Non-elitist Evolutionary Multi-objective Optimizers Revisited

Ryoji Tanabe
Shenzhen Key Laboratory of Computational Intelligence,
University Key Laboratory of Evolving Intelligent Systems
of Guangdong Province,
Department of Computer Science and Engineering,
Southern University of Science and Technology
rt.ryoji.tanabe@gmail.com

Hisao Ishibuchi*
Shenzhen Key Laboratory of Computational Intelligence,
University Key Laboratory of Evolving Intelligent Systems
of Guangdong Province,
Department of Computer Science and Engineering,
Southern University of Science and Technology
hisao@sustech.edu.cn

ABSTRACT

Since around 2000, it has been considered that elitist evolutionary multi-objective optimization algorithms (EMOAs) always outperform non-elitist EMOAs. This paper revisits the performance of non-elitist EMOAs for bi-objective continuous optimization when using an unbounded external archive. This paper examines the performance of EMOAs with two elitist and one non-elitist environmental selections. The performance of EMOAs is evaluated on the bi-objective BBOB problem suite provided by the COCO platform. In contrast to conventional wisdom, results show that non-elitist EMOAs with particular crossover methods perform significantly well on the bi-objective BBOB problems with many decision variables when using the unbounded external archive. This paper also analyzes the properties of the non-elitist selection.

CCS CONCEPTS

• **Mathematics of computing → Evolutionary algorithms**;

KEYWORDS

Evolutionary multi-objective optimization, continuous optimization, non-elitist environmental selections

ACM Reference Format:
Ryoji Tanabe and Hisao Ishibuchi. 2019. Non-elitist Evolutionary Multi-objective Optimizers Revisited. In *Genetic and Evolutionary Computation Conference (GECCO '19), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3321707.3321754

1 INTRODUCTION

Since no solution can simultaneously minimize multiple conflicting objective functions in general, the ultimate goal of multi-objective optimization problems (MOPs) is to find a Pareto optimal solution preferred by a decision maker [27]. When the decision maker's preference information is unavailable a priori, an "a posteriori" decision making is performed. The decision maker selects the final solution from a solution set that approximates the Pareto front in the objective space.

An evolutionary multi-objective optimization algorithm (EMOA) is frequently used to find an approximation of the Pareto front for the "a posteriori" decision making [8]. A number of EMOAs have been proposed in the literature. Classical EMOAs include VEGA [29], MOGA [16], and NSGA [31] proposed in the 1980s and 1990s. They are non-elitist EMOAs, which do not have a mechanism to maintain non-dominated solutions in the population. Some elitist EMOAs (e.g., SPEA [38], SPEA2 [37], and NSGA-II [10]) have been proposed in the early 2000s. Elitist EMOAs explicitly keep non-dominated solutions found during the search process.

Some EMOAs store non-dominated solutions found so far in an unbounded or bounded external archive independently from the population. For example, MOGLS [21] proposed in the mid-1990s does not maintain elite solutions in the population but stores all non-dominated solutions found so far in the unbounded external archive. ϵ-MOEA [13] stores non-dominated solutions in the population and ϵ-nondominated solutions in the unbounded external archive. PESA [7] uses the non-elitist population and the elitist bounded external archive. The external archive in these EMOAs (e.g., MOGLS, ϵ-MOEA, and PESA) plays two roles. The first role is to provide non-dominated solutions found so far to the decision maker. The performance of these types of EMOAs is also evaluated based on solutions in the external archive, rather than the population. The second role is to perform an elitist search. For example, parents for mating are selected from the external archive in PESA. Some elitist individuals in the external archive can enter the population in MOGLS. Since these types of EMOAs explicitly exploit elitist solutions as explained above, they can be categorized into elitist EMOAs.

Apart from algorithm development, the external archive has been used only for the first role (e.g., [4, 5, 16, 25, 35]). As pointed out in [4], good potential solutions found so far are likely to be discarded from the population. The external archive that stores all non-dominated solutions independently from EMOAs can address this issue. The external archive for the first role can be incorporated into all EMOAs without any changes in their algorithmic behavior. The external archive is useful for real-world problems where the evaluation of each solution is expensive, i.e., the total number of examined solutions is limited, and the archive maintenance cost is relatively small in comparison with the solution evaluation cost. If the decision maker wants to examine a small number of non-dominated solutions, solution selection methods are available such as hypervolume indicator-based selection methods (e.g., [4]) and distance-based selection methods (e.g., [30]).

[*] Corresponding author: Hisao Ishibuchi

GECCO '19, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6111-8/19/07…$15.00
https://doi.org/10.1145/3321707.3321754

This paper revisits non-elitist EMOAs with the unbounded external archive only for the first role (performance evaluation). When the performance of EMOAs is evaluated based on solutions in the external archive as in [4, 5, 25, 35], the role of EMOAs is only to find non-dominated solutions with high quality. Thus, EMOAs do not need to maintain non-dominated solutions found so far in the current population with the population size μ. We investigate three environmental selections: best-all (BA), best-family (BF), and best-children (BC). While BA and BF are elitist selections, BC is a non-elitist selection. Although BA is a traditional $(\mu + \lambda)$-selection, BF and BC restrict a selection only among k parents and λ children. Thus, $\mu - k$ non-parents do not directly participate in the selection process in BF and BC unlike traditional $(\mu+\lambda)$- and (μ, λ)-selections. In BC, all k parents are removed from the population regardless of their quality. Then, the top-ranked k out of λ children enter the population. Subsection 2.3 explains BA, BF, and BC in detail. We examine the performance of EMOAs with the three selections on the bi-objective BBOB problem suite [33]. We use five crossover methods and four ranking methods in representative EMOAs.

Our contributions in this paper are at least threefold:

- We demonstrate that the non-elitist BC selection performs significantly well on the bi-objective BBOB problems with many decision variables when using the unbounded external archive. Although most EMOAs proposed in the 2000s are elitist EMOAs, our results indicate that efficient non-elitist EMOAs could be designed. Thus, our results significantly expand the design possibility of EMOAs.
- We demonstrate that restricted replacements in BF and BC are suitable for crossover methods with the preservation of statistics [23] (e.g., the property where the covariance matrix of children is the same as that of the parents) such as SPX [32] and REX [2].
- We discuss why the simple BA selection performs worse than the restricted BF and BC selections. We also analyze the properties of the non-elitist BC selection.

