

# Efficient Parent Selection for Approximation-Guided Evolutionary Multi-Objective Optimization

Markus Wagner

Evolutionary Computation Group  
The University of Adelaide  
Adelaide, SA 5005, Australia

Tobias Friedrich

Friedrich-Schiller-Universität Jena  
Ernst-Abbe-Platz 2  
07743 Jena, Germany

**Abstract**—The Pareto front of a multi-objective optimization problem is typically very large and can only be approximated. Approximation-Guided Evolution (AGE) is a recently presented evolutionary multi-objective optimization algorithm that aims at minimizing iteratively the approximation factor, which measures how well the current population approximates the Pareto front. It outperforms state-of-the-art algorithms for problems with many objectives. However, AGE’s performance is not competitive on problems with very few objectives. We study the reason for this behavior and observe that AGE selects parents uniformly at random, which has a detrimental effect on its performance. We then investigate different algorithm-specific selection strategies for AGE. The main difficulty here is finding a computationally efficient selection scheme which does not harm AGEs linear runtime in the number of objectives. We present several improved selections schemes that are computationally efficient and substantially improve AGE on low-dimensional objective spaces, but have no negative effect in high-dimensional objective spaces.

## I. INTRODUCTION

Real-world optimization problems often aim at minimizing and/or maximizing several objectives at the same time. If these objectives are in conflict with each other (like e.g. minimizing production time and maximizing sales price), we get a Pareto front of all compromise solutions. The difficulty is that this set can be very large and can even be infinite for continuous optimization problems. The aim of evolutionary multi-objective optimization algorithms (EMOA) is therefore finding a *good approximation* of this Pareto front. Different EMOAs have (often implicitly) used very different definitions of what kind of approximation they want to optimize.

One popular measure is the volume of the objective space dominated by the set of solutions called *hypervolume* [29]. There are various well-established hypervolume-based EMOAs such as MO-CMA-ES [18] or SMS-EMOA [3]. This approach has theoretical and practical drawbacks. First, the hypervolume is computationally demanding. It cannot be computed in time polynomial in the number of objectives unless  $P = NP$  [5]. It is also NP-hard to determine which solution gives approximately the least contribution to the total hypervolume [6, 7]. It is also dissatisfying theoretically that maximizing the hypervolume does not imply that the resulting point set in the limit is any form of approximation of the true Pareto front. The only known relationship is that the worst-case approximation obtained by optimal hypervolume distributions is asymptotically equivalent to the best worst-case approximation achievable by all sets of the same size [4].

We aim at finding a small set of solutions with best possible *approximation of the Pareto set*. More formally, we want to find a population that additively  $\varepsilon$ -approximates the Pareto front with the smallest  $\varepsilon$  possible (a point  $\varepsilon$ -approximates all points which are (i) worse or (ii) better by at most  $\varepsilon$  in all objectives) [2]. For a rigorous definition, see Section II. This notion of multi-objective approximation was introduced in the 80’s by several authors [15, 17, 20, 23, 24] and its theoretical properties are extensively studied, e.g. in [9, 12, 21, 22, 25]. Laumanns et al. [19] have incorporated this notion of approximation in an EMOA. However, this algorithm is mainly of theoretical interest as the desired approximation is determined by a parameter of the algorithm and is not improved over time.

Two years ago, Bringmann et al. [8] introduced the framework of *Approximation-Guided Evolution (AGE)*. It uses above formal notion of approximation and improves the approximation quality during its iterative process. As the algorithm does not have complete knowledge about the true Pareto front, it uses the best knowledge obtained so far during the optimization process: it guides the search not by the approximation factor regarding the true Pareto front, but by the approximation factor regarding the set of all non-dominated points seen so far. For a more detailed description of the algorithm, see Section II. The experimental results in [8] show that given a fixed time budget it outperforms current state-of-the-art algorithms in terms of the desired additive approximation, as well as the covered hypervolume on standard benchmark functions. In particular, this holds for problems with many objectives, which most other algorithms have difficulties dealing with.

The runtime of AGE is linear in the number of objectives. This results in a very good performance for problems with many objectives, where it outperformed all competitors in [8]. However, the situation changes for problems with two or three objectives. There its performance is inferior to other EMOAs. A simple approach to solve this problem would be to pair AGE with one of the other algorithms and then decide deterministically which algorithm to use based on the number of objectives. However, we will show that with a minor adjustment to AGE, this pairing is not required. The key discovery is that the random parent selection of AGE is free of any bias. For problems with many objectives, this is not a problem, and can even be seen as its biggest advantage. For problems with few objectives, however, it is well known that algorithms can do better than random selection, such as selection based on crowding distance, hypervolume contribution, etc. Such strategies then select potential candidates based

---

**Algorithm 1:** Measure approximation quality of a population ([8])

---

**input :** Archive  $A$ , Population  $P$   
**output:** Indicator  $S_\alpha(A, P)$

```

1  $S \leftarrow \emptyset$ ;
2 foreach  $a \in A$  do
3    $\delta \leftarrow \infty$ ;
4   foreach  $p \in P$  do
5      $\rho \leftarrow -\infty$ ;
6     for  $i \leftarrow 1$  to  $d$  do
7        $\rho \leftarrow \max\{\rho, a_i - p_i\}$ ;
8      $\delta \leftarrow \min\{\delta, \rho\}$ ;
9    $S \leftarrow S \cup \{\delta\}$ ;
10 sort  $S$  decreasingly;
11 return  $S$ ;
```

---

on their relative position in the current population. For AGE, the lack of this bias means that solutions can be picked for parents that are not necessarily candidates with high potential. Consequently, it is not surprising to see that the original AGE is outperformed by algorithms that do well with their parent selection strategy, if their strategy is effective. This paper shows that a non-uniform parent selection significantly increases AGEs performance for small dimensions without loosing in higher dimensions or increasing its runtime.

