A New Multi-Objective Evolutionary Algorithm Based on a Performance Assessment Indicator

Cynthia A. Rodríguez Villalobos
Computer Science Department
CINVESTAV-IPN
México, DF
cynthiadriana@gmail.com

Carlos A. Coello Coello*
Computer Science Department
CINVESTAV-IPN
México, DF
ccoello@cs.cinvestav.mx

ABSTRACT

An emerging trend in the design of multi-objective evolutionary algorithms (MOEAs) is to select individuals through the optimization of a quality assessment indicator. However, the most commonly adopted indicator in current use is the hypervolume which becomes very expensive (computationally speaking) as we increase the number of objectives. In this paper, we propose, instead, the use of another indicator called Δ_p. Although the Δ_p indicator is not Pareto compliant, we show here how it can be incorporated into the selection mechanism of an evolutionary algorithm (for that sake, we adopt differential evolution as our search engine) in order to produce a MOEA. The resulting MOEA (called Δ_p-Differential Evolution, or DDE) is validated using standard test problems and performance indicators reported in the specialized literature. Our results are compared with respect to those obtained by both a Pareto-based MOEA (NSGA-II) and a hypervolume- based MOEA (SMS-EMOA). Our preliminary results indicate that our proposed approach is competitive with respect to these two MOEAs for continuous problems having two and three objective functions. Additionally, our proposed approach is better than NSGA-II and provides competitive results with respect to SMS-EMOA for continuous many-objective problems. However, in this case, the main advantage of our proposal is that its computational cost is significantly lower than that of SMS-EMOA.

Categories and Subject Descriptors

I.2.8 [**Computing Methodologies**]: Artificial Intelligence—*Problem Solving, Control Methods, and Search.*

General Terms

Algorithms, Theory.

*The second author acknowledges support from CONACyT project no. 103570.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GECCO'12, July 7–11, 2012, Philadelphia, Pennsylvania, USA.
Copyright 2012 ACM 978-1-4503-1177-9/12/07 ...$10.00.

Keywords

Multi-objective optimization, multi-objective evolutionary algorithms, performance assessment indicators

1. INTRODUCTION

A wide variety of evolutionary algorithms (EAs) have been proposed for solving multi-objective optimization problems (MOPs) [7]. Since EAs do not guarantee the optimality of their solution sets, it is of interest to compare their performance. Several quality assessment indicators have been proposed to evaluate and compare the outcome sets of multi-objective evolutionary algorithms (MOEAs). Assuming that a quality assessment indicator provides a good ordering among sets that represent Pareto approximations, the fitness function used to select individuals may be defined in such a way that the chosen indicator is optimized. In fact, the use of performance indicators to guide the search of a MOEA has been a relatively recent trend regarding algorithmic design [2, 13]. The main motivation has been to overcome the poor performance exhibited by Pareto-based selection schemes when dealing with many-objective optimization problems (i.e., MOPs having four or more objectives) [15]. It is worth noting, however, that indicators can be incorporated into a MOEA in different ways, depending of our aim. In the specialized literature, we can find three main approaches to integrate an indicator into a MOEA:

1. **As an archiving algorithm:**
   An indicator is used to decide which subset of the non-dominated solutions are stored in a bounded archive [14, 12].

2. **As a selection mechanism:**
   The objective function is recast as the optimization of an indicator through appropriate selection mechanisms [22, 2, 19].

3. **As a set preference relation:**
   The optimization problem is viewed as a set problem where the search space consists of all possible Pareto set approximations. An indicator can be used to compare such sets [23, 1, 24].

To date, most indicator-based MOEAs rely on Pareto compliant indicators [25]. However, the only unary Pareto-dominance compliant indicator currently known is the hypervolume indicator [5].[1] The main advantage of the hypervolume indicator is that it has been proved that the

[1]The **Hypervolume** (also known as the S metric or the

maximization of this performance measure is equivalent to finding the Pareto optimal set [11]. This has been empirically corroborated [14, 10] and, in fact, the maximization of this indicator also leads to sets of solutions whose spread along the Pareto front is maximized (although the distribution of such solutions is not necessarily uniform). However, the main disadvantage of adopting this indicator is that the best algorithms known to compute the hypervolume have a computational cost which grows exponentially on the number of objectives [3, 4].

In spite of the evident disadvantages of the hypervolume, its nice mathematical properties have triggered an important amount of research, as well as the development of MOEAs whose selection mechanism is based on this indicator (see for example the S Metric Selection-Evolutionary Multiobjective Optimization Algorithm (SMS-EMOA) [2], which bases its selection mechanism on the hypervolume combined with the non-dominated sorting procedure adopted in NSGA-II [8]).

An interesting question, however, is the following: is it possible to develop a selection mechanism based on another (computationally inexpensive) performance indicator that can properly guide the search of a MOEA in an analogous way as the hypervolume does it? Here, we attempt to answer to this question in a positive way. For that sake, we explore the use of the Δ_p indicator which, in spite of not being Pareto compliant, has other properties that can be exploited by a MOEA. As we will see later on, the selection mechanism that is introduced in this paper has a similar performance as that of a representative Pareto-based MOEA (the NSGA-II [8]) and that of a representative hypervolume-based MOEA (SMS-EMOA [2]). Additionally, when the number of objectives is increased, our proposed approach outperforms NSGA-II and remains competitive with respect to SMS-EMOA, but at a much lower computational cost.

