



This is a repository copy of *PSA based multi objective evolutionary algorithms*.

White Rose Research Online URL for this paper:  
<http://eprints.whiterose.ac.uk/98424/>

Version: Accepted Version

---

**Proceedings Paper:**

Salomon, S., Domínguez-Medina, C., Avigad, G. et al. (4 more authors) (2014) PSA based multi objective evolutionary algorithms. In: *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation III*. *EVOLVE*, August 2012, Mexico City, Mexico. *Studies in Computational Intelligence*, 500 . Springer , pp. 233-259.

[https://doi.org/10.1007/978-3-319-01460-9\\_11](https://doi.org/10.1007/978-3-319-01460-9_11)

---

**Reuse**

Unless indicated otherwise, fulltext items are protected by copyright with all rights reserved. The copyright exception in section 29 of the Copyright, Designs and Patents Act 1988 allows the making of a single copy solely for the purpose of non-commercial research or private study within the limits of fair dealing. The publisher or other rights-holder may allow further reproduction and re-use of this version - refer to the White Rose Research Online record for this item. Where records identify the publisher as the copyright holder, users can verify any specific terms of use on the publisher's website.

**Takedown**

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing [eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk) including the URL of the record and the reason for the withdrawal request.



[eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk)  
<https://eprints.whiterose.ac.uk/>

# PSA based Multi Objective Evolutionary Algorithms

Shaul Salomon<sup>1</sup>, Christian Domínguez-Medina<sup>2</sup>, Gideon Avigad<sup>3</sup>, Alan Freitas<sup>4</sup>, Alex Goldvard<sup>3</sup>, Günter Rudolph<sup>5</sup>, Oliver Schütze<sup>6</sup>, and Heike Trautmann<sup>7</sup>

**Abstract** It has generally been acknowledged that both proximity to the Pareto front and a certain diversity along the front, should be targeted when using evolutionary multiobjective optimization. Recently, a new partitioning mechanism, the *Part and Select Algorithm* (PSA), has been introduced. It was shown that this partitioning allows for the selection of a well-diversified set out of an arbitrary given set, while maintaining low computational cost. When embedded into an evolutionary search (NSGA-II), the PSA has significantly enhanced the exploitation of diversity. In this paper, the ability of the PSA to enhance evolutionary multiobjective algorithms (EMOAs) is further investigated. Two research directions are explored here. The first one deals with the integration of the PSA within an EMOA with a novel strategy. Contrary to most EMOAs, that give a higher priority to proximity over diversity, this new strategy promotes the balance between the two. The suggested algorithm allows some dominated solutions to survive, if they contribute to diversity. It is shown that such an approach substantially reduces the risk of the algorithm to fail in finding the Pareto front. The second research direction explores the use of the PSA as an archiving selection mechanism, to

---

<sup>1</sup> Department of Automatic Control and Systems Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK. e-mail: ssalomon1@sheffield.ac.uk

<sup>2</sup> Computer Research Center, CIC-IPN, Av. Juan de Dios Bátiz Esq. Miguel Othon de Mendizabal, 07738, Mexico City, Mexico. e-mail: hdomigueza@sagitario.cic.ipn.mx

<sup>3</sup> ORT Braude College of Engineering, Snunit 51, Karmiel 21982, Israel. e-mail: gideona@braude.ac.il, goldvard@braude.ac.il

<sup>4</sup> Programa de Pós-Graduação em Engenharia Elétrica - Universidade Federal de Minas Gerais - Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil. e-mail: alandefreitas@gmail.com

<sup>5</sup> TU Dortmund University, 44227 Dortmund, Germany. e-mail: Guenter.Rudolph@tu-dortmund.de

<sup>6</sup> Computer Science Department, CINVESTAV-IPN, Av. IPN 2508, Col. San Pedro Zacatenco, Mexico City, Mexico. e-mail: schuetze@cs.cinvestav.mx

<sup>7</sup> Information Systems and Statistics, University of Münster, Leonardo-Campus 3, 48149 Münster, Germany. e-mail: trautmann@wi.uni-muenster.de

improve the averaged Hausdorff distance obtained by existing EMOAs. It is shown that the integration of the PSA into NSGA-II-I and  $\Delta_p$ -EMOA as an archiving mechanism leads to algorithms that are superior to base EMOAs on problems with disconnected Pareto fronts.

## 1 Introduction

In many real-world applications, several objectives must be optimized at the same time, leading to a multi-objective optimization problem (MOP). Mathematically, a MOP can be stated as follows:

$$\min_{\mathbf{x} \in Q} \mathbf{F}(\mathbf{x}) \quad (1)$$

where  $Q \subset \mathbb{R}^d$  is a domain in  $d$ -dimensional real space,  $\mathbf{F}(\mathbf{x})$  is defined as the vector of the  $k$  objective functions:

$$\mathbf{F}(\mathbf{x}) = [f_1(\mathbf{x}), \dots, f_k(\mathbf{x})]^T$$

where each objective function  $f_i(\mathbf{x})$ ,  $i = 1, \dots, k$ , maps the vector  $\mathbf{x} \in \mathbb{R}^d$  to  $\mathbb{R}$ . The set of optimal solutions of the problem (1) is usually called the Pareto set  $\mathcal{P}$ . The task of many set-oriented search procedures is to find a suitable finite sized approximation of the Pareto front  $\mathbf{F}(\mathcal{P})$  (i.e., the image of the Pareto set), since this front represents the set of optimal compromises measured in objective space, which usually is of primary interest. Out of the set-oriented search procedures for the numerical treatment of MOPs, EMOAs are widely used due to their global and universal approach and their high robustness [1, 2]. Most EMOAs simultaneously attempt to account for both proximity of the approximation set to the Pareto front and its diversity [3]. It has been indicated in [3] that both proximity and diversity should be explored and exploited during the evolutionary search. Exploration of diversity and proximity may be related to the selection of the next generation's parents and/or the control of crossover/mutation rates [4], [5], [6]. For example, in [4] the authors suggested an adaptive variation operator that exploits the chromosomal structure (binary representation) and controls crossover/mutation rates during the evolution in order maximize the information gain and to prevent information flow disruption between the different chromosomal structures. Within the exploration phase, the authors in [7] suggested to iteratively explore for good children through iterative density estimation of different optional children combinations. In that work good candidate parents have been searched for through clustering of their related performances in the objective space. It should be noted that this procedure is applied only to non-dominated candidate solutions.

On the other hand, exploitation of proximity and diversity is related to the selection of the solutions that will be saved for the next generation (through elitism or archiving) and will take place in reproduction. Domination is the predominant approach used to exploit proximity to the true Pareto front. Diversity is exploited by different approaches that can be classified into three main categories. The first treats diversity as a property of a set and evolves sets with a good diversity. The diversity can be measured according to the accumulated distances between the members of the set [8], [9], or indirectly by the hypervolume measure [10] or the averaged Hausdorff distance  $\Delta_p$  [11]. Algorithms in the second category treat diversity as a property of each individual according to the density of solutions surrounding it. Fitness sharing of NPGA [12], crowding distance of NSGA-II [13], the diversity metric based on entropy [14] and the density estimation technique used in SPEA2 [15] are examples of this category. Algorithms of the third category decompose the multi-objective problem into a number of single objective problems (scalarization). Each of these problems ideally aims for a different zone on the Pareto front such that the set of solutions to the auxiliary problems form a diverse set of optimal solutions. MOEA/D [16] is probably the most famous method within this category. A recent method from this category [17] combines Pareto dominance with Chebyshev decomposition for the selection process.

When selection takes place for the sake of exploiting proximity and diversity, proper selection criteria must be formulated, in order to achieve a balance among these two inspirations. Such a balance is not easy to achieve because it has been shown that these motivations are contradicting [18]. An improvement in one usually involves regression in the other. A balance between proximity and diversity within the exploitation phase has been targeted in various ways. One way is to select the elite population by pure truncation selection. In truncation selection, the algorithm sorts all individuals based on their domination level and includes the first individuals as the elite population. Truncation selection is exploited in many EMOAs, such as NSGA-II [13] and SPEA2 [15]. In those algorithms the exploitation of proximity takes over that of diversity as the solutions are primarily chosen based on domination relations. Some efforts to overcome this drawback have been made e.g., using the Balanced Truncation Selection (BTS) [7] within MIDEA (Multi-objective Mixture based Iterated Density Estimation Evolutionary Algorithm). In that algorithm, the exploitation of diversity can be improved by a tuned truncation threshold. The idea is to include in the elite population more diverse solutions by allowing higher truncation threshold values at the beginning of the search. It is noted that also in this algorithm, the non-dominated solutions will be preferred over dominated solutions. In other words, a solution dominated by most of the population will not be selected even though it is most isolated.

Another way to allow for a better balance between proximity and diversity is to change the dominance relation among the solutions by changing the area considered as dominated by a solution. Laumanns and Ocenasek [19]

proposed to use the concept of  $\varepsilon$ -dominance [20] which is a modification of the original Pareto dominance. The underlying principle of  $\varepsilon$ -dominance is that two solutions are not allowed to be non-dominated to each other, if the difference between them is less than a properly chosen value. Extensions based on this idea are the CDAS [21], where the user can control the size of a solution's dominated area and the cone  $\varepsilon$ -dominance [18], where the shape of the dominated area is a cone.

Recently, the Part and Select Algorithm (PSA) was introduced to select a diverse subset from a given set of points [22]. This mechanism has a low computational complexity, and it is capable to select a diverse subset, of any size, even if the original set is poorly distributed. These properties make PSA suitable as a selection mechanism within EMOAs. It has been shown in [22] that the integration of the PSA into NSGA-II improves its ability to find a diverse approximated set. In [23] a niching mechanism based on the PSA was used to find a set of different cross sections for a topology optimization problem.

In this paper, the ability of the PSA to improve EMOAs is further investigated. Two research directions are explored here. The first one deals with embedding the PSA within a novel genetic algorithm. The algorithm adjusts the balance between proximity and diversity by allowing some dominated solutions to survive if they improve the diversity. The second one explores the use of the PSA as an archiving selection mechanism, to improve the averaged Hausdorff distance  $\Delta_p$  obtained by existing EMOAs.

The remainder of this paper is organized as follows. The PSA is described in Section 2, and its previous utilization within an EA is briefly surveyed. In Section 3 a novel PSA based EMOA with an adjustable parameter to control the trade-off between proximity and diversity is introduced. The effect of this parameter is studied, and a comparison with NSGA-II-PSA is conducted in Section 4 to highlight the algorithm's advantage in dealing with a poor initial population. The implementation of PSA as an archiving mechanism integrated into NSGA-II-I and  $\Delta_p$ -EMOA is presented in Section 5. The performance of these PSA based algorithms is compared with the original EMOAs. Finally, conclusions are drawn in Section 6.