The rest of this paper is organized as follows. Section 2 provides some preliminaries of this paper, including the definition of MOPs, the five crossover methods, and the three environmental selections. Section 3 describes the experimental setup. Section 4 examines the performance of the three environmental selections. Section 5 concludes this paper with discussions on future research directions.

2 PRELIMINARIES

2.1 Definition of continuous MOPs

A continuous MOP is to find a solution $\boldsymbol{x} \in \mathbb{S}$ that minimizes a given objective function vector $\boldsymbol{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m, \boldsymbol{x} \mapsto \boldsymbol{f}(\boldsymbol{x})$. Here, $\mathbb{S} \subseteq \mathbb{R}^n$ is the n-dimensional solution space, and $\mathbb{R}^m$ is the m-dimensional objective space. n is the number of decision variables, and m is the number of objective functions.

A solution $\boldsymbol{x}^{(1)}$ is said to dominate $\boldsymbol{x}^{(2)}$ iff $f_i(\boldsymbol{x}^{(1)}) \leq f_i(\boldsymbol{x}^{(2)})$ for all $i \in \{1, ..., m\}$ and $f_i(\boldsymbol{x}^{(1)}) < f_i(\boldsymbol{x}^{(2)})$ for at least one index i. If $\boldsymbol{x}^*$ is not dominated by any other solutions in $\mathbb{S}$, $\boldsymbol{x}^*$ is a Pareto optimal solution. The set of all $\boldsymbol{x}^*$ is the Pareto optimal solution set, and the set of all $\boldsymbol{f}(\boldsymbol{x}^*)$ is the Pareto front. The goal of MOPs for the "a posteriori" decision making is to find a non-dominated solution set that approximates the Pareto front in the objective space.

Table 1: Properties of the five crossover methods, including the center of the distribution of children (parent or mean), the type of probability distribution (U: uniform or N: normal), the rotational invariance, the preservation of statistics, the number of parents k, and other control parameters.

	Cent.	Prob.	Rot.	Sta.	k	Parameters
SBX	parent	?			2	$\eta_c = 20, \eta_m = 20$
BLX	mean	U			2	$\alpha = 0.5$
PCX	parent	N	✓		3	$\sigma_\zeta^2 = 0.1, \sigma_\eta^2 = 0.1$
SPX	mean	U	✓	✓	$n + 1$	$\epsilon = \sqrt{n+2}$
REX	mean	N	✓	✓	$n + 1$	$\sigma^2 = 1/(k-1)$

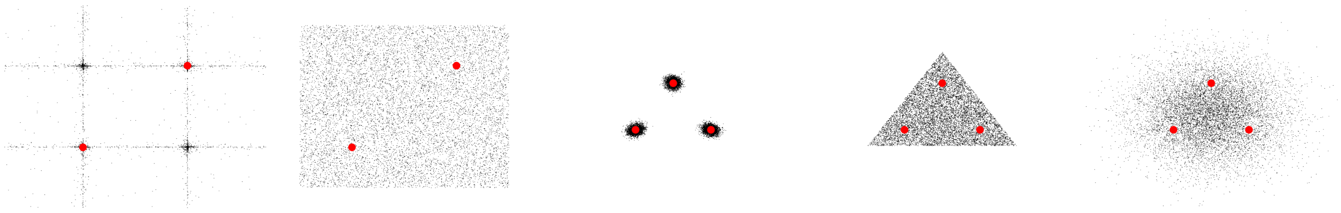
2.2 Crossover methods in real-coded GAs

We use the following five crossover methods in real-coded GAs: simulated binary crossover (SBX) [9], blend crossover (BLX) [15], parent-centric crossover (PCX) [11], simplex crossover (SPX) [32], and real-coded ensemble crossover (REX) [2]. Here, we briefly explain the five crossover methods.

Traditional GAs use two variation operators: crossover and mutation. In contrast, real-coded GAs with BLX, PCX, SPX, and REX do not need the mutation operator because they can generate diverse children by adjusting their control parameters (e.g., the expansion rate ϵ in SPX). However, the polynomial mutation (PM) [9] is applied to two children generated by SBX in most studies. In other words, SBX and PM have been considered to be a set. For this reason, we apply PM to children generated only by SBX. We refer to "SBX and PM" as "SBX" for simplicity.

Table 1 shows the properties of the five crossover methods. Although SBX and BLX are traditional two-parent crossover methods, PCX, SPX, and REX are multi-parent crossover methods. PCX, SPX, and REX are rotationally invariant. The performance of EMOAs with rotationally invariant operators does not depend on the coordinate system. While PCX and REX use a Normal probability distribution, BLX and SPX use a uniform probability distribution. The probability distribution used in SBX is unclear. Although the center of the distribution of children is the mean vector of k parents in BLX, SPX, and REX, that is one of the parents in SBX and PCX. SPX and REX have a property called the "preservation of statistics" proposed in [23]. In a crossover method with this property, children inherit the statistics (e.g., the mean vector and the covariance matrix) from their parents.

Figure 1 shows the distribution of children generated by the five crossover methods. SBX simulates the working principle of the single-point crossover in binary-coded GAs. Since SBX is a variable-wise operator, most children are generated along the coordinate axes. The distribution of children is controlled by η_c in SBX (and η_m in PM). In BLX, the j-th element ($j \in \{1, ..., n\}$) of a child is uniformly randomly selected from the range $[l_j, u_j]$. Here, $l_j = \min(x_j^{(1)}, x_j^{(2)}) - \alpha|x_j^{(1)} - x_j^{(2)}|$ and $u_j = \max(x_j^{(1)}, x_j^{(2)}) + \alpha|x_j^{(1)} - x_j^{(2)}|$. $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ are parents, and α is the expansion factor.



(a) SBX (b) BLX (c) PCX (d) SPX (e) REX

Figure 1: Distribution of children generated by the five crossover methods. Large red points are their parents.