The remainder of this paper is organized as follows. We define the basic concepts related to the contents of this paper in Section 2. The Δ_p indicator is presented in Section 3 while its integration into a MOEA is discussed in Section 4. Section 6 presents our experimental design (indicating a short description of the test problems and performance assessment indicators adopted in our experimental study) and a case study that aims to analyze the scalability of our proposed approach (regarding the number of objectives). Finally, Section 7 provides our conclusions and some possible paths for future research.

2. BASIC CONCEPTS

In multi-objective optimization problems (MOPs), the aim is to find a set of decision variable vectors which represent optimal trade-offs among all the objectives. We assume that each objective function should be minimized.

The most commonly adopted approach for solving MOPs is to compare their decision variable vectors using the Pareto dominance relation.

Definition 1. *Given two decision variable vectors $\vec{x}, \vec{y} \in \mathbb{R}^n$ and a function $F : \mathbb{R}^n \rightarrow \mathbb{R}^k$, $\vec{x}$ dominates $\vec{y}$ ($\vec{x} \prec \vec{y}$) if*

$$\forall i \in \{1, \ldots, k\} \quad f_i(\vec{x}) \leq f_i(\vec{y}) \tag{1}$$

$$\exists j \in \{1, \ldots, k\} \quad f_j(\vec{x}) < f_j(\vec{y}) \tag{2}$$

otherwise $\vec{x} \not\preceq \vec{y}$

Within Pareto dominance, we can distinguish between *strong* dominance and *weak* dominance.

Definition 2. *A solution $\vec{x}$ strongly dominates $\vec{y}$ if $\vec{x}$ is strictly better than $\vec{y}$ in all objectives*

Definition 3. *A solution $\vec{x}$ weakly dominates $\vec{y}$ if $\vec{x}$ is better than $\vec{y}$ in at least one objective and is as good as $\vec{y}$ in all other objectives.*

Neither type of Pareto dominance induces a total order in $\mathbb{R}^k$ since some solutions may be *incomparable*. Therefore, most MOPs do not have a single solution but a set of incomparable solutions which is called the *Pareto optimal set*.

Definition 4. *In a MOP, the Pareto optimal set O is defined as*

$$O = \{\vec{x} \in \mathbb{R}^k | \forall \vec{y} \in \mathbb{R}^k \ \vec{y} \not\preceq \vec{x}\} \tag{3}$$

Definition 5. *Given a MOP and its Pareto optimal set O, the Pareto front is defined as*

$$PF = \{\vec{u} = (f_1(\vec{x}), \ldots, f_k(\vec{x})) | \vec{x} \in O\} \tag{4}$$

The Pareto front of a MOP is bounded by the *ideal* and *nadir* objective vectors

Definition 6. *Given a MOP and its Pareto optimal set O, the ideal objective vector is defined as*

$$f_{ideal} = \left(\inf_{x \in O} f_1(\vec{x}), \ldots, \inf_{x \in O} f_k(\vec{x}) \right) \tag{5}$$

If the ideal objective vector represents an existing solution, then the solution of the MOP is unique.

Analogously, the nadir objective vector is defined as

$$f_{nadir} = \left(\sup_{x \in O} f_1(\vec{x}), \ldots, \sup_{x \in O} f_k(\vec{x}) \right) \tag{6}$$

3. THE Δ_P INDICATOR

The goal of a MOEA is to find a set of solutions that are as close as possible to the true Pareto front of a MOP. It is also desirable that such solutions are well-distributed in objective function space. The quality of the approximation set generated by a MOEA is often evaluated through quality assessment indicators which allow us to perform a quantitative comparison of the performance of several MOEAs. Next, we will discuss a recently proposed performance indicator called Δ_p, which is the one adopted for the work reported here.

The Δ_p indicator [17], which can be viewed as an averaged Hausdorff distance between an approximation set and the Pareto front of a MOP, is composed of two (slightly modified) quality indicators: Generational Distance (GD) [18] and Inverted Generational Distance (IGD) [6].

Definition 7. *Given an approximation set A and a discretized Pareto front PF= $(p_1, p_2, \ldots, p_{|PF|})$ of a MOP, the (slightly modified) GD indicator is defined as [18, 17]*

$$I_{GD_p} = \left(\frac{1}{|A|} \sum_{i=1}^{|A|} d_i^p \right)^{\frac{1}{p}} \tag{7}$$

where d_i is the Euclidean distance from a_i to its nearest member of PF.

Lebesgue Measure) of a set of solutions measures the size of the portion of objective space that is dominated by those solutions collectively [20].