## 2 PSA – Part and Select Algorithm

The Part and Select Algorithm (PSA) has been recently introduced in [22] as an algorithm for selecting  $m$  well-spread points from a set of  $n$  points. It has a low computational complexity ( $O(nmk)$ , where  $k$  is the dimensionality of the points), and can be used for many applications. The procedure has two steps: First, the set is partitioned into subsets so that similar members are grouped in the same subset. Next, a diverse subset is formed by selecting

one member from each generated subset. The following description of the algorithm is borrowed from [22].

## 2.1 Partitioning a Set

The core of the PSA is the algorithm of partitioning a given set of points in the objective space into smaller subsets. In order to partition a set into  $m$  subsets, PSA performs  $m - 1$  divisions of one single set into two subsets. At each step, the set with the greatest dissimilarity among its members is divided. This is repeated until the desired stopping criterion is met. The criterion can be either a predefined number of subsets (i.e., the value of  $m$ ) or a maximal dissimilarity among each of the subsets. The dissimilarity of a set  $A$  is defined by the measure  $\varnothing A$  as follows:

Let  $A := \{\mathbf{f}_1 = [f_{11}, \dots, f_{1k}]^T, \dots, \mathbf{f}_n = [f_{n1}, \dots, f_{nk}]^T\} \subset \mathbb{R}^k$  (i.e.,  $n$  objective vectors  $\mathbf{f}_i = \mathbf{F}(\mathbf{x}_i)$  for vectors  $\mathbf{x}_i \in Q$ ), and denote

$$a_j := \min_{i=1, \dots, n} f_{ij}, \quad b_j := \max_{i=1, \dots, n} f_{ij}, \quad \Delta_j := b_j - a_j, \quad j = 1, \dots, k \quad (2)$$

$$\varnothing A := \max_{j=1, \dots, k} \Delta_j \quad (3)$$

In fact,  $\varnothing A$  is the diameter of the set  $A$  in the Chebyshev metric. The size of  $\varnothing A$  is a measure of the dissimilarity among the members of  $A$ , with a large  $\varnothing A$  indicating a large dissimilarity among the members of  $A$ .

The pseudocode of PSA for a fixed value of  $m$  is shown in Algorithm 1. At every iteration the algorithm finds the subset with the largest diameter, and parts it into two subsets.

---

### Algorithm 1 Partitioning a set $A$ into $m$ subsets

---

```

 $A_1 \leftarrow A$ 
Evaluate  $\varnothing A_1$  according to Eq. (3) and store  $\varnothing A_1$  in an archive.
 $i \leftarrow 2$ 
while  $i < m$  do
    Find  $A_j$  and coordinate  $p_j$  such that  $\varnothing A_j = \Delta_{p_j} = \max_{l=1, \dots, i-1} \varnothing A_l$ 
    Part  $A_j$  to subsets  $A_{j1}, A_{j2}$ :
     $A_{j1} \leftarrow \left\{ \mathbf{f} = [f_1, \dots, f_{p_j}, \dots, f_k]^T \in A_j, f_{p_j} \leq a_{p_j} + \varnothing A_j / 2 \right\}$ 
     $A_{j2} \leftarrow \left\{ \mathbf{f} = [f_1, \dots, f_{p_j}, \dots, f_k]^T \in A_j, f_{p_j} > a_{p_j} + \varnothing A_j / 2 \right\}$ 
    Evaluate  $\varnothing A_{j1}$  and  $\varnothing A_{j2}$  according to Eq. (3), and replace in the archive  $\varnothing A_j$  and
     $p_j$  with the pairs  $\varnothing A_{j1}, \varnothing A_{j2}$  and  $p_{j1}, p_{j2}$  accordingly.
     $S \leftarrow \{A_1, \dots, A_{j1}, A_{j2}, \dots, A_i\}$ 
     $i \leftarrow i + 1$ 

```

---

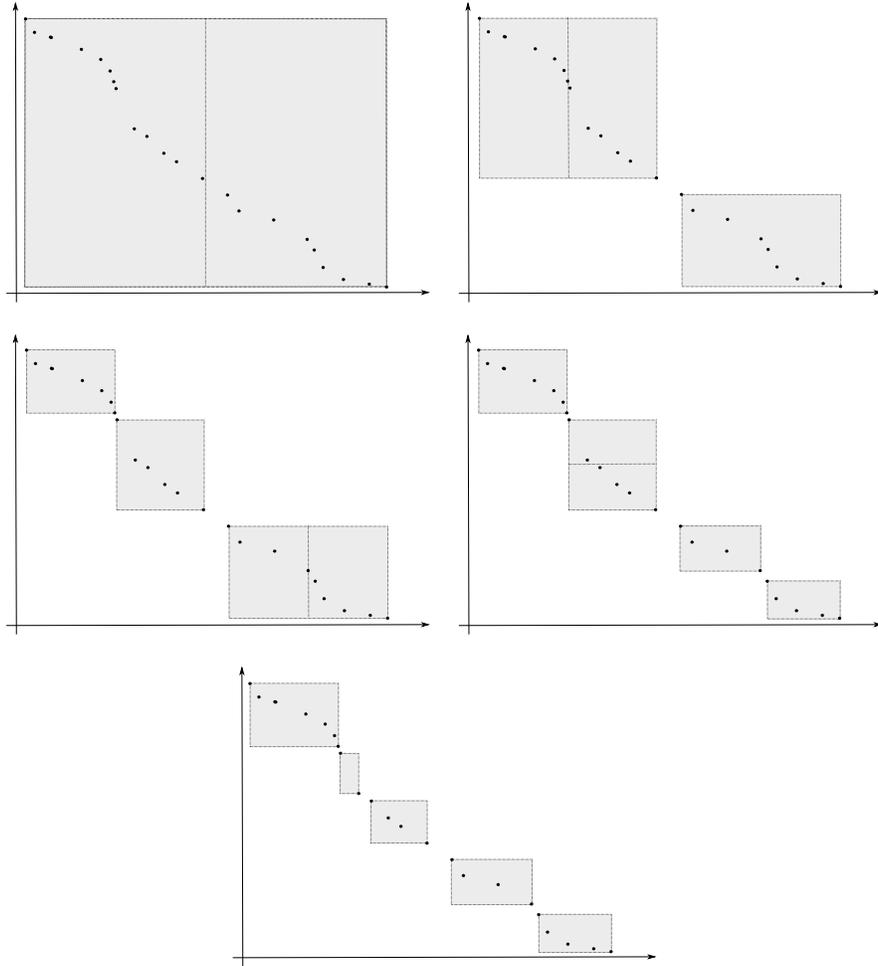
Figure 1 demonstrates the steps of the algorithm and highlights the results obtained by its use. Consider the set of 24 points in the bi-objective space depicted in the top left panel of Figure 1. Suppose that the purpose is to partition this set into  $m = 5$  subsets. The gray rectangle represents the region in the objective space that contains the solutions of the set. According to Eq. (3), the diameter of the given set is the length of the horizontal side of the rectangle. Therefore, the first partition is made by vertical incision (indicated by the vertical line in the middle of the rectangle). The results of this partition are depicted in the top right panel of Figure 1. The left subset in this panel has the greatest diameter (in horizontal direction). Therefore, the next partition is made on this subset by vertical incision. The results of this partition are depicted in the middle left panel of Figure 1. The other two panels of Figure 1 depict the results of the next two iterations of Algorithm 1.

Note that the results of the partitioning are different than the results of using a common grid in the original space. With a common grid, an initial interval in every dimension is divided into equal sections, resulting in the division of the hyperbox into smaller hyperboxes of equal space. Since the original set  $A$  does not necessarily ‘cover’ the entire space, each hyperbox in the grid might or might not contain a member of  $A$ . Hence, there is no way to predict which resulting grid will have the desired number of occupied boxes. In addition, there are certain limitations on the number of hyperboxes in the grid. For example, in a two-dimensional grid it is possible to create  $m = \{1, 2, 4, 6, 9, 12, \dots\}$  boxes, while only a number of  $m = n^2$ , when  $n$  is a positive integer, will produce an even grid. With PSA, only the occupied space (marked as the gray rectangles in Figure 1) is considered. When a set  $A_i$  is partitioned into two subsets  $A_{i_1}$  and  $A_{i_2}$ , the space considered from now on is given only by the two hyperboxes circumscribing  $A_{i_1}$  and  $A_{i_2}$ . The rest of the space in  $A_i$  is discarded. Every partition increases the number of subsets by one, and therefore any desired number of subsets can be created.

## 2.2 Selection of a Representative Subset

Once the set  $A$  has been partitioned into the  $m$  subsets  $A_1, \dots, A_m$ , the ‘most suitable’ element from each subset must then be chosen in order to obtain a subset  $A_{(r)}$  of  $A$  that contains  $m$  elements. This is of course problem dependent. Since this study aims for high diversity of the chosen elements, the following heuristic is suggested (denoted as center point selection): From each set  $A_i$  choose the member which is closest (in Euclidean metric) to the center of the hyperrectangle circumscribing  $A_i$ . If there exist more than one member closest to the center, one of them is chosen randomly.

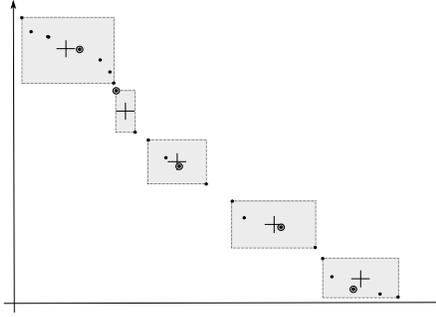
Figure 2 illustrates this rule. The original set of 24 elements (compare to Figure 1) was partitioned by Algorithm 1 into five subsets. The centres of the grey rectangles are marked with a cross. In each subset the member closest



**Fig. 1** Partitioning of 24 elements in bi-objective space into  $m = 5$  subsets (indicated by the gray boxes). (borrowed from [22])

to the center is circled (a random member is circled in the subset with only two members). The representative set  $A_{(r)} = \{a_1, a_2, a_3, a_4, a_5\}$  is the set of all circled points.