The outline of this paper is as follows. We introduce some basic definitions as well as the original AGE in Section II. In Section III we outline the problem of finding a suitable selection scheme. Then, we motivate, investigate and compare different selection schemes for AGE. Finally, we finish with some conclusions.

## II. PRELIMINARIES

In our work, we consider minimization problems with  $d$  objective functions, where  $d \geq 2$  holds. Each objective function  $f_i: S \mapsto \mathbb{R}$ ,  $1 \leq i \leq d$ , maps from the considered search space  $S$  into the real values. In order to simplify the presentation we only work with the dominance relation on the objective space and mention that this relation transfers to the corresponding elements of  $S$ .

Given two points  $x = (x_1, \dots, x_d)$  and  $y = (y_1, \dots, y_d)$ , with  $x, y \in \mathbb{R}^d$  we define the following dominance relation:

$$\begin{aligned}
 x \preceq y &: \Leftrightarrow x_i \leq y_i \text{ for all } 1 \leq i \leq d, \\
 x \prec y &: \Leftrightarrow x \preceq y \text{ and } x \neq y.
 \end{aligned}$$

The typical notions of approximation used in theoretical computer science are multiplicative and additive approximation. We use the following definition

*Definition 1:* For finite sets  $S, T \subset \mathbb{R}^d$ , the additive approximation of  $T$  with respect to  $S$  is defined as

$$\alpha(S, T) := \max_{s \in S} \min_{t \in T} \max_{1 \leq i \leq d} (s_i - t_i).$$

This paper only regards additive approximation. However, this approach can be easily adapted to multiplicative approx-

imation where in Definition 1 just the term  $s_i - t_i$  has to be replaced by  $s_i/t_i$ .

The aim of AGE is to minimize the additive approximation of the population  $P$  we output with respect to the archive  $A$  of all points seen so far, i.e., we want to minimize  $\alpha(A, P)$ . The problem is that  $\alpha(A, P)$  is not sensitive to local changes of  $P$ .  $\alpha(A, P)$  only measures improvements of points which are currently worst approximated.

To get a sensitive indicator that can be used to guide the search, we consider instead the set  $\{\alpha(\{a\}, P) \mid a \in A\}$  of all approximations of the points in  $A$ . We sort this set decreasingly and call the resulting sequence

$$S_\alpha(A, P) := (\alpha_1, \dots, \alpha_{|A|}).$$

The first entry  $\alpha_1$  is again  $\alpha(A, P)$ . Our new goal it then to minimize  $S_\alpha(A, P)$  *lexicographically*. Note that this is an augmentation of the order induced by  $\alpha(A, P)$ : If we have  $\alpha(A, P_1) < \alpha(A, P_2)$  then we also have  $S_\alpha(A, P_1) <_{\text{lex}} S_\alpha(A, P_2)$ . Moreover, this indicator is locally sensitive. Algorithm 1 describes how to calculate it.

### A. Approximation-Guided Evolution

AGE [8] is an evolutionary multi-objective algorithm that works with a formal notion of approximation. The experimental results presented in [8] show that given a fixed time budget it outperforms current state-of-the-art algorithms in terms of the desired additive approximation, as well as the covered hypervolume on standard benchmark functions. In particular, this holds for problems with many objectives, which most other algorithms have difficulties dealing with.

Interestingly, it is the other way around when the problems have just very few objectives. As can be seen in Figure 4 (that will serve us for our final evaluation), the original AGE ( $\rightarrow$ ) is clearly outperformed by other algorithms in several cases when the problem has just two to three objectives. Of course, a simple approach to solve this problem would be to pair AGE with one of the other algorithms and then decide deterministically which algorithm to use based on the number of objectives. However, we will see, that with a minor adjustment to AGE, this pairing is not required.

The original AGE is outlined in Algorithm 2. One important observations can be made. In line 6, the parents for the mating process are selected uniformly at random. Interestingly, this random selection does not seem to be detrimental to the algorithm's performance on problems with many objectives. However, realizing that the selection process might be improved motivated us to investigate algorithm-specific selection processes. In Section III, we will show that the achievable quality can be improved significantly by selecting strategically.

### B. On Problems with Many Objectives

Let us start with an easy observation. An algorithm like NSGA-II that is known for its deficiencies on problems with many objectives<sup>1</sup> can be made *competitive* when compared to algorithms that seem to work better, such as IBEA and

---

<sup>1</sup>The cuboids used to determine the crowding distances fail to distinguish between points that are in the "nearest corner" and the "far-end corner".

---

**Algorithm 2:** Outline of the  $(\mu + \lambda)$ -Approximation Guided EA ([8])

---

```
1 Initialize population  $P$  with  $\mu$  random individuals;
2 Set archive  $A \leftarrow P$ ;
3 foreach generation do
4   Initialize offspring population  $O \leftarrow \emptyset$ ;
5   for  $j \leftarrow 1$  to  $\lambda$  do
6     Select two random individuals from  $P$ ;
7     Apply crossover and mutation;
8     Add new individual to  $O$ ;
9   foreach  $p \in O$  do
10    Insert offspring  $p$  in the archive  $A$  such that
    only non-dominated solutions remain;
11  Add offsprings to population, i.e.,  $P \leftarrow P \cup O$ ;
12  while  $|P| > \mu$  do
13    Remove  $p$  from  $P$  that is of least importance to
    the approximation (for details on this step see
    [8]);
```