PCX is a parent-centric version of UNDX-m [24], which is a multi-parent extension of unimodal normal distribution crossover (UNDX) [28]. While the center of the distribution of children is the mean vector of parents in UNDX-m, that is one of the parents in PCX. PCX requires two parameters σ_{ζ}^2 and σ_{η}^2 that control the variances of two Normal distributions. SPX can be viewed as being a rotationally invariant version of BLX. SPX uniformly generates children inside an expanded simplex formed by k parents. The theoretical analysis presented in [19] shows that SPX with the expansion factor $\epsilon = \sqrt{n+2}$ satisfies the preservation of statistics. REX is a generalized version of UNDX-m. REX using a zero-mean Normal distribution with the variance $\sigma^2 = 1/(k-1)$ satisfies the preservation of statistics [1].

2.3 Environmental selections

We consider a "simple" EMOA shown in Algorithm 1. After the initialization of the population $\boldsymbol{P}$ with the population size μ (line 1), the following operations are repeatedly performed until a termination condition is satisfied. First, k parents are randomly selected from $\boldsymbol{P}$ such that their indices are different from each other (line 3). Let $\boldsymbol{R}$ be a set of the k parents. Then, λ children are generated by applying a crossover method to the same k parents λ times (line 4).[1] To effectively exploit the neighborhood of the k parents, the same parents are generally used to generate children in GAs for single-objective optimization [1]. Let $\boldsymbol{Q}$ be a set of the λ children. At the end of each iteration, the environmental selection is performed using $\boldsymbol{P}$, $\boldsymbol{R}$, and $\boldsymbol{Q}$ (line 5).

Below, we explain the following three environmental selections: best-all (BA), best-family (BF), and best-children (BC). Note that our main contributions in this paper are analysis of BA, BF, and BC in Section 4, not proposing BA, BF, and BC. Algorithms 2, 3, and 4 show BA, BF, and BC, respectively. While BA and BF are elitist selections, BC is a non-elitist selection. The three selections require a method of ranking individuals based on their quality. Similar to MO-CMA-ES [20], BA, BF, and BC can be combined with any ranking method. In this paper, we use four ranking methods in NSGA-II [10], SMS-EMOA [3], SPEA2 [37], and IBEA with the additive ϵ indicator [36]. We denote their ranking methods as "NS", "SM", "SP", and "IB", respectively. Individuals are ranked based on their non-domination levels in NS and SM. The tie-breakers are the crowding distance in NS and the hypervolume contribution in SM. In SP and IB, individuals are sorted based on their so-called fitness values in descending order. In this paper, X-Y-Z represents the EMOA (Algorithm 1) with an environmental selection X, a crossover method Y, and a ranking method Z. For example, BA-SBX-NS is the EMOA with BA, SBX, and NS.

Algorithm 1: The simple EMOA

```
1 $t \leftarrow 1$, initialize the population $\boldsymbol{P} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(\mu)}\}$;
2 while The termination criteria are not met do
3     $\boldsymbol{R} \leftarrow$ Randomly select $k$ parents from $\boldsymbol{P}$;
4     $\boldsymbol{Q} \leftarrow$ Generate $\lambda$ children by applying the crossover method to $\boldsymbol{R}$;
5     $\boldsymbol{P} \leftarrow$ environmentalSelection($\boldsymbol{P}$, $\boldsymbol{Q}$, $\boldsymbol{R}$);
6     $t \leftarrow t + 1$;
```

Algorithm 2: BA (the elitist selection)

```
Assign ranks to all individuals in $\boldsymbol{P} \cup \boldsymbol{Q}$;
$\boldsymbol{S} \leftarrow \boldsymbol{P} \cup \boldsymbol{Q}$ and $\boldsymbol{P} \leftarrow \emptyset$;
for $i \in \{1, \ldots, \mu\}$ do
    $\boldsymbol{x} \leftarrow$ Select the best ranked individual from $\boldsymbol{S}$;
    $\boldsymbol{P} \leftarrow \boldsymbol{P} \cup \{\boldsymbol{x}\}$ and $\boldsymbol{S} \leftarrow \boldsymbol{S} \setminus \{\boldsymbol{x}\}$;
```

Algorithm 3: BF (the elitist restricted selection)

```
Assign ranks to all individuals in $\boldsymbol{P} \cup \boldsymbol{Q}$;
$\boldsymbol{S} \leftarrow \boldsymbol{Q} \cup \boldsymbol{R}$ and $\boldsymbol{P} \leftarrow \boldsymbol{P} \setminus \boldsymbol{R}$;
for $i \in \{1, \ldots, k\}$ do
    $\boldsymbol{x} \leftarrow$ Select the best ranked individual from $\boldsymbol{S}$;
    $\boldsymbol{P} \leftarrow \boldsymbol{P} \cup \{\boldsymbol{x}\}$ and $\boldsymbol{S} \leftarrow \boldsymbol{S} \setminus \{\boldsymbol{x}\}$;
```

Algorithm 4: BC (the non-elitist restricted selection)

```
$\boldsymbol{P} \leftarrow \boldsymbol{P} \setminus \boldsymbol{R}$;
Assign ranks to all individuals in $\boldsymbol{P} \cup \boldsymbol{Q}$;
for $i \in \{1, \ldots, k\}$ do
    $\boldsymbol{x} \leftarrow$ Select the best ranked individual from $\boldsymbol{Q}$;
    $\boldsymbol{P} \leftarrow \boldsymbol{P} \cup \{\boldsymbol{x}\}$ and $\boldsymbol{Q} \leftarrow \boldsymbol{Q} \setminus \{\boldsymbol{x}\}$;
```

In BA (Algorithm 2), the top-ranked μ individuals are selected from the union of $\boldsymbol{P}$ and $\boldsymbol{Q}$. BA is the traditional elitist $(\mu + \lambda)$-selection used in most EMOAs (e.g., NSGA-II and SPEA2). It should

[1] Since SBX generates two children in a single operation, SBX is performed $\lambda/2$ times.

be noted that BA-SBX-NS is not identical to NSGA-II. The differences between BA-SBX-NS and NSGA-II are the random parent selection and the children generation. The same k parents are used to generate λ children in BA. For the same reason, BA-SBX-SP, BA-SBX-SM, and BA-SBX-IB are not identical to SPEA2, SMS-EMOA, and IBEA, respectively.