Definition 8. *Given an approximation set A and a discretized Pareto front PF= $(p_1, p_2, \ldots, p_{|PF|})$ of a MOP, the (slightly modified) IGD indicator is defined as [6, 17]*

$$I_{IGD_p} = \left(\frac{1}{|PF|} \sum_{i=1}^{|PF|} \tilde{d}_i^p \right)^{\frac{1}{p}} \tag{8}$$

where d_i is the Euclidean distance from pf_i to its nearest member of A.

Both I_{GD_p} and I_{IGD_p} have (weak) metric properties [17]:

- I_{GD_p} and I_{IGD_p} are non-negative
- I_{GD_p} and I_{IGD_p} are non-symmetric
- I_{GD_p} and I_{IGD_p} don't satisfy the (relaxed) triangle inequality

Definition 9. *Given an approximation set A and a discretized Pareto front PF= $(p_1, p_2, \ldots, p_{|PF|})$ of a MOP, the Δ_p indicator is defined as [17]:*

$$I_{\Delta_p} = \max\left(I_{GD_p}, I_{IGD_p}\right) \tag{9}$$

The Δ_p indicator has better metric properties than either the GD or the IGD indicators [17]:

- It is positive and symmetric: I_{Δ_p} is a semi-metric.
- If the magnitudes of the sets are bounded, the relaxed triangle inequality is satisfied and I_{Δ_p} is a pseudo-metric.
- If $p = \infty$ then I_{Δ_p} is a metric (the Hausdorff distance).

The Δ_p indicator is not Pareto compliant but its properties can be exploited by a MOEA to guide its search in a proper way.

4. OUR PROPOSED APPROACH

The aim of the work reported here was to design a MOEA which attempted to minimize the Δ_p indicator. The idea was, of course, that such an algorithm would be able to operate as traditional MOEAs (i.e., that it would provide solutions as close as possible to the true Pareto front and with a reasonably good distribution in objective function space). In our proposed approach, the MOP is recast as:

$$\min I_{\Delta_p}(P) \tag{10}$$

where P is the population used by a differential evolution algorithm (in its DE/rand/1/bin version) where:

- individual fitness is determined according to an individual's contribution to the indicator
- survival selection is replaced with a Δ_p-based mechanism
- the outcome of the algorithm is a set instead of an individual

4.1 Fitness Assignment

As previously stated, our aim is to minimize the Δ_p indicator

$$I_{\Delta_p} = \max\left(I_{\mathrm{GD}_p}, I_{\mathrm{IGD}_p}\right) \tag{11}$$

which implies the simultaneous minimization of the GD_p and IGD_p indicators.

Therefore, the fitness assignment of an individual must consider its contribution to I_{GD_p} and I_{IGD_p} to properly reflect its contribution to I_{Δ_p}.

4.1.1 Individual contribution to the GD_p indicator

Since the GD_p indicator takes the power mean of the Euclidean distances between the elements of a set A and the Pareto front PF, an individual's contribution must reflect its distance to PF. This leads us to the following definition.

Definition 10. *The contribution of an individual a_i to the GD indicator ($I_{GD}[a_i]$) can be defined in a straightforward manner as:*

$$I_{GD}[a_i] = d_i \tag{12}$$

where d_i is the Euclidean distance from a_i to its nearest member of PF.

The elements of A with lower GD contributions are preferred to those with higher GD contributions.

4.1.2 Individual contribution to the IGD_p indicator

The IGD_p indicator takes the power mean of the Euclidean distances between the elements of the Pareto front PF and a set A. Since I_{IGD_p} considers only the elements of A which are closest to at least one element of PF, some elements of A may not contribute to the value of this indicator while other elements can have several contributions. This leads us to the following definition.

Definition 11. *Let Q be the set of all elements of the Pareto front PF for which $a_i \in A$ is the closest element in A. The contribution of a_i to I_{IGD_p} can then be defined as:*

$$I_{IGD}[a_i] = \begin{cases} \sqrt[p]{\sum_{q \in Q} dist(q, a_i)^p} & Q \neq \emptyset \\ -1 & Q = \emptyset \end{cases} \tag{13}$$

where dist is the Euclidean distance between two points.

The elements of A with higher IGD contributions are preferred to those with lower IGD contributions, since the former often cover larger sections of the Pareto front.

4.1.3 Individual contribution to the Δ_p indicator

An individual's contribution to the Δ_p indicator must consider its contributions to both the GD and the IGD indicators. Since the indicator is to be minimized, individuals with lower Δ_p contributions should be preferred with respect to those with higher Δ_p contributions.

Definition 12. *Let A be an approximation set for a MOP with Pareto front PF. Given two elements $a_i, a_j \in A$, a_i contributes less than a_j to I_{Δ_p} if one of the following conditions holds:*

1. *a_i contributes to I_{IGD_p} and a_j does not.*

$$I_{IGD}[a_i] > 0 \quad \textit{and} \quad I_{IGD}[a_j] < 0 \tag{14}$$

2. *a_i contributes more to I_{IGD_p} than a_j*

$$I_{IGD}[a_i] > I_{IGD}[a_j] \tag{15}$$

3. *a_i and a_j contribute equally to I_{IGD_p} but a_i is closer to the Pareto front*

$$I_{IGD}[a_i] = I_{IGD}[a_j] \quad \textit{and} \quad I_{GD}[a_i] < I_{GD}[a_j] \tag{16}$$

In our proposed approach, an individual's contribution to the IGD indicator is given more importance than its contribution to the GD indicator since I_{IGD_p} reflects an individual's importance in terms of both convergence and coverage of the Pareto front while I_{GD_p} only reflects its contribution to convergence.