---

SMS-EMOA, with a very simple modification: the modified NSGA-II focusses exclusively on the extreme points<sup>2</sup>. The result is truly stunning: with this simple modification, the algorithm iteratively improves the extreme points, and as the computational complexity of the algorithm is very low, it rushes to the true Pareto front in short time. Now, the task for a researcher is to assess the quality of the new algorithm by assessing the generated solutions. The problem is that for the class of DTLZ functions the true fronts are very small, and that an additive approximation of 1.0 or better can already be achieved if the algorithm finds all extreme points. This might be quite surprising: a rather simple algorithm can achieve higher quality solutions than a more complex one that tries to maintain a higher level of diversity.<sup>3</sup> This modified NSGA-II has one big disadvantage, if the task is to find a diverse set of solutions: points between the extreme points are very unlikely to be discovered, as the algorithm’s new focus is to find a good local optima for the individual objectives; the final population’s diversity is insufficient and the dominance of huge portions of the objective space would make it necessary for the multi-objective algorithm to “step back” in order to discover new multi-dimensional basements of attraction (volumes that are likely to attract the population and that lead to new local optima). Thus, one has created a fast algorithm that is incapable of finding intermediate compromise solutions.

We see this as another advantage of the epsilon indicator. Based on the knowledge gained from experiments, the actual “dimensions” of the Pareto front become clearer based on a single indicator value, and with the final quality assessments, a single number can make a reliable statement about the distribution of the points. Note that the approximation of a set of solutions with respect to some unknown Pareto front can be approximated as well: the non-dominated solutions of

---

<sup>2</sup>These are the solutions that have small objective values for at least one objective, but not necessarily for all objectives. Thus, they mark the ‘corners’ of the population in the objective space.

<sup>3</sup>Similarly, the hypervolume value will be larger, but as so often it does not indicate much about the actual distribution of the points in the objective space.

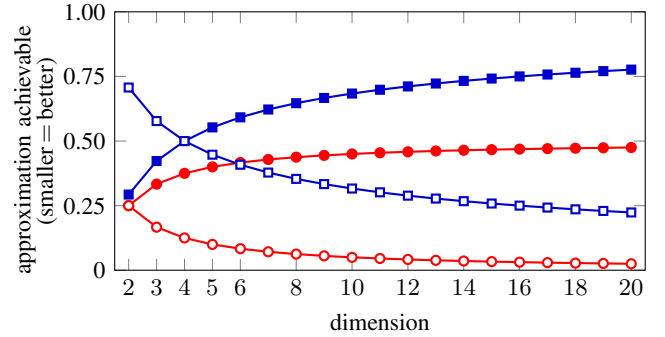


Fig. 1. Achievable additive approximations of the Pareto fronts, when only all corner points are found for DTLZ 1 (—●—) and for DTLZ 2/3/4 (—■—), and when only the centre of the Pareto front is found for DTLZ 1 (—○—) and for DTLZ 2/3/4 (—□—)

the union of all points found during multiple repetitions of the experiment can be seen as an approximation of the true Pareto front.

As we will later-on compare the algorithms on the DTLZ benchmark family [11], let us briefly highlight some of their properties. We used the functions DTLZ 1–4, each with 30 function variables and between 2 to 20 objective values/dimensions. The fronts of DTLZ 3 and DTLZ 4 are equivalent to DTLZ 2; they only differ in the mapping from the search space to the objective space. Regarding the objective spaces themselves, DTLZ 1 and DTLZ 3 are similar. Their Pareto fronts are merely points in the corner of voluminous multi-objective search spaces. Search points sampled uniformly at random from the search space  $[0, 1]^{30}$  typically have in the objective space an  $\ell_\infty$ -distance to the Pareto front of DTLZ 1 of 200–700. This implies that a random population has an additive approximation of this order of magnitude. Analogously, the objective spaces of DTLZ 2 and DTLZ 4 are similar. There, randomly drawn solutions typically have an  $\ell_\infty$ -distance of 1–2 from the Pareto front. This implies that for these two functions randomly drawn solutions achieve a significantly smaller (*better*) approximation of the Pareto front than randomly drawn solutions approximate DTLZ 2 and 4, i.e., random populations are significantly “closer” to the fronts of DTLZ 2 and 4 than to DTLZ 1 and 3. In practise, this means the following: DTLZ 2/4 proved to be especially hard to optimise, when compared to DTLZ 1/3, as the objective space is significantly smaller. Even though it is easy to find some of the corner points, it proves to be difficult to find all corner points, and to maintain a diverse population at the same time.

To understand the consequences of the corner points of the Pareto fronts on the achievable additive epsilon approximation, let us consider Figure 1. The red and blue plots show the achievable approximations for DTLZ 1 and DTLZ 2/3/4. As it can be seen, a single solution that is placed on the centre of the front can achieve very good approximation values, whereas a good approximation via the corner points is very difficult to achieve. Consequently, based on this figure, it is possible to determine whether an algorithm is capable of finding intermediate compromise solutions even when the dimensionality of the objective space makes it difficult for

many people to get an intuitive understanding of the achieved indicator values.

### III. APPROXIMATION-SPECIFIC PARENT SELECTION

We are seeking to improve AGE’s performance, subject to the following conditions:

- 1) The introduced computation time required to select parents should be polynomial in the number of objectives  $d$ .
- 2) The selection mechanism should significantly improve the performance on problems with few objectives, while not influencing the performance on problems with many objectives.
- 3) The selection scheme should favour individuals that have the potential to improve the approximation quality.

Note that most hypervolume-based algorithms, such as SMS-EMOA and MO-CMA-ES, violate condition (1), as some of the computations that are associated with the selection process take time exponential in  $d$ . However, we have to note that it is possible to deal with this drawback by approximating the hypervolume, as shown and demonstrated in [26]. Nevertheless, we opted against the use of (approximated) hypervolume-based indicators because their computational approximation comes at the cost of reduced final quality, and because maximization of the hypervolume can interfere with our goal of improving the approximative quality.