Figure 3 illustrates the performance of PSA in selecting a subset from a randomly chosen (non-dominated) population in a three-objective space. A set of 500 randomly distributed points is depicted in Figure 3(a). The set is partitioned into 40 subsets, and the central member of each subset is selected as a representative point to form the representative subset depicted in Figure 3(f). According to Eq. (3), the diameter of the given set is the distance over  $f_2$ . Therefore, the first partition is made over  $f_2$ . At the second partition,



**Fig. 2** Selection of a representative subset  $A_{(r)}$  out of  $A$  using center point selection (borrowed from [22])

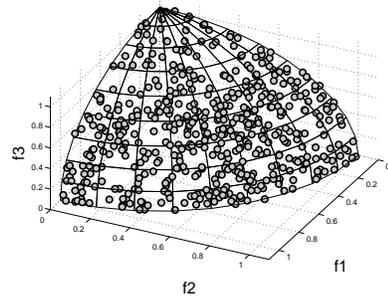
the subset of the circles from Figure 3(b) has the largest dissimilarity and therefore is partitioned (over  $f_1$ ). At the next partition the subset of gray stars is partitioned over  $f_1$  to form the four subsets shown in Figure 3(d). The final stage of Algorithm 1 is shown in Figure 3(e). The subset shown in Figure 3(f) is obtained by selecting the point closest to the center of each of the 40 subsets. Figure 3(a) clearly shows that the distribution of the points in the original set is not uniform. Nevertheless, PSA managed to select a subset of fairly evenly distributed points from it.

### 2.3 NSGA-II-PSA

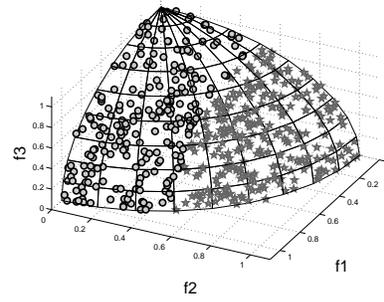
NSGA-II-PSA was introduced in [22] as an improvement of the well-known NSGA-II [13] by a straightforward integration of the PSA into it. The algorithm differs from its base EMOA in the selection of the elite population, and in the crowding measure assignment; both of which are conducted by using the PSA. The approximated sets obtained by NSGA-II-PSA were better than those obtained by NSGA-II in terms of both spread and convergence. Figure 4 depicts some of the comparative results between NSGA-II and NSGA-II-PSA, conducted in [22].

## 3 A New EMOA with PSA as a Diversity Preservation Operator

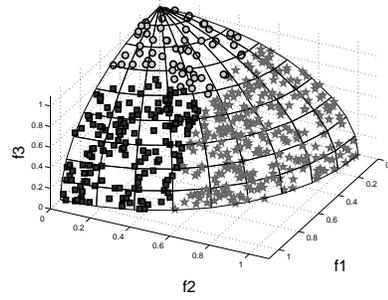
In this section, a new EMOA is suggested - Diversity Preservation Genetic Algorithm (DPGA) - that aims simultaneously for proximity and diversity. It is designed for MOPs that pose a special challenge to spread the candi-



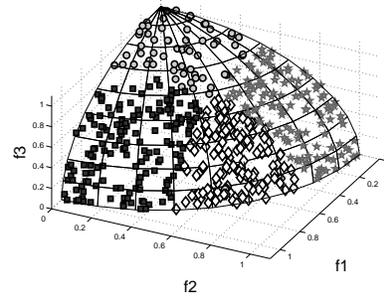
(a) original set of randomly distributed points



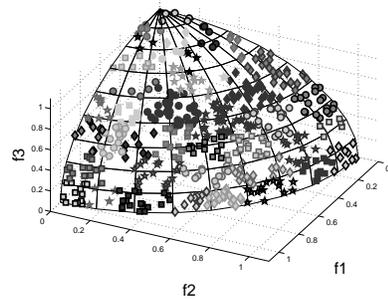
(b) two sets after first partition



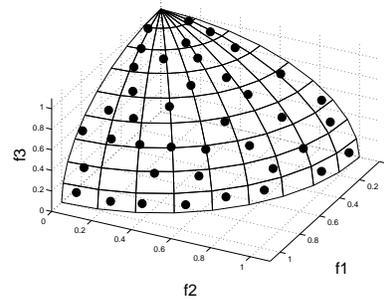
(c) three sets



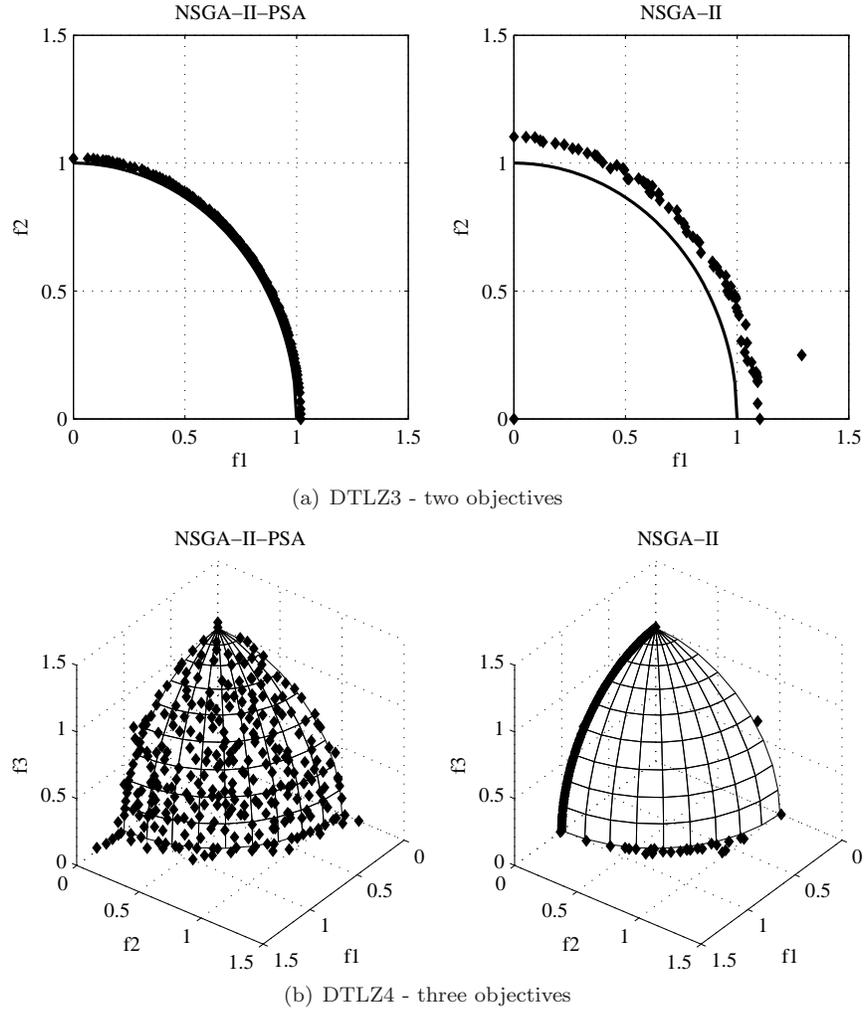
(d) four sets



(e) final partition - 40 sets

(f) the selected subset  $A_{(r)}$  containing the centers of the 40 sets

**Fig. 3** Demonstration of PSA in a three-dimensional space: Selection of a representative subset of 40 points from a randomly distributed set of 500 points. (borrowed from [22])



**Fig. 4** The final approximated set obtained by NSGA-II-PSA and NSGA-II after 25,000 function evaluations for two objectives and 75,000 for three objectives. (borrowed from [22])

date solutions along the Pareto front. The basic structure of DPGA is similar to the structure of NSGA-II-PSA [22]. NSGA-II-PSA, as most EMOAs, inherently favors proximity over diversity. The reason for that is the selection mechanism, that selects according to non-dominance, and diversity is related as a second goal. In order to overcome this property, the selection in DPGA is conducted with two parallel goals; some solutions are selected according to their rank of non-dominance, while some solutions are selected by their remoteness from other solutions in the objective space.

In DPGA, two parent populations  $P_{t+1}^P$  and  $P_{t+1}^D$  are selected from the current population  $R_t$ .  $P_{t+1}^P$  is selected according to proximity, while  $P_{t+1}^D$  is selected according to diversity. These two populations form the new parent population:  $P_{t+1} = P_{t+1}^P \cup P_{t+1}^D$ . The proportion between the sizes of the two sets is controlled by the proximity factor  $\alpha$  in the following manner:  $|P_{t+1}^P| = \alpha N$ ,  $|P_{t+1}^D| = (1 - \alpha)N$ , where  $N = |P_{t+1}|$ . The tournament selection for each population is also conducted according to its aim: Members from  $P_{t+1}^P$  are compared, as in NSGA-II-PSA, according to proximity and secondly, as a tiebreaker, according to diversity. Members from  $P_{t+1}^D$  are compared according to diversity, and secondly according to proximity. After selection, the members of both sets are combined, and crossover and mutation are applied to form the next offspring population  $Q_{t+1}$ . This procedure might produce offspring that are better both in proximity and in diversity.

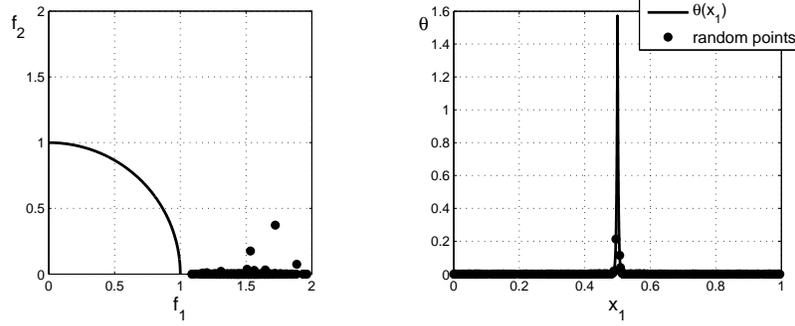
By selecting according to remoteness, a highly dominated solution can be graded with a high fitness. This approach is not intuitive, and indeed, there are no methods known to the authors that give high priority to dominated solutions. Therefore, a justification of that novel approach is given here through an example. Consider the following MOP, which is a slight variation<sup>1</sup> of DTLZ4 for two objectives [24]:

$$\begin{aligned}
 &\text{Minimize } f_1(\mathbf{x}) = r(\mathbf{x}) \cos(\theta(\mathbf{x})) \\
 &\text{Minimize } f_2(\mathbf{x}) = r(\mathbf{x}) \sin(\theta(\mathbf{x})) \\
 &\text{where } \mathbf{x} = [x_1, \dots, x_7]^T, \quad 0 \leq x_i \leq 1 \\
 &\quad \theta(\mathbf{x}) = \frac{\pi}{2} (1 - 2|x_1 - 0.5|)^{100} \\
 &\quad r(\mathbf{x}) = 1 + \sum_{i=2}^7 (x_i - 0.5)^2
 \end{aligned} \tag{4}$$

Proximity to the true Pareto front is defined by the value of  $r(\mathbf{x})$ , and the location along the Pareto front is defined by the value of  $\theta(\mathbf{x})$ . The Pareto optimal set corresponds to  $r = 1$ , i.e.,  $x_i = 0.5$  for all  $i = 2, \dots, 7$ , and to all the values of  $\theta$  between 0 and  $\pi/2$ . The mapping from  $x_1$  to  $\theta$ , as depicted in Figure 5(b), results in  $\theta$  values close to zero for 98% of the  $x_1$  values. All other values of  $\theta$  correspond to  $0.49 < x_1 < 0.51$ . Figure 5 depicts a random population of 500 solutions. Only two solutions of this population have  $\theta$  value greater than 0.1 radian. Both of them are dominated by most of the other solutions. An algorithm that favours non-dominated solutions will skip these two solutions, and their genetic information (i.e.,  $x_1$  close to 0.5), which is important to spread the approximated set along the Pareto front, will be lost.