4.2 Reference Set Construction

The Δ_p indicator needs two sets:

- the *approximation set* A:
  given by the population of the MOEA
- the *reference set* R:
  a (discretized) Pareto front for the MOP

Since the true Pareto front is normally unknown, an approximation must be constructed to be used as the reference set for the MOEA. Some desirable properties for the reference set are the following:

1. All elements in A should be dominated by elements of R.
2. R should be (easily) updated if A is slightly modified.
3. No element in R must (strongly) dominate other elements in the reference set
4. The elements of R should be evenly spaced.

Our proposal for building the reference set is described in Algorithm 1.

Algorithm 1 Reference set construction

1: Approximate the ideal and nadir objective vectors (see Section 4.2.1)
2: Build an upper k-dimensional frame bounded by the ideal and nadir vectors (see Section 4.2.2)
3: Fit the frame to the non-dominated points in A (see Section 4.2.3)
4: Remove duplicated points (see Section 4.2.4)

4.2.1 Ideal and nadir vector approximation

The approximate ideal vector is set as the known minimum for each objective function:

$$\vec{b}_i = \left(\min_{\vec{x} \in A} f_1(\vec{x}), \min_{\vec{x} \in A} f_2(\vec{x}), \ldots, \min_{\vec{x} \in A} f_k(\vec{x}) \right) \tag{17}$$

where A contains all known non-dominated solutions to the MOP. An element of the approximate ideal vector is updated whenever an improvement is achieved in the corresponding objective function.

The approximate nadir vector is defined using only the non-dominated solutions in the current generation G

$$\vec{b}_n = \left(\max_{\vec{x} \in G} f_1(\vec{x}), \max_{\vec{x} \in G} f_2(\vec{x}), \ldots, \max_{\vec{x} \in G} f_k(\vec{x}) \right) \tag{18}$$

To reduce the computational cost of building the reference set too often, an element of the nadir vector approximation is only updated when the corresponding element in $\vec{b}_i$ changed or the difference between the current value and the updated value is "large enough".

4.2.2 Frame building

In order to approximate the Pareto front, a box of uniformly distributed points is set bounded by $\vec{b}_i$ and $\vec{b}_n$. However, only the walls of the box which contain the approximate nadir vector are included in the frame. The frame is built by setting k independent $(k-1)$-dimensional walls. A point $\vec{x}$ in the i-th wall should only be added to the frame if

$$\exists \vec{y} \in Q \quad \forall j \in \{1, \ldots, k\} \backslash \{i\} \quad f_j(\vec{x}) \leq f_j(\vec{y}) \tag{19}$$

where Q is the set of non-dominated solutions in the current population. Since each wall of the frame has some points on the edges of the box, those points will appear duplicated.

The distance between points should be

$$s = \frac{\sum_{j=1}^{k} (b_n[j] - b_i[j])}{kr} \tag{20}$$

where r is the resolution (average number of points on the edges of the frame) of the reference set which is given as a parameter to the MOEA.

4.2.3 Frame fitting

Since the frame was built using the nadir point, several points in the frame could be dominated by points in the population. Therefore, the dominated points must be rearranged to ensure that none of the frame points is strongly dominated and that every element in the population is (weakly) dominated by at least one element in the frame.

The reference set is fitted as described in Algorithm 2.

Algorithm 2 Reference set fitting

1: **for** each objective function f_i **do**
2: **for** each point x on the i-th wall of the frame **do**
3: $Q \leftarrow \{q \in \text{non-dominated solutions} \mid f_m(q) \leq x[m]\}$, $m = \{1, \ldots, k\} \backslash \{i\}$
4: $q_{\min} \leftarrow \min_{q \in Q}(f_i(q))$
5: $x_i \leftarrow b_n[i] - \frac{\lceil b_n[i] - q_{\min}[i] \rceil}{s} s$
6: **end for**
7: **end for**

4.2.4 Removing duplicated points

The fitting proposed before, can lead to several points located in the same position on the reference set. Such points would bias the MOEA towards sections with duplicated points which is evidently, undesirable. Thus, all but one point in a given position are marked as duplicates with only unmarked points contributing to the fitness assignment. The points cannot be completely deleted from the reference set since a future fitting could have those points in different positions.