Also note that the exclusive use of domination based-criteria is problematic: assuming a general  $d$ -dimensional unbounded space (with  $d \geq 2$ ), then a point in this space dominates  $1/2^d$  of the volume. Obviously then, a pure dominance check in high-dimensional spaces is extremely likely to fail. Or, when interpreted the other way around, this means that a check of the dominance relation between two solutions is extremely unlikely to bring up any additional information about the relative quality between these two solutions.

As already discussed in Section II-B, we found out in preliminary testing that it is very easy to design algorithms that easily discover the corner points of the Pareto front. However, the problem of finding points “between” those corner points proved to be much more difficult. Selection mechanisms for the (fitness-based) parent selection and the offspring selection tend to have different biases that result in different preferences for corner points or central points, depending on the “shape” of the intermediate populations and on the shape of the true Pareto front. With an increasing number of dimensions, this problem becomes even more apparent, as solutions should evenly cover the front, while not concentrating only on extreme points.

#### A. Experimental Setup

Our selection strategies were implemented in the jMetal framework [13].<sup>4</sup> We compared the performance of our selection schemes against each other, and finally the scheme that we deemed best to the original AGE, and to the established MOO algorithms IBEA [28], NSGA-II [10], MO-CMA-ES [18], and SMS-EMOA [14], and SPEA2 [30]

on the DTLZ benchmark family [11]. The test setup is identical to that of [8]. We used the functions DTLZ 1-4, each with 30 function variables and between 2 to 20 objective values/dimensions. The Pareto front in the objective space of DTLZ 1 is a hyperplane. The Pareto fronts of DTLZ 2, DTLZ 3 and DTLZ 4 are hyperspheres; they only differ in the mapping from the search space to the objective space. Except for the original AGE, parents are selected via a binary tournament in all algorithms. We limit the calculations of the algorithms to a maximum of 100,000 fitness evaluations *and* a maximum computation time of 4 hours per run. Note that the time restriction had to be used as the runtime of some algorithms increases exponentially with respect to the size of the objective space.

As variation operators, the polynomial mutation and the simulated binary crossover [1] were applied, which are both used widely in MOO algorithms [10, 16, 30]. The distribution parameters associated with the operators were  $\eta_m = 20.0$  and  $\eta_c = 20.0$ . The crossover operator is biased towards the creation of offspring that are close to the parents, and was applied with  $p_c = 0.9$ . The mutation operator has a specialized explorative effect for MOO problems, and was applied with  $p_m = 1/(\text{number of variables})$ . Population size was set to  $\mu = 100$  and  $\lambda = 100$ . Each setup was given a budget of 100,000 evaluations. In the following, we report the averages of 100 independent runs.

We assess the selection schemes and algorithms using the *approximation* measure. However, as it is difficult to compute the exact achieved additive approximation of a known Pareto front, we approximate it. First, we draw one million points of the front uniformly at random. Then, we compute the additive approximation that the final population achieved for this set ([8]).

#### B. Step 1 — Single-Level Selection Mechanisms

As the basis of our investigations, we use AGE as defined in [8]. The key discovery was that the random parent selection of AGE is free of any bias. For problems with many objectives, this is not a problem, and can even be seen as an advantage. For problems with few objectives, however, it is well known that one can do better than random selection, such as selection based on crowding distance, hypervolume contribution, etc. Such strategies then select potential candidates based on their relative position in the current population. For AGE, the lack of this bias means that solutions can be picked for parents that are not necessarily candidates with high potential. Consequently, it is not surprising to see that AGE is outperformed by algorithms that do well with their parent selection strategy.

The following are our selection schemes:

- 1) RS: Parents are drawn randomly from the population, following the original AGE.
- 2) RS<sup>front1</sup>: First, the population is sorted into fronts of non-dominating solutions, then parents are drawn randomly from only the first front of the population, as these are expected to be good candidates to improve the approximation of the archive.
- 3) RS<sup>thinning</sup>: After thinning the population (a solution is dropped from the population<sup>5</sup> with  $p = 0.5$ ),

<sup>4</sup>The code is available here <http://cs.adelaide.edu.au/~ec/research/age.php>.

<sup>5</sup>Note that the solution will remain in the archive.

parents are drawn randomly from it. Via this disruptive thinning, the exploration of the search space per generation is increased, which allows for the discovery of new multi-dimensional basements of attraction.

- 4)  $BT^{\text{front\_no}}$ : First, the population is sorted into fronts of non-dominating solutions, then the fitness  $F(X)$  of a solution  $X$  is set to its front number before then binary tournament is performed (preferring smaller front numbers).

Note that we use superscript and subscript to indicate the approaches used for fitness assignment and population thinning.

We apply binary tournaments for candidates  $X$  and  $Y$  as follows. In a first step, dominance is checked, and the dominating solution wins the tournament. If the two solutions are incomparable, then the fitness values  $F(X)$  and  $F(Y)$  are compared and the better solution wins the tournament.

Note that the dominance check in binary tournament does not render the ranking in  $BT^{\text{front\_no}}$  obsolete: two solutions that are in different ranks can still be incomparable. In particular, this is likely to be the case when the number of objectives is high.