---

<sup>1</sup> The difference of the problem in Eq. (4) from DTLZ4 is that the peak of  $\theta(x_1)$  is at  $x_1 = 0.5$  rather than at  $x_1 = 1$ . It moves the area of interest away from the limits of the design space, which is more likely to be sampled by many EAs, including NSGA-II-PSA and DPGA.



(a) Objective values of random 500 solutions. The real Pareto front is marked with a bold line.

(b) The mapping from  $x_1$  to  $\theta$

**Fig. 5** A random initial population of 500 solutions for the MOP of Eq. 4.

The algorithm of DPGA is presented in Algorithm 2. A discussion about the setting of  $\alpha$  (Step 4) appears in Section 3.3. Steps 5–7 are explained in Section 3.1. Steps 8–9 are explained in Section 3.2.

---

**Algorithm 2** DPGA - Diversity Preservation Genetic Algorithm

---

- 1:  $R_1 \leftarrow$  Generate a random set of solutions of size  $2N$
  - 2:  $t \leftarrow 1$
  - 3: **while** Stopping criteria not met **do**
  - 4:   Set  $\alpha$
  - 5:    $P_{t+1}^P \leftarrow$  Preserve  $\alpha N$  solutions from  $R_t$  based on non-dominance
  - 6:    $R_t^* \leftarrow R_t \setminus P_{t+1}^P$
  - 7:    $P_{t+1}^D \leftarrow$  Preserve  $(1 - \alpha)N$  solutions from  $R_t^*$  based on diversity.
  - 8:    $Q_{t+1}^P \leftarrow S_P(P_{t+1}^P)$
  - 9:    $Q_{t+1}^D \leftarrow S_D(P_{t+1}^D)$
  - 10:    $Q_{t+1}^* \leftarrow Q_{t+1}^P \cup Q_{t+1}^D$
  - 11:    $Q_{t+1}^{**} \leftarrow \text{Crossover}(Q_{t+1}^*)$
  - 12:    $Q_{t+1} \leftarrow \text{Mutation}(Q_{t+1}^{**})$
  - 13:    $R_{t+1} \leftarrow P_{t+1}^P \cup P_{t+1}^D \cup Q_{t+1}$
  - 14:    $t \leftarrow t + 1$
- 

### 3.1 Elite Preservation in DPGA

At each generation DPGA preserves  $N$  members in the elite (parent) population  $P_{t+1}$ , from the current population  $R_t$  of size  $2N$ . This is done in two stages: First,  $\alpha N$  members are selected from  $R_t$  to form  $P_{t+1}^P$  according to the

elite preservation procedure of NSGA-II-PSA [22]. Next,  $(1 - \alpha)N$  members are selected from the remaining members in  $R_t$  to form  $P_{t+1}^D$ . This second selection is done by partitioning the remaining members of  $R_t$  to  $1 - \alpha$  subsets using the PSA, and including one member of each subset in  $P_{t+1}^D$ . During the elite preservation stage every member  $i$  in  $P_{t+1}$  is given a proximity measure  $i_{rank}$  and a diversity measure  $i_{diversity}$ . The criteria for these measures are different for the members of  $P_{t+1}^P$  and  $P_{t+1}^D$ . The exact procedure of the elite preservation and the fitness assignment is described in Algorithm 3. The procedure is illustrated in Figure 6.

---

**Algorithm 3** Elite Preservation in DPGA
 

---

```

 $P_{t+1}^P \leftarrow$  Preserve  $\alpha N$  solutions from  $R_t$  according to NSGA-II-PSA
assign proximity and diversity measures to the solutions in  $P_{t+1}^P$  according to
NSGA-II-PSA.
 $R_t^* \leftarrow R_t \setminus P_{t+1}^P$ 
Partition  $R_t^*$  with PSA to  $(1 - \alpha)N$  subsets  $\mathcal{D} = \{D_1, \dots, D_{(1-\alpha)N}\}$ 
 $P_{t+1}^D \leftarrow \emptyset$ 
for each  $D_i \in \mathcal{D}$  do
   $D_{i,nd}$  = nondominated solutions of  $D_i$ 
   $\mathbf{d}_i \leftarrow$  center point selection from  $D_{i,nd}$ 
  Assign a diversity measure to  $\mathbf{d}_i$  equal to  $|D_{i,nd}|$ 
   $P_{t+1}^D \leftarrow \{P_{t+1}^D, \mathbf{d}_i\}$ 
Sort  $P_{t+1}^D$  to ranks of non-dominance, and assign a proximity measure to each member
according to its rank
 $P_{t+1} \leftarrow P_{t+1}^P \cup P_{t+1}^D$ 

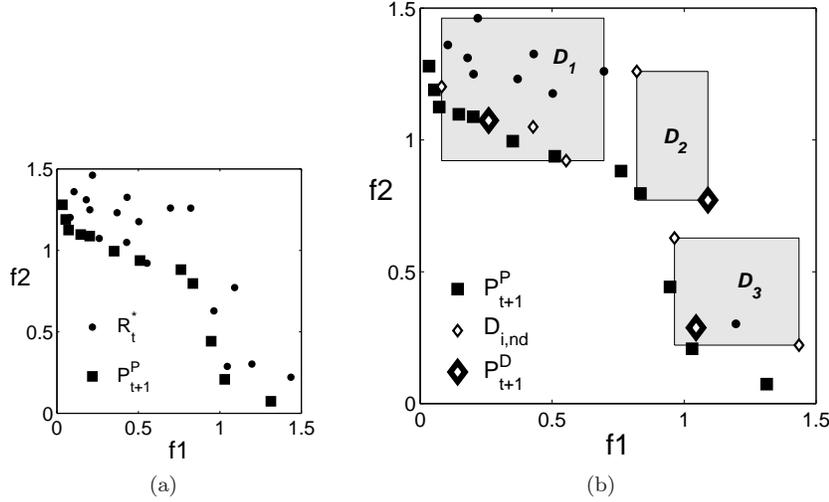
```

---

### 3.2 Selection in DPGA

As NSGA-II, DPGA also uses a binary tournament selection from  $P_{t+1}$  to form the children population  $Q_{t+1}$ . The comparison between two candidate parents is done according to the proximity and diversity measures assigned to each member in  $P_{t+1}$ . The difference from NSGA-II is that two tournaments are done in parallel; one for the population of  $P_{t+1}^P$ , and another for  $P_{t+1}^D$ .

The diversity oriented selection operator  $S_D$ , applied on  $P_{t+1}^D$ , is described in Algorithm 4. The proximity oriented selection operator  $S_P$ , which is in fact the crowded comparison operator  $\prec_n$  of NSGA-II, is the same as  $S_D$ , except for Condition 4, that in the case of  $S_P$  gives the first priority to the rank of non-dominance and the second to diversity.



**Fig. 6 Elite selection in DPGA.** The 12 members preserved for proximity from the 30 members of the previous generation’s population are marked with squares in (a). Preservation of 3 additional members for diversity is demonstrated in (b). The remaining members are divided into 3 sets  $D_1$ ,  $D_2$  and  $D_3$ . The non dominated members of each set  $D_{i,nd}$  are marked with small diamonds. The central members of each  $D_{i,nd}$ , marked with large diamonds, are preserved in  $P_{t+1}^D$ .

---

**Algorithm 4**  $S_D$  - Diversity Oriented Selection Operator

---

```

1:  $Q_{t+1}^D \leftarrow \emptyset$ 
2: for  $k = 1$  to  $(1 - \alpha)N$  do
3:   Randomly select members  $i$  and  $j$  from  $P_{t+1}^D$ 
4:   if  $(i_{diversity} < j_{diversity})$  or  $((i_{diversity} = j_{diversity})$  and  $(i_{rank} < j_{rank}))$  then
5:      $Q_{t+1}^D \leftarrow Q_{t+1}^D \cup \{i\}$ 
6:   else
7:      $Q_{t+1}^D \leftarrow Q_{t+1}^D \cup \{j\}$ 

```

---

### 3.3 Sensitivity to Parameters

The performance of DPGA is highly affected by the proximity factor  $\alpha$ . Setting  $\alpha$  too low will hold back the algorithm from converging towards the Pareto front, since the computational power is wasted on too many dominated solutions. On the other hand, a too high value of  $\alpha$  may lead to premature convergence, and to loss of important genetic information that may lead to undiscovered non-dominated regions. There is a stage in the evolutionary progress, when it does not make sense anymore to maintain dominated solu-

tions, since they lack the time to reach the first front. Hence, the value of  $\alpha$  should not be fixed for the entire run of the algorithm.

One possible way for the setting of  $\alpha$  is suggested here. In this heuristic, DPGA consists of two stages; at the first stage a constant value of  $\alpha \in (0, 1)$  is set; at the second stage the selection is done as in NSGA-II-PSA (it can be conducted by simply set  $\alpha$  to one). This heuristic requires two a-priori decisions – the value of  $\alpha$  at the first stage, and when to switch from the first to the second stage. The second decision can be described through a parameter  $\mu$  – the portion of the generations in which the selection is done according to DPGA. The proper values of  $\alpha$  and  $\mu$  are problem dependent, and it is out of the authors' ability at the moment to suggest a generic way to determine them. An analysis of the performance of DPGA for one benchmark, with different values of these parameters, is given in Section 4. The conclusions on the parameters setting for this benchmark can be implemented as a starting point for other problems.

Other heuristics, such as a gradual increase of  $\alpha$ , or setting  $\alpha$  as a function of the generation count, can lead to better performance, but may be associated with more parameters. Probably, the proper way is to change  $\alpha$  according to the progress of the global search. Meaning, to decrease it when the elite population loses its diversity, and to increase it otherwise. This should be done automatically within the evolutionary algorithm.