5. THE ALGORITHM

The full description of our proposed approach (called Δ_p-*Differential Evolution* (DDE) is shown in Algorithm 3. It is worth remarking that some modifications were done to DE/rand/1/bin to adapt the algorithm for multi-objective

optimization using the proposed Δ_p-based selection mechanism:

- The creation and update of the reference set (lines 3 and 18) was included
- Δ_p contributions must be computed considering the objective function values of the population and the reference set (lines 4 and 19)
- The (1+1)-selection mechanism was replaced by an (NP+NP)-selection (lines 17-20)
- The outcome of DE is its final population instead of a single individual (line 22)

Algorithm 3 DE/*rand*/1/*bin* with Δ_p-based selection (DDE)

1: Generate initial population (A) of NP individuals *randomly*
2: Evaluate the objective function for each individual
3: *Initialize the reference set* S
4: *Compute the* I_{Δ_p} *contributions of each individual according to* S
5: **repeat**
6: $\{N\} \leftarrow \emptyset$
7: **for** each individual $\vec{x} \in A$ **do**
8: Pick three distinct individuals $\vec{b}, \vec{c}, \vec{d} \in A$.
9: Pick an index $R \in \{1, \ldots, D\}$
10: **for** $i = 1, \ldots, D$ **do**
11: Pick $r_i \in (0, 1)$ uniformly at random
12: **if** ($i = R$ or $r_i <$CR) **then** $y_i = b_i + F(c_i - d_i)$
13: **else** $y_i = x_i$
14: **end for**
15: $\{N\} \leftarrow \{N\} + \vec{y}$
16: **end for**
17: *Evaluate the objective function for all the generated offspring* N
18: *Update* S
19: *Compute the* I_{Δ_p} contributions *of* $A \cup N$ *according to the updated* S
20: $A \leftarrow$ *the NP individuals which the lowest* I_{Δ_p} *contributions*
21: **until** a stopping criterion is met
22: **return** *the final population*

Our proposed approach assumes that the real Pareto front of the problem is $(k-1)$-dimensional and continuous. Therefore, we don't expect our proposed approach to work well with functions where the Pareto front lost dimensionality or where it is discontinuous.

5.1 Influence of the resolution parameter

The resolution r for the reference set was included as an additional parameter to our proposed approach. This parameter has a strong influence on both the quality of the outcome set of DDE as well as on its computational cost. The main issues to consider are the following:

- If the reference set is too small (i.e., if it has less points than the population), then the quality of the outcome set drastically decreases since it cannot distribute well all the points in the population (the Pareto front would be under-sampled)
- If the reference set is too large (i.e., if it has at least twice as many points as the population), then the computational cost of our approach increases (because of the cost of building and updating a large reference set), while the quality of the outcome set does not improve.

Ideally, the reference set should have approximately the same size as the population for the MOEA since in that scenario, the Pareto front is appropriately sampled for the number of solutions used while maintaining the computational cost as low as possible.

We recommend the value of r to be at least

$$r = \max(3, \sqrt[k-1]{N}) \tag{21}$$

where N is the size of the population since any value lower than 3 would consider only the extreme known values for the functions and any value less than $\sqrt[k-1]{N}$ would under-sample the Pareto front. However, it is important to consider that more points are deleted from the reference set in higher dimensions which implies that, for many-objective problems, the value of r should be larger than the given bound.

6. PERFORMANCE ASSESSMENT

Our proposed DDE was tested on 12 problems from the Zitzler-Deb-Thiele (ZDT) [21] and the Deb-Thiele-Laumanns-Zitzler (DTLZ) [9] test suites. These test problems were adopted with the settings shown in Table 1.

Problem	# variables	# objectives
ZDT 1-3	30	2
ZDT 4,6	10	2
DTLZ 1	13	3
DTLZ 2-6	12	3
DTLZ 7	22	3

Table 1: Test problem settings

We compared our approach with respect to NSGA-II [8] (which is a Pareto-based MOEA representative of the state-of-the-art) and SMS-EMOA [2] (which is a hypervolume-based MOEA representative of the state-of-the-art) using the hypervolume [20] and the Δ_1 [16] indicators.

6.1 Parameters Settings

30 independent runs of each algorithm were performed with the parameters shown in Table 2. The three MOEAs adopted a population size of 100 individuals and were run during 200 generations.

DDE	NSGA-II	SMS-EMOA
$F = 1.0$	$p_c = 0.9$	
Cr$= 0.4$	$p_m = 1/\lvert\vec{x}\rvert$	$p_m = 1/\lvert\vec{x}\rvert$
$p = 1.0$	$n_c = 15$	$n_c = 15$
$r = \lceil \sqrt[k-1]{100} \rceil$	$n_m = 20$	$n_m = 20$

Table 2: Parameters used for each MOEA

The outcome sets were compared using the hypervolume indicator [20] and the Δ_1 indicator [16]. For the Δ_1 indicator, a set of ≈ 150 evenly distributed points of the Pareto front was used.