The results of the experimental investigations are shown in Figure 2. First, it can be observed that the different schemes have a measurable impact on the approximation quality achieved for problems with few objectives. At the same time, the performance on problems with many objectives remains relatively unchanged. However, there is no single scheme that performs well on all four different function classes, when one compares their performance for low values of  $d$ :

- The very aggressive selection scheme  $RS^{\text{front1}}$  (—■) performs very well on the low-dimensional DTLZ 1 and DTLZ 3 variants, but loses to every other scheme on the DTLZ 4 variants.
- $RS^{\text{thinning}}$  (—▲) maintains some of our base-AGE’s good performance on low-dimensional DTLZ 4 variants, and can be seen as a useful trade-off between the base-AGE and the aggressive  $RS^{\text{front1}}$ , as even dominated solutions can become parents.
- $BT^{\text{front\_no}}$  (—★) on average shows a performance that is comparable to that of  $RS^{\text{thinning}}$ . While the assignment of the rank numbers as the fitness values proves to bring small benefits in some cases over the random selection, it is in other cases even minimally detrimental.
- The selection schemes have no significant effects on the resulting achieved additive approximations for the DTLZ 2 variants.

### C. Step 2 — Incorporating Crowding Distance

In general, NSGA-II’s crowding distance proves to be a good tool to maintain diverse populations on problems with two or three objectives. In the following, we build upon this observation, and we investigate empirically different selection schemes:

- 1)  $BT^{\text{crowding}}$ : First, the population is sorted into fronts, then the fitness of a solution is set to its crowding distance, then a binary tournament is performed (preferring larger crowding distance values).
- 2)  $BT^{\text{front\_no,crowding}}$ : A binary tournament is used, where the individuals are compared firstly according to their front number (preferring lower numbers). In case of equal front numbers, the solution with the larger crowding distance is selected.
- 3)  $BT^{\text{crowding}}_{\text{thinning\_firstfronts}}$ : First, the population is reduced: solutions in the front  $i$  have a probability of  $1/i$  of staying in the population. Subsequently, a binary tournament is performed where solutions of higher crowding distance are preferred. Thus, solutions that are dominated multiple times are less likely to be selected as a potential parent.
- 4)  $BT^{\text{crowding}}_{\text{thinning\_center}}$ : First, the population is reduced: solutions within one front are ordered per objective, then a solution at position  $i$  (of a total  $t$ ) has probability  $1/i$  to stay in the population if  $i < t/2$ , else  $1/t - i$ . This has the effect that more “central” solutions have a higher chance of getting dropped from the population. Subsequently, a binary tournament is performed where solutions of higher crowding distance are preferred. This parent selection mechanism is also used in [27].

The consequence of the thinning approaches used in the latter two schemes is that all solutions that form the first front (including the extreme points) are kept in the population.

The achieved additive approximations are shown in Figure 3. The main observations are as follows:

- The performance of  $BT^{\text{crowding}}$  (—◆) on the low-dimensional DTLZ 4 variants is remarkable, and in many cases better than anything that we have seen so far (including other algorithms). This comes, however, at a price: its performance on the DTLZ 1 and DTLZ 3 variant is easily beaten by the other selection mechanisms.
- $BT^{\text{front\_no,crowding}}$  (—■) represents an alternative to  $BT^{\text{crowding}}$ , as it incorporates both the rank number as a fitness values (proved to be beneficial in the previous section) and the crowding distance. As a result, it inherits some of the good properties from both approaches: the good performance on DTLZ 1/3 and on DTLZ 4.
- Both  $BT^{\text{crowding}}_{\text{thinning\_linear}}$  (—▲) and  $BT^{\text{crowding}}_{\text{thinning\_center}}$  (—★) perform very similarly, indicating that the choice of the thinning scheme may be just of minor importance.
- Again, the selection schemes have no significant effects on the resulting achieved additive approximations for the DTLZ 2 variants.

### D. Comparison With Other Algorithms

Based on the previous experiments, we choose  $BT^{\text{crowding}}_{\text{thinning\_center}}$  for the final comparison against the established algorithms. In the following, we will refer to