## 4 Simulations for DPGA

In this section, the proposed DPGA is evaluated and the sensitivity of the parameters  $\alpha$  and  $\mu$  is studied. The algorithm is analyzed on the DTLZ4 benchmark with 3 objectives. This benchmark is used, since it poses a special challenge in spreading the approximated set. This is exactly the kind of problems the DPGA should be used for. The conclusions on the parameters setting for this problem can be implemented as a starting point for other problems. The approximated sets are evaluated by the hypervolume measure (HV) [10].

First, the algorithm is tested for different values of  $\alpha$  and  $\mu$ . The values of  $\alpha = \{0, 0.15, 0.3, 0.45, 0.6, 0.75, 0.9, 1\}$  and  $\mu = \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$  were examined for all possible combinations. Fifty independent runs were carried out for each setting. For the sake of proper comparison, all combinations of parameter setting ran on the same fifty initial populations. The parameter setting of  $\alpha = 1$  or  $\mu = 0$  is the NSGA-II-PSA algorithm without the modifications of DPGA. Therefore, these settings are evaluated only once on the test set, and the corresponding results are referred to as "NSGA-II-PSA".

According to the results of this analysis, another comparison is made to check the ability of DPGA to handle a poor initial population. DPGA with the best combination of  $\alpha$  and  $\mu$ , is compared with NSGA-II-PSA as a ref-

erence in this test. Each algorithm solves the problem for 100 times with the same initial population which caused in the worst performances in the previous simulations.

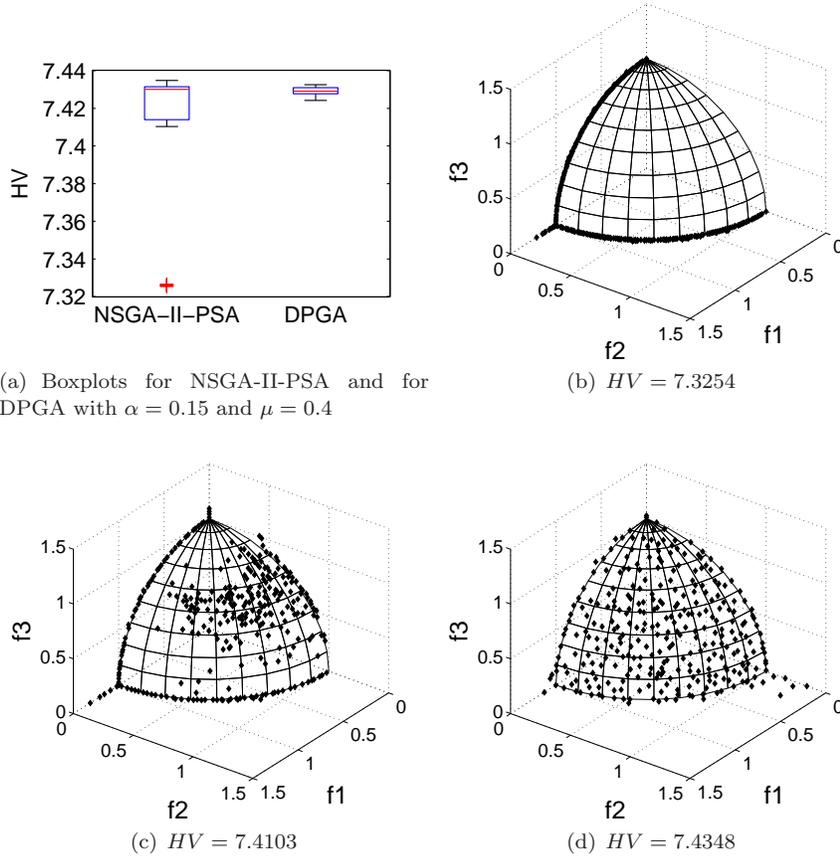
#### 4.1 *Experimental Setup*

Both algorithms are given real-valued decision variables. They use the simulated binary crossover (SBX) operator and polynomial mutation [25], with distribution indices of  $\eta_c = 20$  and  $\eta_m = 20$  respectively. A crossover probability of  $p_c = 1$  and a mutation probability of  $p_m = 1/3$  are used. The population size is set to 300, and the number of generations to 250.

#### 4.2 *Results of DPGA with Various Parameter Settings*

The HV values of the final results in all the tests varied between 7.325 and 7.435. Approximated sets with values larger than 7.4 include at least some solutions on the surface of the sphere of the Pareto front. Sets with lower HV values consist of solutions on the  $f_1 - f_2$  plane and  $f_1 - f_3$  plane only. Results of that kind are considered as a failure of the algorithm to spread the approximated set along the Pareto front. Figure 7 depicts a boxplot of the statistic results of the NSGA-II-PSA ( $\alpha = 1, \mu = 0$ ) and one parameter setting ( $\alpha = 0.15, \mu = 0.4$ ), as well as three approximated fronts and their HV values. The results in Figure 7(b) are considered as a failure. Those in Figure 7(c) are quite poor, and the results in Figure 7(d) are considered as good results. The boxplot of NSGA-II-PSA in Figure 7(a) shows the failures of the algorithm as outliers. The boxplots in Figure 7(a) indicate that there is no statistically significant difference in location of the HV values of NSGA-II-PSA and DPGA. On the other hand, DPGA with the above parameter setting is much more consistent regarding to different initial populations, and has no failures in spreading the approximated front. NSGA-II-PSA has 11 failures out of 50.

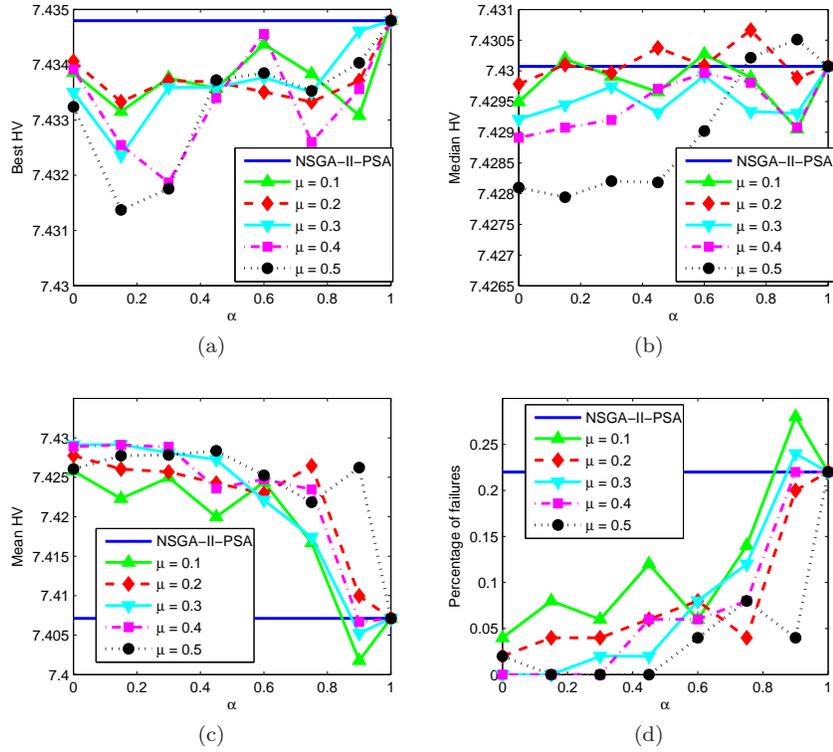
The results of the statistic evaluation of all the combinations of  $\alpha$  and  $\mu$  values are depicted in Figure 8. Four statistical qualities are concerned here: Figure 8(a) depicts the best HV of 50 tests; Figure 8(b) depicts the median HV; Figure 8(c) depicts the mean HV; and Figure 8(d) depicts the percentage of failures. Note that all the values of  $\mu$  converge to the same point when  $\alpha = 1$ , since the results do not depend on  $\mu$  in that case, and the algorithm is simply NSGA-II-PSA. The same statement holds for  $\mu = 0$  as it is the same for all values of  $\alpha$  (the blue line labeled "NSGA-II-PSA"). Three clear observations can be made from the results shown in Figure 8: (a) the best results from 50 trials are obtained with NSGA-II-PSA; (b) DPGA



**Fig. 7** HV values of 50 tests for NSGA-II-PSA and DPGA, and examples for the HV measure associated with three approximated Pareto fronts. HV values that are less than 7.4 are considered as a failure to spread the approximated set along the Pareto front (e.g., the outliers of NSGA-II-PSA, marked as red crosses, and the results in Figure 7(b)).

reduces the chance of a failure for most parameter settings (especially for  $\alpha < 0.5$  and  $\mu \geq 0.2$ ); (c) for this benchmark, the mean performance is more affected from the number of failures, and therefore, DPGA has a better mean HV than NSGA-II-PSA for most of the parameter settings.

These results corroborate the hypothesis that dominated solutions might contain crucial information, and the preservation of some diversified dominated solutions at the beginning of the evolutionary process can prevent premature convergence. It is worth reminding that NSGA-II-PSA is already an improvement of NSGA-II, and by recalling Figure 6(b) in [22], all of the approximated sets found with NSGA-II for DTLZ4 with 3 objectives have HV lower than 7.4.



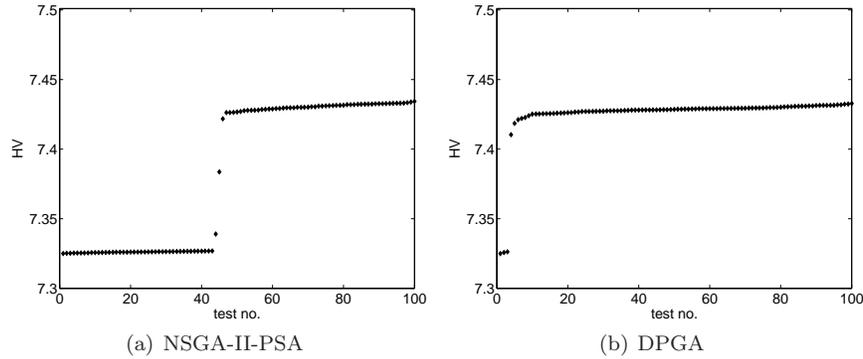
**Fig. 8** Statistic results of 50 tests for NSGA-II-PSA and for DPGA with different values of  $\alpha$  and  $\mu$ .