In BF (Algorithm 3), the environmental selection is performed only among the so-called "family" that consists of λ children in $\boldsymbol{Q}$ and k parents in $\boldsymbol{R}$. After all individuals in the union of $\boldsymbol{P}$ and $\boldsymbol{Q}$ have been ranked, only k parents in $\boldsymbol{R}$ are removed from $\boldsymbol{P}$. Then, the best k individuals are selected from the union of $\boldsymbol{Q}$ and $\boldsymbol{R}$. Although non-parents in $\boldsymbol{P}$ do not directly participate in the selection process, they contribute to assign ranks to individuals in the union of $\boldsymbol{Q}$ and $\boldsymbol{R}$. While the maximum number of individuals replaced by children is μ in BA, that is k in BF. Since only k parents can be replaced by children in BF, non-parents can survive to the next iteration with no comparison. Selections among families as in BF are used in GAs for single-objective optimization (e.g., the deterministic crowding [26]).

In BC (Algorithm 4), the environmental selection is performed among λ children in $\boldsymbol{Q}$. We assume that $\lambda \geq k$. After k parents in $\boldsymbol{R}$ have been removed from $\boldsymbol{P}$, all individuals in the union of $\boldsymbol{P}$ and $\boldsymbol{Q}$ are ranked. Then, the best k individuals are selected from $\boldsymbol{Q}$. Since all k parents are deleted from $\boldsymbol{P}$ regardless of their quality, BC does not maintain non-dominated individuals in $\boldsymbol{P}$. Thus, BC is a non-elitist selection in contrast to the elitist BA and BF selections. While μ individuals in $\boldsymbol{P}$ are replaced with λ children in $\boldsymbol{Q}$ in most classical (μ, λ)-EMOAs (e.g., MOGA), only k parents in $\boldsymbol{R}$ are replaced with the best k out of λ children in $\boldsymbol{Q}$ in BC. Thus, BC is different from the traditional (μ, λ)-selection.

BC can be viewed as being an extension of just generation gap (JGG) [1] to multi-objective optimization. JGG is an environmental selection in GAs for single-objective continuous optimization. The only difference between BC and JGG is how to assign ranks to individuals. Individuals $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots$ are ranked based on their objective values $f(\boldsymbol{x}^{(1)}), f(\boldsymbol{x}^{(2)}), \ldots$ in JGG and their objective vectors $\boldsymbol{f}(\boldsymbol{x}^{(1)}), \boldsymbol{f}(\boldsymbol{x}^{(2)}), \ldots$ in BC. The results presented in [1] show that non-elitist GAs with JGG significantly outperform elitist GAs on single-objective test problems (especially multimodal problems) when using crossover methods with the preservation of statistics.

3 EXPERIMENTAL SETTINGS

We conducted all experiments using the comparing continuous optimizers (COCO) platform [17]. COCO is the standard platform used in the black box optimization benchmarking (BBOB) workshops held at GECCO (2009–present). We used the latest COCO software (version 2.2.2) downloaded from https://github.com/numbbo/coco. COCO provides six types of BBOB problem suites, including the single-objective BBOB noiseless problem suite [18]. The bi-objective BBOB problem suite [33] consists of 55 bi-objective test problems $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{55}$ designed based on the idea presented in [5]. Each bi-objective BBOB problem is constructed by combining two single-objective BBOB problems. For example, the first and second objective functions of $\boldsymbol{f}_7$ are the Sphere function and the rotated Rastrigin function, respectively. The number of decision variables n is $n \in \{2, 3, 5, 10, 20, 40\}$. For details of the 55 bi-objective test problems, see [33]. For each problem, 15 runs were performed. These settings adhere to the analysis procedure adopted by the GECCO BBOB community. The maximum number of function evaluations was set to $10^4 \times n$.

COCO also provides the post-processing tool that aggregates experimental data. COCO automatically stores all non-dominated solutions found by an optimizer in the unbounded external archive. The performance indicator I_{COCO} [6] in COCO is mainly based on the hypervolume value of non-dominated solutions in the unbounded external archive. When no solution in the external archive dominates a predefined reference point in the normalized objective space, the I_{COCO} value is calculated based on the distance to the so-called region of interest. For details of I_{COCO}, see [6].

We implemented all algorithms using jMetal [14]. Source codes of all algorithms are available at https://sites.google.com/view/nemorgecco2019/. For all five crossover methods (except for PCX), we used the control parameters recommended in the literature shown in Table 1. Since PCX with $k = 3$ performed poorly in our preliminary study, we set k to $n + 1$ similar to SPX and REX. For comparison, we evaluated the performance of the original NSGA-II, SPEA2, SMS-EMOA, and IBEA. SBX and PM with $p_c = 0.9$, $\eta_c = 20$, $p_m = 1/n$, and $\eta_m = 20$ were used in the original EMOAs. As in [34], μ was set to $\lfloor 100 \ln(n) \rfloor$. The number of children λ was set to $10n$. We set the λ value based on our preliminary results and studies of GAs for single-objective optimization (e.g., [1, 2]).

4 RESULTS

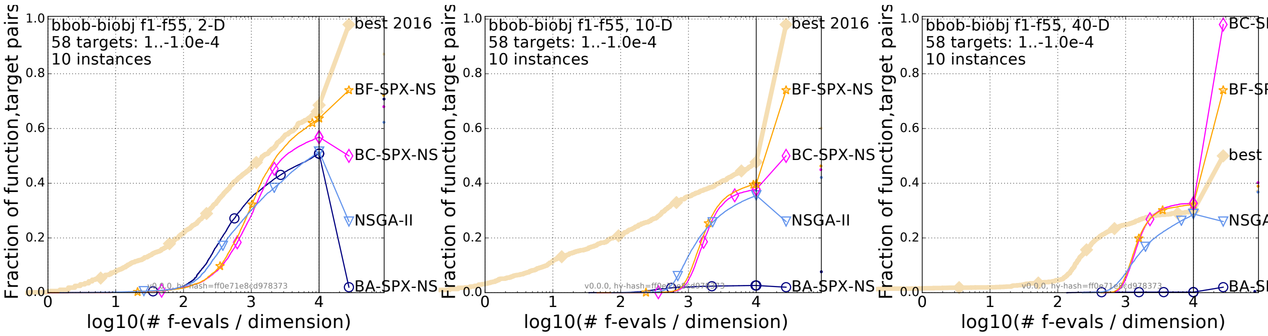
This section shows analysis of the three environmental selections (BA, BF, and BC). Since SPX is suitable for BF and BC, we mainly discuss results of EMOAs with SPX. Although results of EMOAs with REX are similar to those with SPX, we do not show them here due to space constraint. As shown in Subsection 4.4, SBX, BLX, and PCX are not suitable for BA, BF, and BC.