Test problem	I_{Δ_1}			I_H			
	DDE	NSGA-II	SMS-EMOA	DDE	NSGA-II	SMS-EMOA	Reference point
ZDT1	0.004625	0.005314	**0.003912**	0.870584	0.868009	**0.871766**	(1.1, 1.1)
ZDT2	0.004638	0.005488	**0.004093**	0.536988	0.533828	**0.538295**	(1.1, 1.1)
ZDT3	0.008469	**0.006065**	0.007855	0.945797	0.951098	**0.951629**	(0.9, 1.1)
ZDT4	**0.011449**	0.020748	0.053781	**0.854655**	0.845837	0.825166	(1.1, 1.1)
ZDT6	**0.003549**	0.016471	0.008221	**0.502041**	0.4823269	0.494348	(1.1, 1.1)
DTLZ1	**0.678735**	10.2047	1.426136	**1.020308**	0.000000	0.399135	(1.1, 1.1, 1.1)
DTLZ2	**0.058214**	0.074047	0.063697	0.722791	0.690814	**0.757763**	(1.1, 1.1, 1.1)
DTLZ3	**2.561606**	32.001436	11.754814	**103.861710**	0.169759	11.16606	(5.0, 5.0, 5.0)
DTLZ4	**0.058748**	0.165438	0.063790	0.719107	0.650300	**0.757853**	(1.1, 1.1, 1.1)
DTLZ5	0.297309	0.005608	**0.004923**	0.292570	0.437321	**0.439354**	(1.1, 1.1, 1.1)
DTLZ6	**0.009725**	1.379611	0.460537	**0.436414**	0.000000	0.090737	(1.1, 1.1, 1.1)
DTLZ7	0.092193	**0.085428**	0.110085	2.946258	2.933973	**3.041450**	(1.1, 1.1, 7.0)

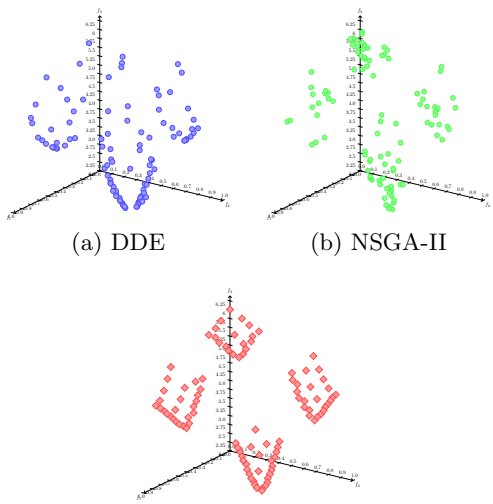
Table 3: Comparison of the average results obtained by NSGA-II, SMS-EMOA and DDE for the ZDT and DTLZ test problems

6.2 Results

The comparison of results is shown in Table 3. We can see in this table that the values obtained by the three MOEAs are, in most cases, very similar. Thus, and because of the obvious space limitations, we will focus our discussion on three particular types of problems:

- Problems with a discontinuous Pareto front
- Problems in which the Pareto front is not $(k-1)$-dimensional
- Multi-frontal problems

As previously stated, we expected our approach to produce bad outcome sets for problems with discontinuous Pareto fronts with the solutions clustering around the boundaries of the Pareto front. Such a behavior can be observed in DTLZ7 (see Figure 1). However, it is interesting to note that the quality of the outcome sets generated by DDE in this case was not much worse than those produced by either NSGA-II or SMS-EMOA in terms of the chosen indicators.

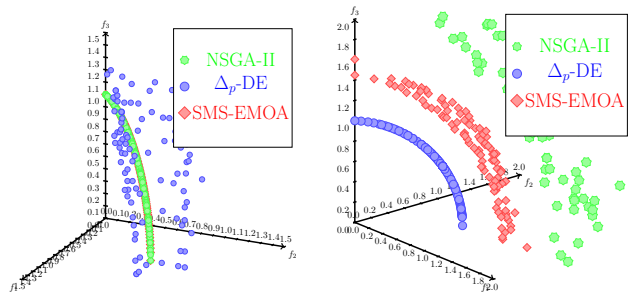


(a) DDE (b) NSGA-II

(c) SMS-EMOA

Figure 1: Graphical results for DTLZ7

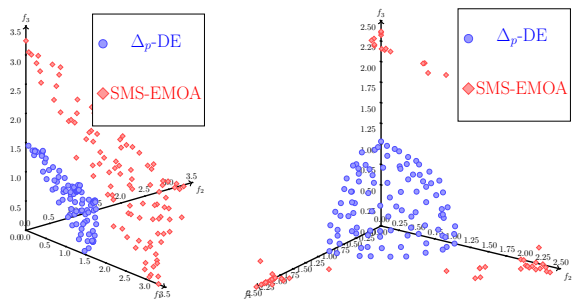
We also indicated that our proposed approach was not expected to work well for problems in which the Pareto front loses dimensionality. Both DTLZ5 and DTLZ6 have this feature (see Figure 2). Although it is clear that our approach has difficulties for solving DTLZ5 (which has a bias towards solutions close to the Pareto front) it could easily solve DTLZ6, which is normally considered to be a harder MOP than DTLZ5.



(a) DTLZ5 (b) DTLZ6

Figure 2: Graphical results for DTLZ5 and DTLZ6

Also, our proposed approach seems less prone to get trapped in local optima than both NSGA-II and SMS-EMOA. The results obtained by our approach in highly multi- frontal problems such as DTLZ1 and DTLZ3 (see Figure 3) are significantly better than those obtained by the two other MOEAs with respect to which it was compared.