To choose the best parameter setting of DPGA for the DTLZ4 benchmark according to these results, the main objective should be the reduction of failures. In general, the percentage of failures decreases with the increase of  $\mu$  and the decrease of  $\alpha$ . Both  $\mu = 0.4$  and  $\mu = 0.5$  satisfy this demand. Due to the inevitable tradeoff between proximity and diversity, the performance should be considered as well, reflected by the mean, median and best HV. Considering all the above, the best parameter setting for this benchmark is  $\alpha = 0.15, \mu = 0.4$ . It had no failures, and has the best performance over all the other settings with no failures.

### 4.3 Poor Initial Population

Here, a comparison between NSGA-II-PSA and DPGA is conducted in order to examine the ability of the algorithms to handle a very poor popu-

lation. The initial population which produced the largest amount of failures in Section 4.2 was used as a benchmark. In this simulation, the worst initial population is given as an input to DPGA with the best combination of  $\alpha$  and  $\mu$ , and to NSGA-II-PSA, and is solved by each algorithm for 100 times.



**Fig. 9** Results of 100 tests with a poor initial population. For clarity, the results are sorted according to performance.

The HV values of the obtained approximated fronts are depicted in Figure 9. The advantage of DPGA over NSGA-II-PSA is clear. While NSGA-II-PSA has failed 45 times in finding solutions on the surface of the sphere, DPGA has only failed 3 times. The HV of the successful results are quite the same for both algorithms.

## 5 Using PSA for Hausdorff Approximations of the Pareto Front

In this section, a first attempt is made to show that PSA can be used successfully within EMOAs to compute Hausdorff approximations of the Pareto front. The Hausdorff distance  $d_H$  (e.g., [26]) prefers, roughly speaking, approximations  $A \subset \mathbb{R}^n$  such that its images are located equally spaced along the Pareto front. Hence,  $d_H$  can be viewed as a performance indicator that is closely related to the terms *spread* and *convergence* as used in the EMO community. PSA is integrated into NSGAI-I [27] to produce two new algorithms: NSGA-II-I-PSA and NSGAI-I- $\Delta_p$ -PSA. Both algorithms use an external archive in addition to the procedures of NSGAI-I. In all the tests conducted in this study NSGA-II-I-PSA achieved better Hausdorff approxi-

mations<sup>2</sup> than its base EMOA, and NSGAI-I- $\Delta_p$ -PSA improved the performance in most cases. On models where the Pareto front is connected, both of the new methods cannot compete with  $\Delta_p$ -EMOA [28], which is a specialized algorithm to produce good Hausdorff approximations. NSGA-II-I-PSA, however, is advantageous in cases where the Pareto front is disconnected. We conjecture that this is the merit of PSA that is independent from the geometry of the underlying model. First results for bi-objective problems (i.e.,  $k = 2$ ) are presented here, and considerations of  $k > 2$  and further improvements of the hybrid are kept for future research.

The performance indicator considered in this section,  $\Delta_p$ , is defined as follows.

**Definition 1 (averaged Hausdorff distance  $\Delta_p$  [11]).** Let  $p \in \mathbb{N}$ ,  $A = \{a_1, \dots, a_r\} \subset \mathbb{R}^d$  be a candidate set and  $Y = \{y_1, \dots, y_r\} \subset \mathbb{R}^k$  be its image, i.e.,  $y_i = F(a_i)$ ,  $i = 1, \dots, r$ . Further, let  $P := \{p_1, \dots, p_m\} \subset \mathbb{R}^k$  be a discretization of the Pareto front. Then it is

$$\Delta_p(Y, P) = \max \left( \left( \frac{1}{r} \sum_{i=1}^r \text{dist}(y_i, P)^p \right)^{1/p}, \left( \frac{1}{m} \sum_{i=1}^m \text{dist}(p_i, Y)^p \right)^{1/p} \right), \quad (5)$$

where  $\text{dist}(x, B) := \inf_{b \in B} \|x - b\|$  denotes the distance between a point  $x$  and a set  $B$ .

$\Delta_p$  is a combination of slight variations of the well-known *Generational Distance* (GD, see [29]) and the *Inverted Generational Distance* (IGD, see [30]). For  $p = \infty$  the indicator coincides with the Hausdorff distance (i.e.,  $\Delta_\infty = d_H$ ), and hence,  $\Delta_p$  can be viewed as an averaged Hausdorff distance.

The NSGA-II-I is a variant of the classical NSGA-II and is based on the conjecture that a sequential update of the crowding distances leads to a more homogeneous distribution of the population than the single determination of the crowding distances of the original NSGA-II. This algorithm is used here as a base EMOA for the new algorithms, that include an additional external archive strategy as indicated in the Figure 10. PSA is being used here for the update of the archive in two variants: (i) it is used as a tool to select the best individuals to be stored in the external archive (NSGA-II-I-PSA), and (ii) PSA is integrated into the procedure of  $\Delta_p$ -EMOA [28] that selects the best individuals to the external archive according to an approximated reference set (NSGAI-I- $\Delta_p$ -PSA). Here, PSA is used as a tool to obtain the reference set required to compute the distance to the set of interest. The procedure of the external archive strategy using PSA as the tool to select the best individuals in each generation (for NSGA-II-I-PSA) is detailed in Algorithm 6. The procedure where PSA is used to generate the reference set that  $\Delta_p$

<sup>2</sup> In fact, we will use the *averaged* Hausdorff distance in order to avoid punishments of single outliers that can occur when using stochastic search methods such as evolutionary algorithms [11].

needs to be computed (NSGAI-I- $\Delta_p$ -PSA) is given in Algorithm 7. In this algorithm, first the set  $\mathcal{ND}$  is computed that consists of all nondominated solutions of the current population  $P_i$ , the new offspring set  $O_i$  and the current archive  $A_i$ . If the magnitude of  $\mathcal{ND}$  is greater than the size of the external archive  $N_A$ , then PSA is applied on  $\mathcal{ND}$  to obtain a reference front  $R$  of magnitude  $N_A$ . This set is further on used to update the archive  $A_i$  by  $O_i$  according to the best  $\Delta_p$  values with respect to  $R$ . Hereby,  $\Delta_p$ -Update denotes the archiver used in [28] which is given in Algorithm 5, where  $h(a)$  is the  $\Delta_p$  value of the set of solutions  $ND$  without the solution  $a$ .

---

**Algorithm 5**  $\Delta_p$ -Update
 

---

**Require:** new solution  $o_i$ , archive  $A_i$ , reference front  $R$ , archive size  $N_A$

**Ensure:** new archive  $A_{i+1}$

$\mathcal{ND}$  = nondominated solutions of  $A_i \cup o_i$

**if**  $|\mathcal{ND}| < N_A$  **then**

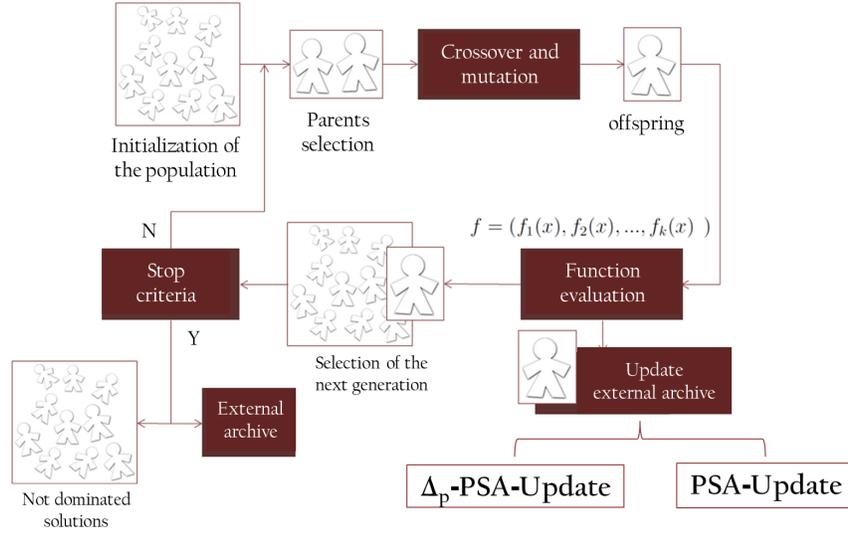
**for all**  $a \in \mathcal{ND}$  **do**

$h(a) = \Delta_p(\mathcal{ND} \setminus \{a\}, R)$

$a^* = \operatorname{argmin}\{h(a) : a \in \mathcal{ND}\}$

$A_{i+1} = \mathcal{ND} \setminus \{a^*\}$

---



**Fig. 10** General NSGAI-I procedure with an external archive. PSA-Update is being used at NSGA-II-I-PSA, and  $\Delta_p$ -PSA-Update is used for NSGAI-I- $\Delta_p$ -PSA.

**Algorithm 6** PSA-Update

---

**Require:** population  $P_i$ , offspring  $O_i$ , archive  $A_i$ , archive size  $N_A$   
**Ensure:** new archive  $A_{i+1}$   
 $\mathcal{ND}$  = nondominated solutions of  $P_i \cup O_i \cup A_i$   
**if**  $|\mathcal{ND}| < N_A$  **then**  
     $A_{i+1} = \mathcal{ND}$   
**else**  
     $A_{i+1} = \text{PSA}(\mathcal{ND}, N_A)$

---

**Algorithm 7**  $\Delta_p$ -PSA-Update

---

**Require:** population  $P_i$ , offspring  $O_i$ , archive  $A_i$ , archive size  $N_A$   
**Ensure:** new archive  $A_{i+1}$   
 $\mathcal{ND}$  = nondominated solutions of  $P_i \cup O_i \cup A_i$   
**if**  $|\mathcal{ND}| < N_A$  **then**  
     $A_{i+1} = \mathcal{ND}$   
**else**  
     $R = \text{PSA}(\mathcal{ND}, N_A)$   
     $A_{i+1} = \emptyset$   
    **for all**  $o \in O_i$  **do**  
         $A_{i+1} = \Delta_p\text{-Update}(o, A_i, R)$