Subsection 4.1 shows a comparison among BA-SPX-NS, BF-SPX-NS, BC-SPX-NS, and the original NSGA-II. Subsection 4.2 investigates why BA performs poorly. Subsection 4.3 analyzes the advantages and disadvantages of the non-elitist BC compared with the elitist BF. Subsection 4.4 examines the performance of BA, BF, and BC with other crossover methods (SBX, BLX, PCX, and REX). Subsection 4.5 presents a comparison of BA, BF, and BC with other ranking methods (SP, SM, and IB).

4.1 Comparison of BA, BF, and BC

Figure 2 shows results of the original NSGA-II, BA-SPX-NS, BF-SPX-NS, and BC-SPX-NS on all 55 BBOB problems with $n \in \{2, 10, 40\}$. Due to space constraint, results for $n \in \{3, 5, 20\}$ are not shown, but they are similar to results for $n \in \{2, 10\}$. In this section, we use the SPX crossover and the NS ranking method. In Figure 2, "best 2016" is a *virtual algorithm portfolio* that is constructed from the performance data of 15 algorithms participating in the GECCO BBOB 2016 workshop. Note that "best 2016" does not mean the best optimizer among the 15 algorithms.

Figure 2 shows the bootstrapped empirical cumulative distribution (ECDF) of the number of function evaluations (FEvals) divided by n (FEvals/n) for 58 target I_{COCO} indicator values $\{-10^{-4},$



(a) $n = 2$ **(b)** $n = 10$ **(c)** $n = 40$

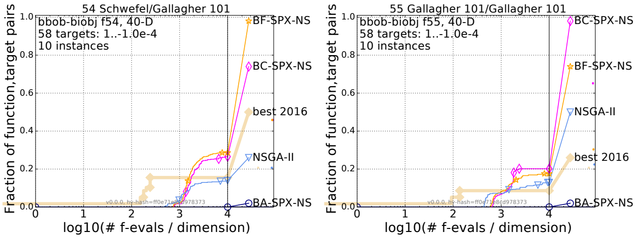
Figure 2: Results of the original NSGA-II, BA-SPX-NS, BF-SPX-NS, and BC-SPX-NS on all 55 bi-objective BBOB test problems with $n \in \{2, 10, 40\}$ (higher is better). For the notation X-Y-Z, see Subsection 2.3.

$-10^{-4.2}, \ldots, 10^{-0.1}, 10^{0}\}$ for all 55 BBOB problems with each n. We used the COCO software to generate all ECDF figures in this paper. In Figure 2, the vertical axis indicates the proportion of target I_{COCO} indicator values which a given optimizer can reach within specified function evaluations. For example, in Figure 2 (b), BF-SPX-NS reaches about 40 percent of all 58 target I_{COCO} indicator values within $10^4 \times n$ evaluations on all 55 problems with $n = 10$ in all runs. If an optimizer finds all Pareto optimal solutions on all 55 problems in all runs, the vertical value becomes 1. More detailed explanations of the ECDF (including illustrative examples) are found in [5, 6].

Statistical significance is also tested with the rank-sum test ($p = 0.05$) for a given target value using the COCO software. However, statistical test results are almost consistent with ECDF figures. Additionally, the space of this paper is limited. For these reasons, we show only ECDF figures. The statistical test results and other ECDF figures are available at https://sites.google.com/view/nemorgecco2019/.

Figure 2 shows that BA-SPX-NS performs the best until $10^3 \times n$ evaluations for $n = 2$. However, the increase of n deteriorates the performance of BA-SPX-NS. The evolution of BA-SPX-NS clearly stagnates for $n \geq 10$. The original NSGA-II is the best performer in the early stage for $n \geq 10$. BF-SPX-NS and BC-SPX-NS perform better than NSGA-II and BA-SPX-NS in the later stage for all n. Interestingly, the non-elitist BC-SPX-NS performs the best in the later stage for $n = 40$. Although it has been believed that elitist EMOAs always outperform non-elitist EMOAs for about two decades, our results show that the non-elitist BC-SPX-NS performs better than the elitist NSGA-II, BA-SPX-NS, and BF-SPX-NS on the bi-objective BBOB problems with $n = 40$ when using the unbounded external archive.

Note that BC-SPX-NS is not always the best optimizer on all 55 BBOB problems with $n = 40$. Figure 3 shows results on f_{54} and f_{55} with $n = 40$. While BF-SPX-NS outperforms BC-SPX-NS on f_{54}, BC-SPX-NS outperforms BF-SPX-NS on f_{55}. Similar to Figure 3, the best optimizer is different depending on the test problem. We attempted to clarify which problem groups BC performs the best (e.g., BC has the best performance on multimodal problems with weak global structure such as f_{54} and f_{55}). Unfortunately, we could not find such a result. An in-depth analysis is needed to understand on which problems BC performs well or poorly.



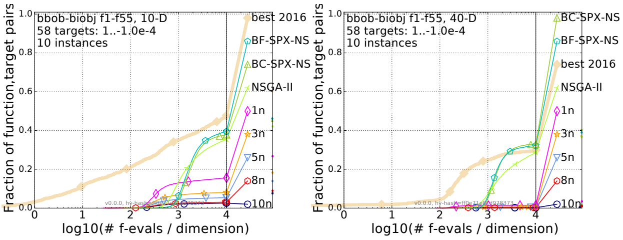
(a) f_{54}**: BF outperforms BC** **(b)** f_{55}**: BC outperforms BF**

Figure 3: Results of NSGA-II, BA-SPX-NS, BF-SPX-NS, and BC-SPX-NS on f_{54} and f_{55} with $n = 40$.