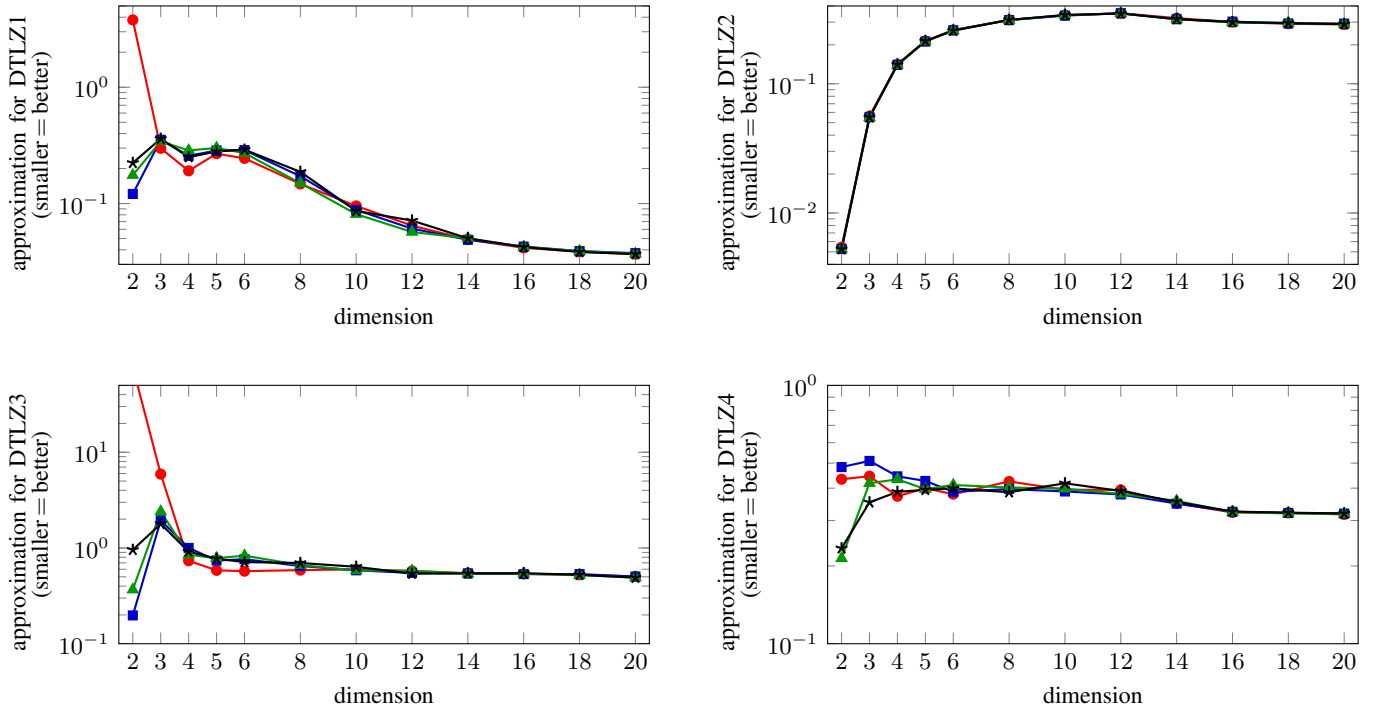


Fig. 2. Comparison of the performance of the different parent selection mechanisms: RS (random selection from the population, —●— ([8])), RS<sup>front1</sup> (random selection from the first front, —■—), RS<sup>thinning</sup> (random selection after thinning, —▲—), BT<sup>front\_no</sup> (the fitness is the front number, —★—) with varying dimension  $d$ . The figures show the average of 100 repetitions each.

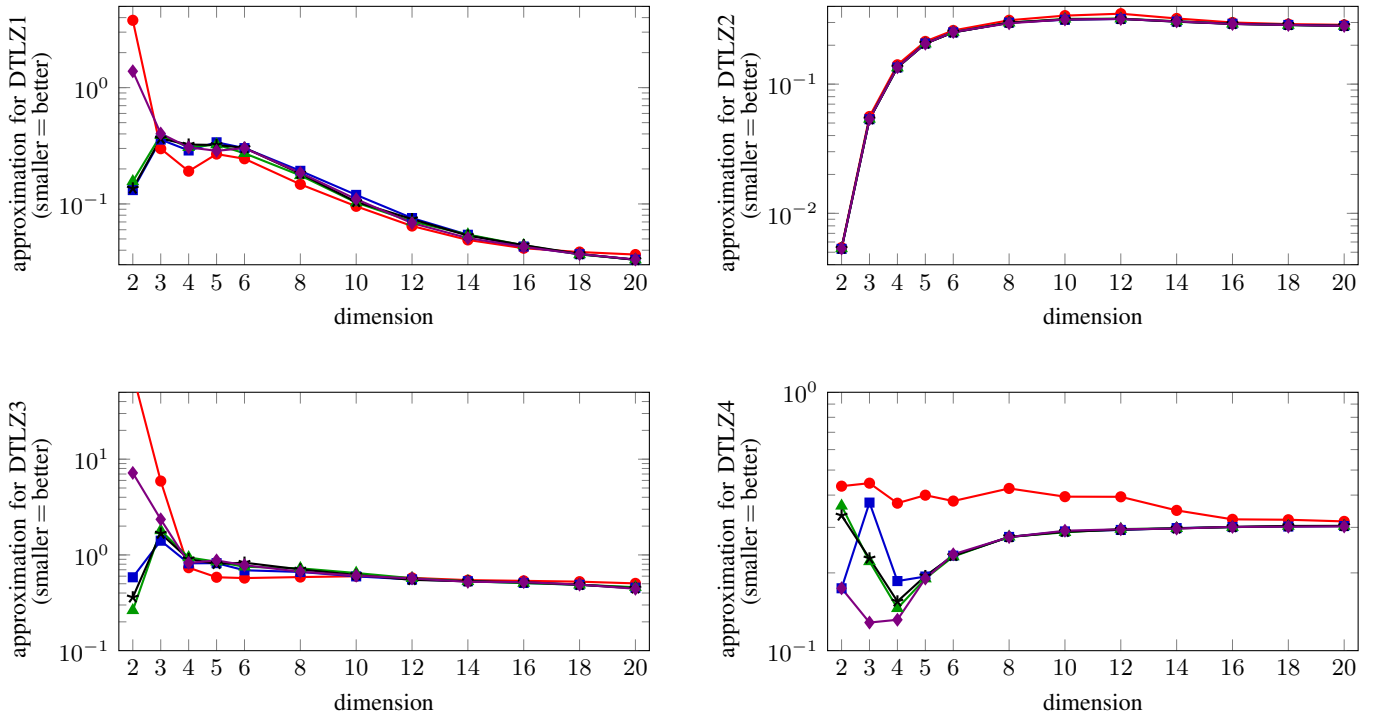


Fig. 3. Comparison of the performance of the different parent selection mechanisms: RS (random selection from the population, —●— ([8])), BT<sup>crowding</sup> (the fitness is the crowding distance within each front, —◆—), BT<sup>front\_no.crowding</sup> (front number and crowding, —■—), BT<sup>crowding\_thinning\_center</sup> (thinning towards each front's centre, —▲—), BT<sup>crowding\_thinning\_firstfronts</sup> (thinning prefers first fronts, —★—) with varying dimension  $d$ . The figures show the average of 100 repetitions each.