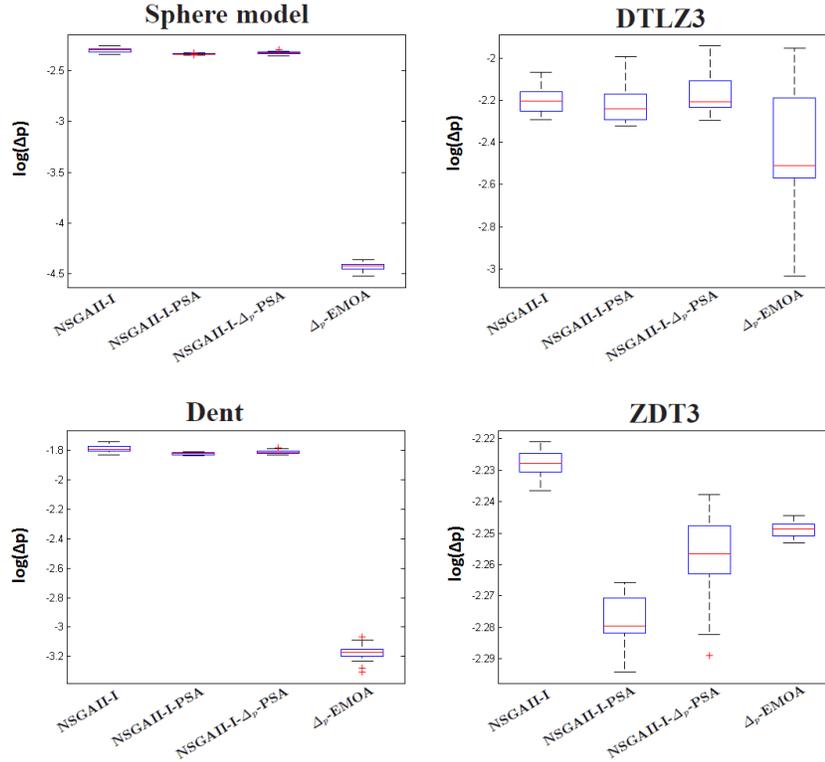
---

To test the new algorithms, they are first evaluated on four test problems with different characteristics: (i) the bi-objective sphere model [28] that has a convex Pareto front, (ii) DTLZ3 [24] that has a concave Pareto front, (iii) the Dent problem [31] that has a convex-concave front, and (iv) ZDT3 [32] where the Pareto front is disconnected. The number of decision variables and their ranges are specified as recommended in literature, for the bi-objective sphere model  $0 \leq x_i \leq 1$  ( $i = 1, 2$ ), for the DTLZ3  $0 \leq x_i \leq 1$  ( $i = 1, \dots, 10$ ), for the Dent  $-1.5 \leq x_i \leq 1.5$  ( $i = 1, 2$ ) and for the ZDT3  $0 \leq x_i \leq 1$  ( $i = 1, \dots, 20$ ). Twenty independent test runs are made, each with a budget of 50,000 function calls, a population size equal to 100 and an archive size  $N_A$  equal to 100. All algorithms have been implemented in jMetal [33]. The simulated binary crossover operator is parameterized by a component-wise probability equal to 0.9 and a distribution index equal to 20. Polynomial mutation is applied using a mutation probability equal to  $1/d$  ( $d =$  number of decision variables) and the distribution index equal to 20. Table 1 shows the obtained numerical results for the  $\Delta_p$  indicator where  $p = 1$ , and Figure 11 shows boxplots of the  $\Delta_p$  values at the final generation. The  $\Delta_p$  indicator is calculated based on fixed reference fronts referred to as benchmark fronts in the following. Ideal benchmark fronts are composed of the set of  $m$  solutions with minimum  $\Delta_p$  value with respect to the true Pareto front, where  $m$  denotes the population size of the EMOA. As the true Pareto fronts of the test problems are known in this study, these fronts are composed by  $m$  well distributed points along the true Pareto front (i.e. the set of  $m$  points with optimal PL-metric as it is defined in [28]). It can be seen that the  $\Delta_p$ -EMOA yields the best results for all models with connected Pareto front, but in the

case of the disconnected front (ZDT3) NSGAII-I-PSA obtains better values. This is also reflected in Figure 11 which shows the respective  $\Delta_p$ -values at the final EMOA generation. Additionally, statistical significance of the results is confirmed by this means, regarding the comparison to the  $\Delta_p$ -EMOA.

**Table 1** Averaged  $\Delta_1$  values for test problems with different characteristics.

	Sphere model	DTLZ3	Dent	ZDT3
NSGAII-I	0.00503875	0.00638702	0.01618773	0.00591195
NSGAII-I-PSA	0.00460146	0.00621689	0.01501212	<b>0.00527150</b>
NSGAII-I- $\Delta_p$ -PSA	0.00473097	0.00680468	0.01539346	0.00552639
$\Delta_p$ -EMOA	<b>0.00003729</b>	<b>0.00495835</b>	<b>0.00067532</b>	0.00777191



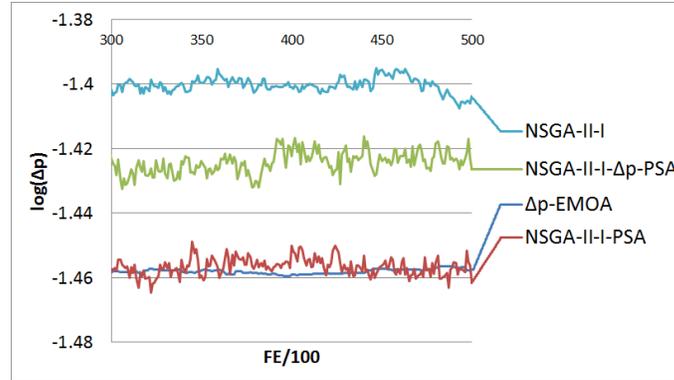
**Fig. 11** Boxplots of  $\Delta_p$ -values at final generation.

In order to investigate the behavior of the PSA based algorithms on models with disconnected fronts, a further test is made using the MOPs ([1])

Kursawe, Poloni, Schaffer, and ZDT3. The setting of the experiments is the same as for the previous ones. The number of decision variables and their ranges are as follows: For the Kursawe  $-5 \leq x_i \leq 5$  ( $i = 1, 2, 3$ ), for the Poloni  $(-1 * \pi) \leq x_i \leq \pi$  ( $i = 1, 2$ ), for the Schaffer  $-5 \leq x_i \leq 10$  ( $i = 1$ ) and for the ZDT3  $0 \leq x_i \leq 1$  ( $i = 1, \dots, 20$ ). Table 2 shows the obtained results, and Figures 12 – 15 show the median distance to the Pareto front in terms of  $\Delta_1$  on the ordinate and the number of function evaluations on the abscissa. NSGA-II-I-PSA wins the competition on all four models which is most probably due to PSA that is independent of the geometry of the problem.  $\Delta_p$ -EMOA prefers connected Pareto fronts since the reference front needed for the  $\Delta_p$  archiver is built on the assumption that the Pareto front is connected [28]. Such an assumption is not made in PSA. To take into account the stochastic nature of the EMOA and to show the performance differences are significant, Figure 16 shows boxplots of the  $\Delta_p$ -indicator at the final generation. The differences in location of the  $\Delta_p$ -values of the NSGAI-I-PSA compared to the other EMOA are statistically significant, beside for Kursawe. These results are encouraging, however, more investigations are required to obtain a better EMOA aiming for Hausdorff approximations which we leave for future work.

**Table 2** Averaged  $\Delta_1$  values for test problems with disconnected fronts.

	Kursawe	Poloni	Schaffer	ZDT3
NSGAI-I	0.03966693	0.06964843	0.02621266	0.00592310
NSGAI-I-PSA	<b>0.03470179</b>	<b>0.05784069</b>	<b>0.02189886</b>	<b>0.00515468</b>
NSGAI-I- $\Delta_p$ -PSA	0.03774589	0.06157774	0.02304279	0.00548551
$\Delta_p$ -EMOA	0.03489292	0.08614311	0.03160346	0.00778455



**Fig. 12** Median distances to the Pareto front w.r.t.  $\Delta_p$  for Kursawe problem.

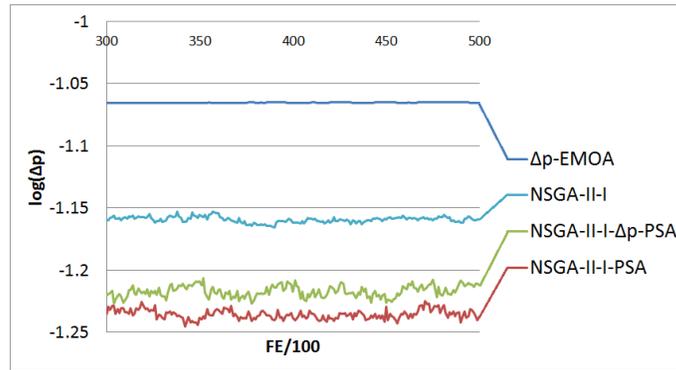


Fig. 13 Median distances to the Pareto front w.r.t.  $\Delta_p$  for Poloni problem.

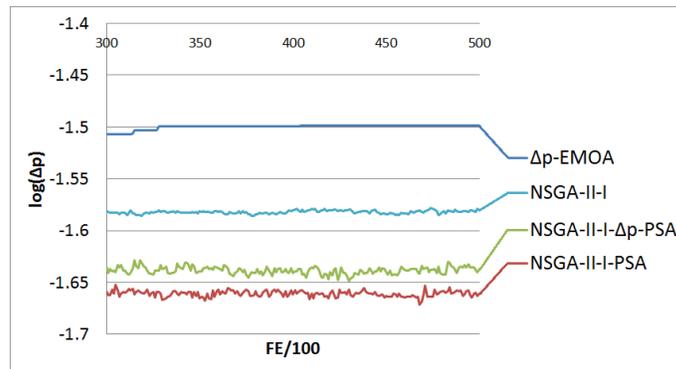


Fig. 14 Median distances to the Pareto front w.r.t.  $\Delta_p$  for Schaffer problem.

## 6 Conclusions and Future Work

In this study, the ability of the PSA (Part and Select Algorithm) as a selection mechanism within EMOAs was examined. In one part of the study, PSA was used for elite selection, and in the other it was used as an archiving tool. For both cases, the results of the PSA based algorithms were satisfactory, and they were found to have better performance than their non-PSA equivalents for certain types of optimization problems.

A new evolutionary optimization approach was presented, that preserves some dominated solutions from one generation to the next. This approach

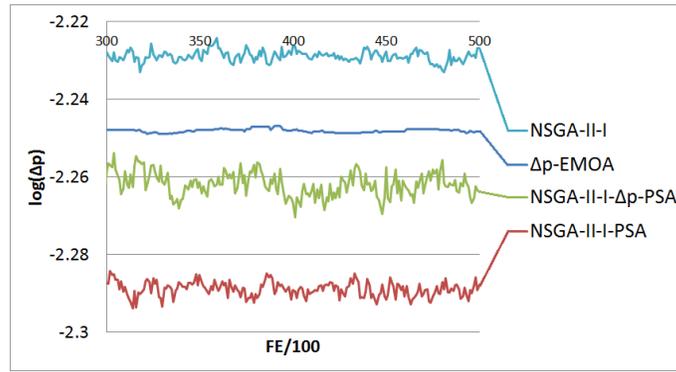


Fig. 15 Median distances to the Pareto front w.r.t.  $\Delta_p$  for ZDT3 problem.