4.2 Why does BA perform poorly?

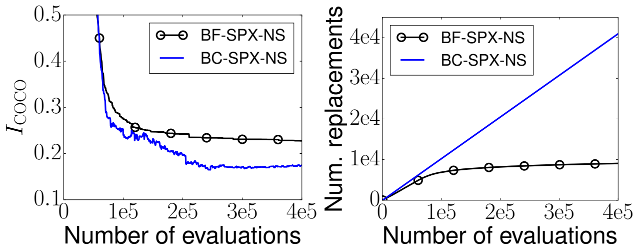
Here, we discuss the poor performance of BA-SPX-NS observed in Subsection 4.1. The biased distribution of children is likely to cause the poor performance of BA-SPX-NS. As shown in Figure 1 (d), SPX generates λ children inside a simplex formed by k parents. If the k parents are close to each other in the solution space, their λ children are likely to be in local area. If non-parents in the population are ranked worse than the children, the non-parents are replaced with the children in BA. This means that non-parents in not-well-explored area cannot survive to the next iteration. Thus, BA-SPX-NS is likely to lose diversity in the solution and objective spaces as the search progresses.

One may think that the above-mentioned issue caused by the biased distribution of children can be addressed by setting λ to a small value. Figure 4 shows BA-SPX-NS with $\lambda \in \{1n, 3n, 5n, 8n, 10n\}$ on all 55 BBOB problems with $n \in \{10, 40\}$. In Figure 4, "$10n$" is identical to BA-SPX-NS in Figure 2. Figure 4 also shows the results of NSGA-II, BF-SPX-NS and BC-SPX-NS derived from Figure 2. Figure 4 shows that the performance of BA-SPX-NS can be improved by setting λ to a small value. However, BA-SPX-NS with any λ is



(a) $n = 10$ **(b)** $n = 40$

Figure 4: Results of BA-SPX-NS with various λ values.



(a) I_{COCO} values **(b) Num. replacements**

Figure 5: (a) Raw I_{COCO} indicator values on f_{46} with $n = 40$ (lower is better). (b) Cumulative number of parents replaced by children. Results of a single run are shown.

outperformed by NSGA-II, BF-SPX-NS, and BC-SPX-NS at the later stage.

In general, a large enough number of children are necessary to find better solutions in the current search area [1]. Thus, BA is in a dilemma. A large λ value is helpful for BA to exploit the current search area, but it causes premature convergence. A small λ value can prevent BA from the premature convergence, but it is not sufficiently large to exploit the current search area. In addition to SPX, we observed the same issue in other crossover methods (except for SBX).

In contrast to BA, only k parents can be replaced with children in BF and BC. This restricted replacement in BF and BC can help the population to maintain the diversity. Even if non-parents in not-well-explored area are dominated by the children, the non-parents can survive to the next iteration with no comparison. Thus, BF and BC can address the BA's dilemma. In fact, BF-SPX-NS and BC-SPX-NS perform significantly better than BA-SPX-NS.

4.3 Advantages and disadvantages of BC

As shown in Subsection 4.1, the non-elitist BC performs better than the elitist BF for $n = 40$. Here, we discuss the advantages and disadvantages of BC compared with BF.

Figure 5 (a) shows raw I_{COCO} indicator values of *the population* in BF-SPX-NS and BC-SPX-NS on f_{46} with $n = 40$, which consists of two rotated Rastrigin function instances. In all 55 BBOB test problems, f_{46} can be viewed as being a representative multimodal problem. We slightly modified the COCO software to calculate the I_{COCO} value of the population (not the external archive). A lower raw I_{COCO} value is better. The range of the I_{COCO} value in Figure 5 (a) is limited to $[0.1, 0.5]$ in order to focus on the interesting behavior of BC-SPX-NS. Although the I_{COCO} value of the elitist BF-SPX-NS almost[2] monotonically decreases as the search progresses, that of the non-elitist BC-SPX-NS is unstable. Since BC does not maintain best-so-far non-dominated solutions in the population, its I_{COCO} value sometimes deteriorates compared with the previous iteration.

Figure 5 (b) shows the cumulative number c of parents replaced by children. In BF-SPX-NS, the evolution of c clearly stagnates after 10^5 function evaluations. This result means that BF-SPX-NS rarely generates better children than parents. In fact, the raw I_{COCO} value of BF-SPX-NS is not significantly improved after 10^5 function evaluations, as shown in Figure 5 (a). Since BC-SPX-NS always replaces k parents with the best k out of λ children for every iteration, c linearly increases. Thus, the replacement of individuals in BC occurs more frequently than that in BF. This property of BC is helpful for exploration of the search space.

The above observations indicate that BC has a similar advantage to simulated annealing [22], which can move to a worse search point. As pointed out by Deb and Goel [12], if an elitist EMOA prematurely converges to local Pareto optimal solutions, it is very likely to stagnate. Unless the elitist EMOA finds better solutions far from the current search area, it cannot escape from local Pareto optimal solutions. In contrast, the non-elitist BC always replaces k parents with children regardless of the quality of k parents. While most elitist environmental selections accept only "downhill" moves on minimization problems, the non-elitist BC can accept "uphill" moves as in simulated annealing. The uphill moves in BC help the population to escape from local Pareto optimal solutions on *some* multimodal problems.

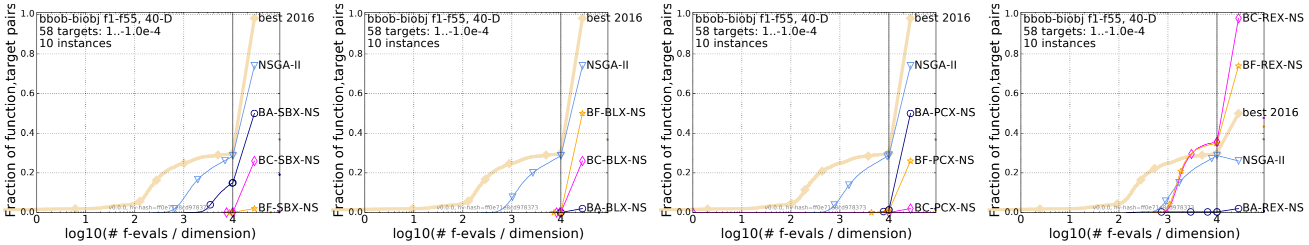
However, BC has at least two disadvantages compared with the elitist BF. First, as discussed in Subsection 4.1, BC performs worse than BF on some problems even with $n = 40$. Second, as reported in Subsection 4.1, BC performs worse than BF at the early stage. Since BC can accept "uphill" moves as in simulated annealing, the exploitative ability of BC is worse than that of BF. A deterministic or adaptive method of switching BC and BF may be promising to exploit their advantages.