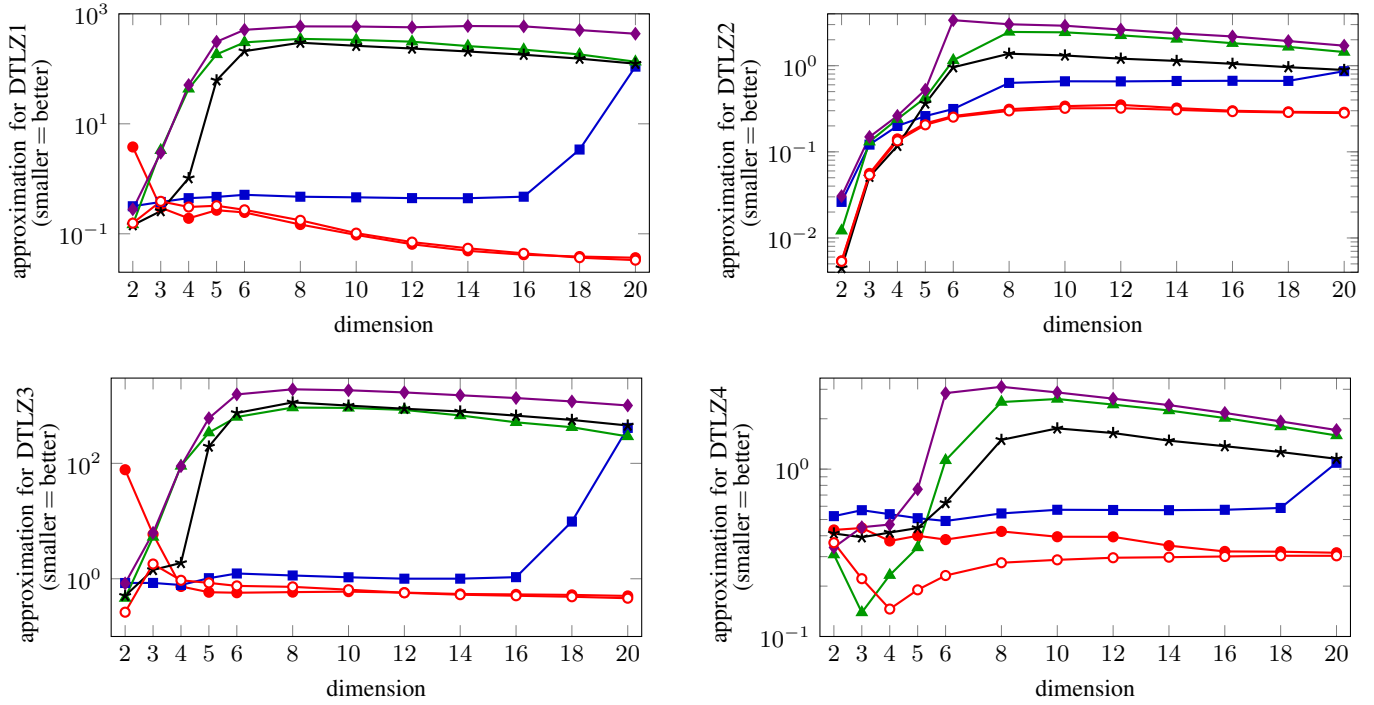


Fig. 4. Comparison of the performance of our AGE-PS ( $\circ$ ) with strategic parent selection with the original AGE ( $\bullet$ ), IBEA ( $\square$ ), NSGA-II ( $\blacktriangle$ ), SMS-EMOA ( $\star$ ), and SPEA2 ( $\blacklozenge$ ) with varying dimension  $d$ . The figures show the average of 100 repetitions each.

the combination of AGE with the parent selection scheme  $BT_{\text{thinning\_center}}^{\text{crowding}}$  as AGE-PS. The results are shown in Figure 4.

Observations:

- AGE-PS ( $\circ$ ) shows a significantly improved performance on the lower-dimensional variants of DTLZ 1, DTLZ 3, and DTLZ 4. Furthermore, it is either the best performing algorithm, or in many cases, it shows at least competitive performance.
- It is interesting to see that even though our AGE-PS incorporates the crowding distance idea from NSGA-II ( $\blacktriangle$ ) for a fitness assignment, it is not influenced by its detrimental effects in higher dimensional objective spaces. This is thanks to the way how the next generation is formed in AGE (i.e., based on contributions to the approximation achieved of the archive).
- When compared with the original AGE ( $\bullet$ ), then our modification results in a performance improvement in nearly all cases. Still, as AGE-PS shows a consistent performance across all scaled functions, we deem the minimal loss in quality (in our experimental setup) as negligible.
- Remarkably, NSGA-II ( $\blacktriangle$ ), SMS-EMOA ( $\star$ ), and SPEA2 ( $\blacklozenge$ ) are unable to find the front of the high-dimensional DTLZ 1 and DTLZ 3 variants. This results in extremely large approximation values.
- The reason for IBEA's ( $\square$ ) decreasing behavior for very large dimension ( $d \geq 18$ ) is that it was stopped after 4 hours and it could not perform 100,000

iterations. The same holds already for much smaller dimensions in the case of SMS-EMOA ( $\star$ ), which uses an exponential-time algorithm to internally determine the hypervolume. It did not finish a single generation for  $d \geq 8$  and only performed around 5,000 iterations within four hours for  $d = 5$ . This implies that the higher-dimensional approximations plotted for SMS-EMOA actually show the approximation of the random initial population.

- Interestingly, the final approximations achieved by NSGA-II ( $\blacktriangle$ ) and SPEA2 ( $\blacklozenge$ ) are even worse than those of their initial populations. Because they are tuned for low-dimensional problems, they spread out their populations too much, pushing them to the boundaries for high dimensions.
- Lastly, it is interesting to see that even though DTLZ 1-4 are sometimes considered to be 'easy', the algorithms perform very differently on the functions. Especially when the number of objectives is increased, the advantages and disadvantages of the selection criteria become apparent.