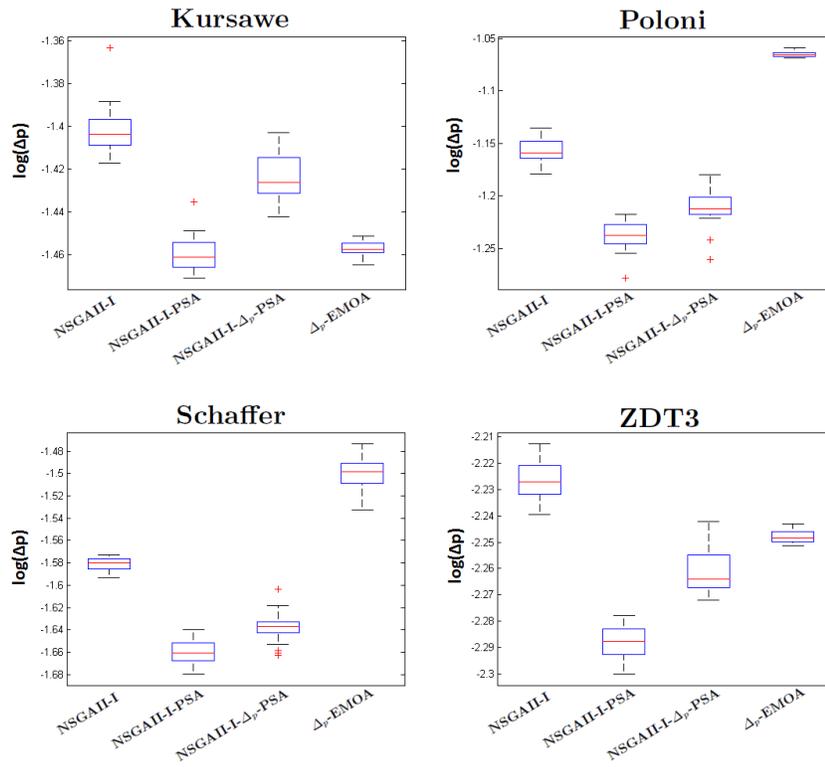


Fig. 16 Boxplots of  $\Delta_p$ -values at final generation.

was studied through a novel PSA based EMOA denoted as DPGA. The algorithm has the capacity to control the trade-off between the exploitation of proximity, to the exploitation of diversity. It was shown that by assigning high fitness to solutions that are isolated in the objective space, even if they are dominated, the chances for a failure in spreading the candidate solutions along the Pareto front decrease. As future work the performance of DPGA should be also evaluated for "regular" optimization problems that do not pose a special challenge to find a diverse set of candidate solutions. Some more comparisons with state-of-the-art EMOAs should be conducted as well. Finally, DPGA can be improved if its related parameter  $\alpha$  could be adjusted automatically. In order to do so, a measure to identify that proximity is over-exploited on the account of diversity, is required. This measure can be used during the progress of the algorithm to decide the appropriate value of  $\alpha$ .

The PSA was found to be also an appropriate archiving tool for Hausdorff approximations inspired EMOAs for special cases. The proposed algorithm NSGAI-I-PSA could not compete with the specialized algorithm for Hausdorff approximations  $\Delta_p$ -EMOA on models where the Pareto front is connected. However in cases where the Pareto front is disconnected, NSGAI-I-PSA has outperformed the  $\Delta_p$ -EMOA, producing better Hausdorff approximations to the Pareto front according to the  $\Delta_p$  indicator for the four benchmark problems selected. The advantage of NSGAI-I-PSA is thanks to that PSA is independent from the geometry of the underlying problem, so the selection of the best solutions with respect to spread and convergence is not affected by the gaps within the Pareto fronts of the problems. We conjecture that the consideration of PSA will be particularly advantageous in cases more than three objectives are under consideration. Hence, the extension of the NSGA-II-I-PSA to higher-dimensional problems seems like a promising research direction.

**Acknowledgements** This research was supported by a Marie Curie International Research Staff Exchange Scheme Fellowship within the 7th European Community Framework Programme.

Oliver Schütze acknowledges support from the CONACyT project no. 128554.

## References

1. Coello, C.A.C., Lamont, G.B., Van Veldhuizen, D.A.: Evolutionary algorithms for solving multi-objective problems. Volume 5. Springer (2007)
2. Deb, K.: Multi objective optimization using evolutionary algorithms. John Wiley and Sons (2001)
3. Bosman, P., Thierens, D.: The balance between proximity and diversity in multiobjective evolutionary algorithms. Evolutionary Computation, IEEE Transactions on **7**(2) (april 2003) 174 – 188
4. Tan, K., Chiam, S., Mamun, A., Goh, C.: Balancing exploration and exploitation with adaptive variation for evolutionary multi-objective optimization. European Journal of

- Operational Research **197**(2) (2009) 701 – 713
5. Abbass, H.: The self-adaptive Pareto differential evolution algorithm. In: Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on. Volume 1. (may 2002) 831 –836
  6. Laumanns, M., Rudolph, G., Schwefel, H.: Mutation control and convergence in evolutionary multi-object optimization. HT014601767 (2001)
  7. Bosman, P.A., Thierens, D.: Multi-objective optimization with diversity preserving mixture-based iterated density estimation evolutionary algorithms. International Journal of Approximate Reasoning **31**(3) (2002) 259 – 289
  8. Li, M., Zheng, J., Xiao, G.: An efficient multi-objective evolutionary algorithm based on minimum spanning tree. In: Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on. (june 2008) 617 –624
  9. Wineberg, M., Oppacher, F.: The underlying similarity of diversity measures used in evolutionary computation. In Cantú-Paz, E., Foster, J., Deb, K., Davis, L., Roy, R., O'Reilly, U.M., Beyer, H.G., Standish, R., Kendall, G., Wilson, S., Harman, M., Wegener, J., Dasgupta, D., Potter, M., Schultz, A., Dowsland, K., Jonoska, N., Miller, J., eds.: Genetic and Evolutionary Computation — GECCO 2003. Volume 2724 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2003) 1493–1504
  10. Zitzler, E.: Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. PhD thesis, Swiss Federal Institute of Technology Zurich (1999)
  11. Schütze, O., Esquivel, X., Lara, A., Coello, C.: Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization. Evolutionary Computation, IEEE Transactions on **16**(4) (aug. 2012) 504 –522
  12. Horn, J., Nafpliotis, N., Goldberg, D.: A niched Pareto genetic algorithm for multiobjective optimization. In: Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on. (jun 1994) 82 –87 vol.1
  13. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. Evolutionary Computation, IEEE Transactions on **6**(2) (apr 2002) 182 –197
  14. Xiaoning, S., Min, Z., Tao, L.: A multi-objective optimization evolutionary algorithm addressing diversity maintenance. In: Computational Sciences and Optimization, 2009. CSO 2009. International Joint Conference on. Volume 1. (april 2009) 524 –527
  15. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In Giannakoglou, K., et al., eds.: Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001), International Center for Numerical Methods in Engineering (CIMNE) (2002) 95–100
  16. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. Evolutionary Computation, IEEE Transactions on **11**(6) (dec. 2007) 712 –731
  17. Zuiani, F., Vasile, M.: Multi agent collaborative search based on Tchebycheff decomposition. Computational Optimization and Applications (March 2013) 1–20
  18. Batista, L., Campelo, F., Guimarães, F., Ramírez, J.: Pareto cone epsilon-dominance: Improving convergence and diversity in multiobjective evolutionary algorithms. In Takahashi, R., Deb, K., Wanner, E., Greco, S., eds.: Evolutionary Multi-Criterion Optimization. Volume 6576 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2011) 76–90
  19. Laumanns, M., Ocenasek, J.: Bayesian optimization algorithms for multi-objective optimization. In Guervós, J., Adamidis, P., Beyer, H.G., Schwefel, H.P., Fernández-Villacañas, J.L., eds.: Parallel Problem Solving from Nature — PPSN VII. Volume 2439 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2002) 298–307

20. Loridan, P.:  $\varepsilon$ -solutions in vector minimization problems. *Journal of Optimization Theory and Applications* **43** (1984) 265–276
21. Sato, H., Aguirre, H.E., Tanaka, K.: Controlling dominance area of solutions and its impact on the performance of moeas. In Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T., eds.: *Evolutionary Multi-Criterion Optimization*. Volume 4403 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg (2007) 5–20
22. Salomon, S., Avigad, G., Goldvard, A., Schütze, O.: PSA – a new scalable space partition based selection algorithm for MOEAs. In Schütze, O., ed.: *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*. Volume 175 of *Advances in Intelligent Systems and Computing*. Springer Berlin Heidelberg (2013) 137–151
23. Avigad, G., Matalon Eisenstadt, E., Salomon, S., Gadelha Guimar, F.: Evolution of contours for topology optimization. In Schütze, O., ed.: *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*. Volume 175 of *Advances in Intelligent Systems and Computing*. Springer Berlin Heidelberg (2013) 397–412
24. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable multi-objective optimization test problems. In: *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*. Volume 1. (may 2002) 825–830
25. Deb, K., Agrawal, R.: Simulated binary crossover for continuous search space. *Complex systems* **9**(2) (1995) 115–148
26. Heinonen, J.: *Lectures on Analysis on Metric Spaces*. Springer New York (2001)
27. Kukkonen, S., Deb, K.: Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems. In: *Proceedings of the 2006 IEEE Congress on Evolutionary Computation*, IEEE Press (2005) 1179–1186
28. Gerstl, K., Rudolph, G., Schütze, O., Trautmann, H.: Finding evenly spaced fronts for multiobjective control via averaging Hausdorff-measure. In: *Int'l. Proc. Conference on Electrical Engineering, Computing Science and Automatic Control (CCE 2011)*. (2011) 975–980
29. Van Veldhuizen, D.A.: *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. PhD thesis, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology
30. Coello, C.A.C., Cortés, N.C.: Solving multiobjective optimization problems using an artificial immune system. *Genetic Programming and Evolvable Machines* **6**(2) (June 2005) 163–190
31. Schütze, O., Laumanns, L., Tantar, E., Coello, C.A.C., Talbi, E.G.: Computing gap free Pareto front approximations with stochastic search algorithms. *Evolutionary Computation* **18**(1) (2010) 65–96
32. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Comput.* **8**(2) (June 2000) 173–195
33. Durillo, J.J., Nebro, A.J.: jmetal: A java framework for multi-objective optimization. *Advances in Engineering Software* **42** (2011) 760–771