4.4 Which crossover methods are suitable for the non-elitist BC?

The results in Subsection 4.1 show that BC-SPX-NS outperforms BA-SPX-NS, BF-SPX-NS, and NSGA-II for $n = 40$. Here, we examine which crossover methods are suitable for BC. We are not interested in which crossover method is best. Even though BC-SPX-NS outperforms BC-PCX-NS, it does not mean that SPX performs better than PCX. It only means that SPX is more suitable for BC than PCX.

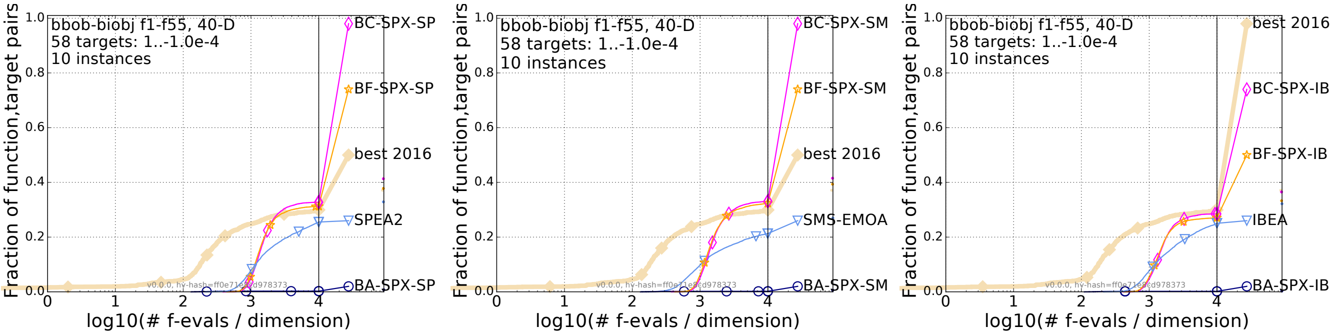
Figure 6 shows results of the three selections with SBX, BLX, PCX, and REX on all 55 BBOB problems with $n = 40$. Due to space constraint, only results for $n = 40$ are shown here. The NS ranking method is used in BA, BC, and BF. Figure 6 (a) shows

[2] The monotonic improvement of the hypervolume value over time is guaranteed only when using the unbounded external archive [25].



(a) SBX (b) BLX (c) PCX (d) REX

Figure 6: Results of BA, BF, and BC with (a) SBX, (b) BLX, (c) PCX, and (d) REX on all 55 BBOB problems with $n = 40$. Results of the original NSGA-II are also shown.



(a) SPEA2 (b) SMS-EMOA (c) IBEA

Figure 7: Results of BA, BF, and BC with the ranking methods in (a) SPEA2, (b) SMS-EMOA, and (c) IBEA on all 55 BBOB problems with $n = 40$. Results of the original SPEA2, SMS-EMOA, and IBEA are also shown.

that BA-SBX-NS outperforms BF-SBX-NS and BC-SBX-NS. This good performance of BA-SBX-NS is inconsistent with the results in Subsection 4.1. Since SBX can generate children far from their parents as shown in Figure 1, the distribution of children discussed in Subsection 4.2 does not significantly influence the performance of BA. However, BA-SBX-NS performs worse than NSGA-II. Figure 6 (b) and (c) show similar results. The evolution of the three selections with BLX and PCX clearly stagnates. Figure 6 (d) shows that results with REX are consistent with the results with SPX. BC-REX-NS is the best optimizer at the later stage. BF-REX-NS also performs better than NSGA-II.

In summary, SPX and REX are suitable for BC and BF, while SBX, BLX, and PCX are not suitable for them. These results indicate that crossover methods with the preservation of statistics are suitable for BC (and BF). As shown in Table 1, only SPX and REX satisfy the preservation of statistics among the five crossover methods. The results presented in [1] show that SPX and UNDX-n (a special version of REX) are suitable for JGG (a similar selection to BC) in GAs for single-objective continuous optimization. Interestingly, our results on continuous MOPs are consistent with the results on single-objective continuous optimization problems. A similarity analysis between single-objective optimizers and multi-objective optimizers as in [35] may be interesting.

4.5 Comparison of BA, BF, and BC with other ranking methods

We used the NS ranking method in Subsection 4.1. We investigate whether similar results can be obtained when using the SP, SM, and IB ranking methods (see Subsection 2.3).

Figure 7 shows the comparison of BA, BF, and BC with SP, SM, and IB for $n = 40$. We do not show results for $n \in \{2, 3, 5, 10, 20\}$, but they are similar to the results in Subsection 4.1. SPX is used as a crossover method. Figures 7 (a), (b), and (c) also show results of the original SPEA2, SMS-EMOA, and IBEA, respectively.

Figure 7 shows that results with SP, SM, and IB are consistent with the results with NS. BF and BC outperform the original SPEA2, SMS-EMOA, and IBEA at the later stage. BC is the best optimizer at the later stage. The poor performance of BA can be observed in Figure 7. Our results show that the relative performance of BA, BF, and BC does not significantly depend on the choice of a ranking method.

5 CONCLUSION

We examined the effectiveness of the two elitist selections (BA and BF) and the non-elitist selection (BC) on the bi-objective BBOB problem suite. We used five crossover methods and four ranking methods. For about two decades, it has been considered that elitist

EMOAs always outperform non-elitist EMOAs. Interestingly, our results show that the non-elitist BC performs better than the two elitist selections and the four original EMOAs (NSGA-II, SPEA2, SMS-EMOA, and IBEA) on the bi-objective BBOB problems with many decision variables when using the unbounded external archive and a crossover method with the preservation of statistics (i.e., SPX and REX). The choice of a ranking method does not significantly influence the relative performance of BC. We also analyzed the advantages and disadvantages of the non-elitist BC selection.

A number of interesting directions for future work remain. Although only elitist EMOAs have been studied in the 2000s, our results indicate that efficient non-elitist EMOAs could be realized. Designing non-elitist versions of MO-ES [35] and MO-CMA-ES [20] based on BC may be promising.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (Grant No. 61876075), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), Shenzhen Peacock Plan (Grant No. KQTD2016112514355531), the Science and Technology Innovation Committee Foundation of Shenzhen (Grant No. ZDSYS201703031748284), and the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008).