(a) DTLZ1 (b) DTLZ3

Figure 3: Graphical results for DTLZ1 and DTLZ3

6.3 Many-Objective Optimization: A Case Study

In order to evaluate the ability of our proposed approach to deal with many-objective optimization problems, we scaled the DTLZ2 test problem from 2 to 10 objectives, in order to use it as our case study. Again, we compared the performance of our approach to both NSGA-II and SMS-EMOA. In this case, however, only 10 independent runs of each algorithm were performed, mainly because of the high computational cost of SMS- EMOA. For these runs, we used a population of 200 individuals running for 200 generations using the same parameters as before but adjusting the resolution values for DDE as shown in Table 4.

Objectives (k)	2	3	4	5	6	7	8	9	10
Resolution (r)	200	15	6	4		3			

Table 4: Resolution values for DDE

The outcome sets of the algorithms were compared using the exact generational distance indicator (which, due to the geometry of the Pareto front for DTLZ2, can be computed as $\sum_{i=1}^{|\mathrm{NP}|}\left(\sqrt{\sum_{j=1}^{k} f_j(a_i)^2}-1\right)$, the hypervolume indicator(using $\overrightarrow{1.1}$ as the reference point), and the running time of each MOEA.

We can see in Figure 4 how NSGA-II has a quick performance degradation as we increased the number of objectives. This does not occur with SMS- EMOA or with our proposed DDE. However, it is clear that SMS-EMOA is better than our proposed approach in terms of the performance indicators adopted. However, quantitatively, the difference is not significant, since our approach reached only slightly worse results than SMS-EMOA. Our proposed DDE obtained, on average, over 95% of the best hypervolume value found by SMS-EMOA. In terms of computational time, however, there is a significant difference, since our proposed approach required less than 0.1% of the computational time consumed by SMS-EMOA (see Figure 5).

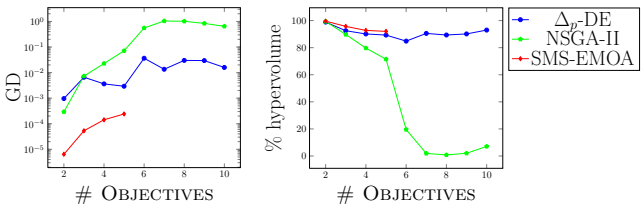


Figure 4: Indicator values for DTLZ2

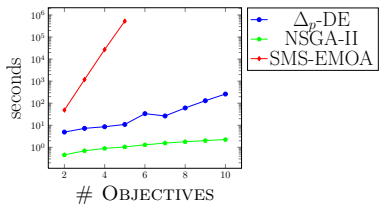


Figure 5: Running time for DTLZ2

These preliminary results lead us to claim that our proposed DDE constitutes a viable alternative for solving MOPs, and we particularly encourage its use in many-objective optimization problems. As we have seen, SMS-EMOA, which is also an indicator-based MOEA, can properly handle many-objective optimization problems. However, its high computational cost can easily make it unaffordable for MOPs having more than 5 objectives. This high computational cost is evidently due to the hypervolume calculation that SMS-EMOA requires.

7. CONCLUSIONS AND FUTURE WORK

We have proposed a MOEA based on the minimization of a performance indicator called Δ_p. Our proposed approach was shown to obtain competitive results when compared to NSGA-II and SMS-EMOA using several test problems and performance indicators taken from the specialized literature.

As we saw in our results, when dealing with problems having two or three objectives, our proposed approach has a similar performance as both NSGA-II and SMS-EMOA. We also saw that in these low dimensionality MOPs, our proposed approach produced significantly better results than NSGA-II and SMS-EMOA in highly multi-frontal MOPs (i.e., problems having many false Pareto fronts). Conversely, the main limitation of our proposed approach is that it is not well-suited for MOPs in which the Pareto front is discontinuous or is not $(k-1)$-dimensional. However, in our experiments, we found the first feature to generate difficulties to NSGA-II and SMS-EMOA as well, while the second feature did not have a significant impact on performance all the time. Nevertheless, these features require more attention and will be the focus of our future work. We believe that it is possible to improve the procedure to construct the reference set, so that we can properly deal with these features, and that will be part of our future work.

When dealing with a MOP having many objectives, our proposed approach performed better than NSGA-II and was outperformed by SMS-EMOA. However, the values of the indicators did not show a significant difference in performance and we showed how our proposed approach was significantly faster than SMS-EMOA as we increased the number of objectives of a MOP. These results are very encouraging, but it is clear to us that a more thorough study of the scalability of our proposed approach is required and that is, indeed, part of our ongoing research.

8. REFERENCES

[1] J. Bader, D. Brockhoff, S. Welten, and E. Zitzler. On Using Populations of Sets in Multiobjective Optimization. In M. Ehrgott, C. M. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux, editors, *Evolutionary Multi-Criterion Optimization. 5th International Conference, EMO 2009*, pages 140–154. Springer. Lecture Notes in Computer Science Vol. 5467, Nantes, France, April 2009.

