Structure optimization of reservoir networks. *Logic Journal of IGPL* **18**:5, 635-669. [Crossref]

229. Guillaume Collange, Stéphane Reynaud, Nikolaus Hansen. Covariance Matrix Adaptation Evolution Strategy for Multidisciplinary Optimization of Expendable Launcher Family . [Crossref]

230. Karl Bringmann, Tobias Friedrich. 2010. An Efficient Algorithm for Computing Hypervolume Contributions. *Evolutionary Computation* **18**:3, 383-402. [Abstract] [PDF] [PDF Plus]

231. Jorge Munoz, German Gutierrez, Araceli Sanchis. Multi-objective evolution for Car Setup Optimization 1-5. [Crossref]

232. Michael de Paly, Niels Schutze, Andreas Zell. Determining crop-production functions using multi-objective evolutionary algorithms 1-8. [Crossref]

233. Guillermo S. Donatti, Oliver Lomp, Rolf P. Wurtz. Evolutionary optimization of growing neural gas parameters for object categorization and recognition 1-8. [Crossref]

234. Lucas S. Batista, Felipe Campelo, Frederico G. Guimaraes, Jaime A. Ramirez. A new self-adaptive approach for evolutionary multiobjective optimization 1-8. [Crossref]

235. Jean-Baptiste Mouret, Stephane Doncieux. Sferes<inf>v2</inf>: Evolvin' in the multi-core world 1-8. [Crossref]

236. B. Soylu, M. Koksalan. 2010. A Favorable Weight-Based Evolutionary Algorithm for Multiple Criteria Problems. *IEEE Transactions on Evolutionary Computation* **14**:2, 191-205. [Crossref]

237. Ofer M. Shir, Michael Emmerich, Thomas Bäck. 2010. Adaptive Niche Radii and Niche Shapes Approaches for Niching with the CMA-ES. *Evolutionary Computation* **18**:1, 97-126. [Abstract] [PDF] [PDF Plus]

238. Maoguo Gong, Chao Liu, Licheng Jiao, Gang Cheng. 2010. Hybrid immune algorithm with Lamarckian local search for multi-objective optimization. *Memetic Computing* **2**:1, 47-67. [Crossref]

239. A. Lara, G. Sanchez, C.A. Coello Coello, O. Schutze. 2010. HCS: A New Local Search Strategy for Memetic Multiobjective Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation* **14**:1, 112-132. [Crossref]

240. E. Zitzler, L. Thiele, J. Bader. 2010. On Set-Based Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* **14**:1, 58-79. [Crossref]

241. Deyan Luan, Fania Szlam, Kenichi A. Tanaka, Philip S. Barie, Jeffrey D. Varner. 2010. Ensembles of uncertain mathematical models can identify network response to therapeutic interventions. *Molecular BioSystems* **6**:11, 2272. [Crossref]

242. Ilya Loshchilov, Marc Schoenauer, Michèle Sebag. Dominance-Based Pareto-Surrogate for Multi-Objective Optimization 230-239. [Crossref]

243. Dimo Brockhoff. Optimal μ-Distributions for the Hypervolume Indicator for Problems with Linear Bi-objective Fronts: Exact and Exhaustive Results 24-34. [Crossref]

244. Nicola Beume. 2009. S-Metric Calculation by Considering Dominated Hypervolume as Klee's Measure Problem. *Evolutionary Computation* **17**:4, 477-492. [Abstract] [PDF] [PDF Plus]

245. Ke Li, Jinhua Zheng, Miqing Li, Cong Zhou, Hui Lv. A novel algorithm for non-dominated hypervolume-based multiobjective optimization 5220-5226. [Crossref]

246. Aimin Zhou, Qingfu Zhang, Yaochu Jin. 2009. Approximating the Set of Pareto-Optimal Solutions in Both the Decision and Objective Spaces by an Estimation of Distribution Algorithm. *IEEE Transactions on Evolutionary Computation* **13**:5, 1167-1189. [Crossref]

247. José Antonio Martín H., Javier de Lope, Matilde Santos. 2009. A method to learn the inverse kinematics of multi-link robots by evolving neuro-controllers. *Neurocomputing* **72**:13-15, 2806-2814. [Crossref]

248. Thorsten Suttorp, Nikolaus Hansen, Christian Igel. 2009. Efficient covariance matrix update for variable metric evolution strategies. *Machine Learning* **75**:2, 167-197. [Crossref]

249. Mao-Guo GONG, Li-Cheng JIAO, Dong-Dong YANG, Wen-Ping MA. 2009. Research on Evolutionary Multi-Objective Optimization Algorithms. *Journal of Software* **20**:2, 271-289. [Crossref]

250. Carlos A. Coello Coello. 2009. Evolutionary multi-objective optimization: some current research trends and topics that remain to be explored. *Frontiers of Computer Science in China* **3**:1, 18-30. [Crossref]

251. Ofer M. Shir, Thomas Bäck. 2009. Niching with derandomized evolution strategies in artificial and real-world landscapes. *Natural Computing* **8**:1, 171-196. [Crossref]

252. Chi-Keong Goh, Kay Chen Tan. 2009. A Competitive-Cooperative Coevolutionary Paradigm for Dynamic Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* **13**:1, 103-127. [Crossref]

253. Ofer M. Shir, Mike Preuss, Boris Naujoks, Michael Emmerich. Enhancing Decision Space Diversity in Evolutionary Multiobjective Algorithms 95-109. [Crossref]

254. M. Shirzaei, T. R. Walter. 2009. Randomly iterated search and statistical competency as powerful inversion tools for deformation source modeling: Application to volcano interferometric synthetic aperture radar data. *Journal of Geophysical Research* **114**:B10. . [Crossref]

255. Johannes Bader, Dimo Brockhoff, Samuel Welten, Eckart Zitzler. On Using Populations of Sets in Multiobjective Optimization 140-154. [Crossref]

256. Thomas Voß, Nikolaus Hansen, Christian Igel. Recombination for Learning Strategy Parameters in the MO-CMA-ES 155-168. [Crossref]

257. Rui Dilão, Daniele Muraro, Miguel Nicolau, Marc Schoenauer. Validation of a Morphogenesis Model of Drosophila Early Development by a Multi-objective Evolutionary Optimization Algorithm 176-190. [Crossref]

258. Cesar Gomes Miguel, Carolina Feher da Silva, Marcio Lobo Netto. Structural and Parametric Evolution of Continuous-Time Recurrent Neural Networks 177-182. [Crossref]

259. Maoguo Gong, Licheng Jiao, Haifeng Du, Liefeng Bo. 2008. Multiobjective Immune Algorithm with Nondominated Neighbor-Based Selection. *Evolutionary Computation* **16**:2, 225-255. [Abstract] [PDF] [PDF Plus]

260. Thomas Voss, Nicola Beume, Gunter Rudolph, Christian Igel. Scalarization versus indicator-based selection in multi-objective CMA evolution strategies 3036-3043. [Crossref]

261. Kalyanmoy Deb. Introduction to Evolutionary Multiobjective Optimization 59-96. [Crossref]

262. M. Jebalia, A. Auger, M. Schoenauer, F. James, M. Postel. Identification of the isotherm function in chromatography using CMA-ES 4289-4296. [Crossref]

263. T. Ganesan, I. Elamvazuthi, K. Z. K. Shaari, P. Vasant. Multiobjective Optimization of Bioethanol Production via Hydrolysis Using Hopfield-Enhanced Differential Evolution 340-359. [Crossref]