REFERENCES

[1] Y. Akimoto. 2010. *Design of Evolutionary Computation for Continuous Optimization.* Ph.D. Dissertation. Tokyo Institute of Technology.

[2] Y. Akimoto, J. Sakuma, I. Ono, and S. Kobayashi. 2009. Adaptation of expansion rate for real-coded crossovers. In *GECCO*. 739–746.

[3] Nicola Beume, Boris Naujoks, and Michael T. M. Emmerich. 2007. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *EJOR* 181, 3 (2007), 1653–1669.

[4] K. Bringmann, T. Friedrich, and P. Klitzke. 2014. Generic Postprocessing via Subset Selection for Hypervolume and Epsilon-Indicator. In *PPSN*. 518–527.

[5] D. Brockhoff, T. Tran, and N. Hansen. 2015. Benchmarking Numerical Multiobjective Optimizers Revisited. In *GECCO*. 639–646.

[6] D. Brockhoff, T. Tusar, D. Tusar, T. Wagner, N. Hansen, and A. Auger. 2016. Biobjective Performance Assessment with the COCO Platform. *CoRR* abs/1605.01746 (2016).

[7] D. Corne, J. D. Knowles, and M. J. Oates. 2000. The Pareto Envelope-Based Selection Algorithm for Multi-objective Optimisation. In *PPSN*. 839–848.

[8] K. Deb. 2001. *Multi-Objective Optimization Using Evolutionary Algorithms.* John Wiley & Sons.

[9] K. Deb and R. B. Agrawal. 1995. Simulated Binary Crossover for Continuous Search Space. *Complex Systems* 9, 2 (1995).

[10] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE TEVC* 6, 2 (2002), 182–197.

[11] K. Deb, A. Anand, and D. Joshi. 2002. A Computationally Efficient Evolutionary Algorithm for Real-Parameter Optimization. *Evol. Comput.* 10, 4 (2002), 345–369.

[12] K. Deb and T. Goel. 2001. Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence. In *EMO*. 67–81.

[13] K. Deb, M. Mohan, and S. Mishra. 2005. Evaluating the epsilon-Domination Based Multi-Objective Evolutionary Algorithm for a Quick Computation of Pareto-Optimal Solutions. *Evol. Comput.* 13, 4 (2005), 501–525.

[14] J. José Durillo and A. J. Nebro. 2011. jMetal: A Java framework for multi-objective optimization. *Adv. Eng. Softw.* 42, 10 (2011), 760–771.

[15] L. J. Eshelman and J. D. Schaffer. 1992. Real-Coded Genetic Algorithms and Interval-Schemata. In *FOGA*. 187–202.

[16] C. M. Fonseca and P. J. Fleming. 1993. Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization. In *ICGA*. 416–423.

[17] N. Hansen, A. Auger, O. Mersmann, T. Tusar, and D. Brockhoff. 2016. COCO: A Platform for Comparing Continuous Optimizers in a Black-Box Setting. *CoRR* abs/1603.08785 (2016).

[18] N. Hansen, S. Finck, R. Ros, and A. Auger. 2009. *Real-Parameter Black-Box Optimization Benchmarking 2009: Noiseless Functions Definitions.* Technical Report RR-6829. INRIA.

[19] T. Higuchi, S. Tsutsui, and M. Yamamura. 2000. Theoretical Analysis of Simplex Crossover for Real-Coded Genetic Algorithms. In *PPSN*. 365–374.

[20] C. Igel, N. Hansen, and S. Roth. 2007. Covariance Matrix Adaptation for Multi-objective Optimization. *Evol. Comput.* 15, 1 (2007), 1–28.

[21] H. Ishibuchi and T. Murata. 1998. A multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Trans. SMC, Part C* 28, 3 (1998), 392–403.

[22] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. 1983. Optimization by simulated annealing. *science* 220, 4598 (1983), 671–680.

[23] H. Kita, I. Ono, and S. Kobayashi. 1998. Theoretical Analysis of the Unimodal Normal Distiibution Crossover for Real-coded Genetic Algorithms. In *IEEE CEC*. 529–534.

[24] H. Kita, I. Ono, and S. Kobayashi. 1999. Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms. In *IEEE CEC*. 1581–1587.

[25] M. López-Ibáñez, J. D. Knowles, and M. Laumanns. 2011. On Sequential Online Archiving of Objective Vectors. In *EMO*. 46–60.

[26] S. W. Mahfoud. 1992. Crowding and Preselection Revisited. In *PPSN*. 27–36.

[27] K. Miettinen. 1998. *Nonlinear Multiobjective Optimization.* Springer.

[28] I. Ono and S. Kobayashi. 1997. A Real Coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distributed Crossover. In *GECCO*. 246–253.

[29] J. D. Schaffer. 1985. Multiple objective optimization with vector evaluated genetic algorithms. In *ICGA*. 93–100.

[30] H. K. Singh, K. S. Bhattacharjee, and T. Ray. 2019 (in press). Distance based subset selection for benchmarking in evolutionary multi/many-objective optimization. *IEEE TEVC* (2019 (in press)).

[31] N. Srinivas and K. Deb. 1994. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evol. Comput.* 2, 3 (1994), 221–248.

[32] S. Tsutsui, M. Yamamura, and T. Higuchi. 1999. Multi-parent Recombination with Simplex Crossover in Real Coded Genetic Algorithms. In *GECCO*. 657–664.

[33] T. Tusar, D. Brockhoff, N. Hansen, and A. Auger. 2016. COCO: The Bi-objective Black Box Optimization Benchmarking (bbob-biobj) Test Suite. *CoRR* abs/1604.00359 (2016).

[34] T. Tusar and B. Filipic. 2016. Performance of the DEMO Algorithm on the Bi-objective BBOB Test Suite. In *GECCO*. 1249–1256.

[35] S. Wessing, R. Pink, K. Brandenbusch, and G. Rudolph. 2017. Toward Step-Size Adaptation in Evolutionary Multiobjective Optimization. In *EMO*. 670–684.

[36] E. Zitzler and S. Künzli. 2004. Indicator-Based Selection in Multiobjective Search. In *PPSN*. 832–842.

[37] E. Zitzler, M. Laumanns, and L. Thiele. 2001. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm.* Technical Report. ETHZ.

[38] E. Zitzler and L. Thiele. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE TEVC* 3, 4 (1999), 257–271.