[2] N. Beume, B. Naujoks, and M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 16 September 2007.

[3] K. Bringmann and T. Friedrich. Approximating the volume of unions and intersections of high-dimensional geometric objects. In *Proceedings of the 19th International Symposium on Algorithms and*

Computation, pages 436–447, Berlin, Heidelberg, 2008. Springer-Verlag.

[4] K. Bringmann and T. Friedrich. Approximating the Least Hypervolume Contributor: NP-Hard in General, But Fast in Practice. In M. Ehrgott, C. M. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux, editors, *Evolutionary Multi-Criterion Optimization. 5th International Conference, EMO 2009*, pages 6–20. Springer. Lecture Notes in Computer Science Vol. 5467, Nantes, France, April 2009.

[5] D. Brockhoff, T. Friedrich, and F. Neumann. Analyzing Hypervolume Indicator Based Algorithms. In G. Rudolph, T. Jansen, S. Lucas, C. Poloni, and N. Beume, editors, *Parallel Problem Solving from Nature–PPSN X*, pages 651–660, Dortmund, Germany, September 2008. Springer. Lecture Notes in Computer Science Vol. 5199.

[6] C. A. Coello Coello and N. Cruz Cortés. Solving Multiobjective Optimization Problems using an Artificial Immune System. *Genetic Programming and Evolvable Machines*, 6(2):163–190, June 2005.

[7] C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, New York, second edition, September 2007. ISBN 978-0-387-33254-3.

[8] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.

[9] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable Test Problems for Evolutionary Multiobjective Optimization. In A. Abraham, L. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, pages 105–145. Springer, USA, 2005.

[10] M. Emmerich, N. Beume, and B. Naujoks. An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, editors, *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005*, pages 62–76, Guanajuato, México, March 2005. Springer. Lecture Notes in Computer Science Vol. 3410.

[11] M. Fleischer. The Measure of Pareto Optima. Applications to Multi-objective Metaheuristics. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, editors, *Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003*, pages 519–533, Faro, Portugal, April 2003. Springer. Lecture Notes in Computer Science. Volume 2632.

[12] S. Huband, P. Hingston, L. White, and L. Barone. An Evolution Strategy with Probabilistic Mutation for Multi-Objective Optimisation. In *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, volume 3, pages 2284–2291, Canberra, Australia, December 2003. IEEE Press.

[13] H. Ishibuchi, N. Tsukamoto, and Y. Nojima. Iterative Approach to Indicator-Based Multiobjective Optimization. In *2007 IEEE Congress on Evolutionary Computation (CEC'2007)*, pages 3967–3974, Singapore, September 2007. IEEE Press.

[14] J. Knowles and D. Corne. Properties of an Adaptive Archiving Algorithm for Storing Nondominated Vectors. *IEEE Transactions on Evolutionary Computation*, 7(2):100–116, April 2003.

[15] J. Knowles and D. Corne. Quantifying the Effects of Objective Space Dimension in Evolutionary Multiobjective Optimization. In S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, editors, *Evolutionary Multi-Criterion Optimization, 4th International Conference, EMO 2007*, pages 757–771, Matshushima, Japan, March 2007. Springer. Lecture Notes in Computer Science Vol. 4403.

[16] O. Schütze, X. Esquivel, A. Lara, and C. A. C. Coello. Measuring the Averaged Hausdorff Distance to the Pareto Front of a Multi-objective Optimization Problem. Technical Report TR-OS-2010-02, CINVESTAV, 2010.

[17] O. Schütze, X. Esquivel, A. Lara, and C. A. C. Coello. Measuring the Averaged Hausdorff Distance to the Pareto Front of a Multi-objective Optimization Problem. *IEEE Transactions on Evolutionary Computation*, 2012. (to appear)

[18] D. A. V. Veldhuizen. *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, Wright-Patterson AFB, Ohio, May 1999.

[19] T. Wagner and H. Trautmann. Integration of Preferences in Hypervolume-Based Multiobjective Evolutionary Algorithms by Means of Desirability Functions. *IEEE Transactions on Evolutionary Computation*, 14(5):688–701, October 2010.

[20] E. Zitzler. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, November 1999.

[21] E. Zitzler, K. Deb, and L. Thiele. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, 8(2):173–195, Summer 2000.

[22] E. Zitzler and S. Künzli. Indicator-based Selection in Multiobjective Search. In X. Y. et al., editor, *Parallel Problem Solving from Nature - PPSN VIII*, volume LNCS 3242, pages 832–842, Birmingham, UK, September 2004. Springer-Verlag.

[23] E. Zitzler, L. Thiele, and J. Bader. SPAM: Set Preference Algorithm for Multiobjective Optimization. In G. Rudolph, T. Jansen, S. Lucas, C. Poloni, and N. Beume, editors, *Parallel Problem Solving from Nature–PPSN X*, pages 847–858. Springer. Lecture Notes in Computer Science Vol. 5199, Dortmund, Germany, September 2008.

[24] E. Zitzler, L. Thiele, and J. Bader. On Set-Based Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 14(1):58–79, February 2010.

[25] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca. Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, April 2003.