#### IV. CONCLUSIONS

Compared to the original AGE with random parent selection, the proposed AGE with strategic parent selection is competitive with state-of-the-art algorithms on problems with few objectives. On problems with many objectives, however, it maintains the dominance over existing algorithms in the achieved additive approximation. The code is available at <http://cs.adelaide.edu.au/~ec/research/age.php>.

## REFERENCES

- [1] R. B. Agrawal and K. Deb. Simulated binary crossover for continuous search space. Technical report, 1994.
- [2] R. Berghammer, T. Friedrich, and F. Neumann. Convergence of set-based multi-objective optimization, indicators, and deteriorative cycles. *Theoretical Computer Science*, 456:2 – 17, 2012.
- [3] N. Beume, B. Naujoks, and M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181: 1653–1669, 2007.
- [4] K. Bringmann and T. Friedrich. Approximation quality of the hypervolume indicator. *Artificial Intelligence*, 195: 265–290, 2013.
- [5] K. Bringmann and T. Friedrich. Approximating the volume of unions and intersections of high-dimensional geometric objects. *Computational Geometry: Theory and Applications*, 43:601–610, 2010.
- [6] K. Bringmann and T. Friedrich. An efficient algorithm for computing hypervolume contributions. *Evolutionary Computation*, 18:383–402, 2010.
- [7] K. Bringmann and T. Friedrich. Approximating the least hypervolume contributor: NP-hard in general, but fast in practice. *Theoretical Computer Science*, 425:104–116, 2012.
- [8] K. Bringmann, T. Friedrich, F. Neumann, and M. Wagner. Approximation-guided evolutionary multi-objective optimization. In *Proc. 22nd International Joint Conference on Artificial Intelligence (IJCAI '11)*, pp. 1198–1203, Barcelona, Spain, 2011. IJCAI/AAAI.
- [9] C. Daskalakis, I. Diakonikolas, and M. Yannakakis. How good is the Chord algorithm? In *Proc. 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '10)*, pp. 978–991, 2010.
- [10] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evolutionary Computation*, 6:182–197, 2002.
- [11] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary Multiobjective Optimization, Advanced Information and Knowledge Processing*, pp. 105–145. 2005.
- [12] I. Diakonikolas and M. Yannakakis. Small approximate Pareto sets for biobjective shortest paths and other problems. *SIAM Journal on Computing*, 39:1340–1371, 2009.
- [13] J. J. Durillo, A. J. Nebro, and E. Alba. The jMetal framework for multi-objective optimization: Design and architecture. In *Proc. Congress on Evolutionary Computation (CEC '10)*, pp. 4138–4325. IEEE Press, 2010.
- [14] M. T. M. Emmerich, N. Beume, and B. Naujoks. An EMO algorithm using the hypervolume measure as selection criterion. In *Proc. Third International Conference on Evolutionary Multi-Criterion Optimization (EMO '05)*, pp. 62–76. Springer, 2005.
- [15] Y. G. Evtushenko and M. Potapov. Methods of numerical solution of multicriterion problem. In *Soviet mathematics – doklady*, Vol. 34, pp. 420–423, 1987.
- [16] M. Gong, L. Jiao, H. Du, and L. Bo. Multiobjective immune algorithm with nondominated neighbor-based selection. *Evolutionary Computation*, 16:225–255, 2008.
- [17] P. Hansen. Bicriterion path problems. In *Multiple Criteria Decision Making: Theory and Applications*, Vol. 177 of *Lecture Notes in Economics and Mathematical Systems*, pp. 109–127, 1980.
- [18] C. Igel, N. Hansen, and S. Roth. Covariance matrix adaptation for multi-objective optimization. *Evolutionary Computation*, 15:1–28, 2007.
- [19] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler. Combining convergence and diversity in evolutionary multi-objective optimization. *Evolutionary Computation*, 10(3): 263–282, 2002.
- [20] P. Loridan.  $\varepsilon$ -solutions in vector minimization problems. *Journal of Optimization Theory and Applications*, 43: 265–276, 1984.
- [21] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Proc. 41st Annual Symposium on Foundations of Computer Science (FOCS '00)*, pp. 86–92. IEEE Press, 2000.
- [22] C. H. Papadimitriou and M. Yannakakis. Multiobjective query optimization. In *Proc. 20th ACM Symposium on Principles of Database Systems (PODS '01)*, pp. 52–59, 2001.
- [23] H. Reuter. An approximation method for the efficiency set of multiobjective programming problems. *Optimization*, 21:905–911, 1990.
- [24] G. Ruhe and B. Fruhwirth.  $\varepsilon$ -optimality for bicriteria programs and its application to minimum cost flows. *Computing*, 44:21–34, 1990.
- [25] S. Vassilvitskii and M. Yannakakis. Efficiently computing succinct trade-off curves. *Theor. Comput. Sci.*, 348:334–356, 2005.
- [26] T. Voß, T. Friedrich, K. Bringmann, and C. Igel. Scaling up indicator-based MOEAs by approximating the least hypervolume contributor: A preliminary study. In *Companion of 12th Annual Conference on Genetic and Evolutionary Computation Conference (GECCO '10)*, pp. 1975–1978. ACM Press, 2010.
- [27] M. Wagner and F. Neumann. A fast approximation-guided evolutionary multi-objective algorithm. In *Proc. 15th Annual Conference on Genetic and Evolutionary Computation (GECCO '13)*. ACM Press, 2013.
- [28] E. Zitzler and S. Künzli. Indicator-based selection in multiobjective search. In *Proc. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, Vol. 3242 of *LNCS*, pp. 832–842. Springer, 2004.
- [29] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evolutionary Computation*, 3:257–271, 1999.
- [30] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Proc. Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001)*, pp. 95–100, 2002.