Solving Multi-Objective Optimization Problems using Differential Evolution and a Maximin Selection Criterion

Adriana Menchaca-Mendez and Carlos A. Coello Coello

***Abstract*—In this paper, we propose a new selection operator (based on a maximin scheme and a clustering tecnique), which is incorporated into a differential evolution algorithm to solve multi-objective optimization problems. The resulting algorithm is called Maximin-Clustering Differential Evolution (MCDE) and, is validated using standard test problems and performance measures taken from the specialized literature. Our preliminary results indicate that MCDE is able to outperform NSGA-II and that is competitive with a hypervolume-based approach (SMS-EMOA), but at a significantly lower computational cost.**

I. Introduction

Many optimization problems arising in the real world involve multiple objective functions which must be satisfied simultaneously. They are generically called *multiobjective optimization problems (MOPs)* and usually their objectives are in conflict. In MOPs, the notion of optimality refers to the best possible trade-offs among the objectives. Consequently, no single solution exists, but several (the so-called *Pareto optimal set* whose image is called the *Pareto front*). When applying evolutionary algorithms to solve MOPs, we normally have two main goals [1]: (i) to find solutions that are, as close as possible, to the true Pareto front and, (ii) to produce solutions that are spread along the Pareto front as uniformly as possible.

When studying multi-objective evolutionary algorithms (MOEAs), we find two main types of approaches: (i) those that incorporate the concept of Pareto optimality in their selection mechanism, and (ii) those that do not use Pareto dominance to select individuals.

Although the use of Pareto-based selection (mainly through the use of some Pareto ranking scheme [1]) has been the most popular choice within the specialized literature for the last 15 years, such type of approach has several limitations. From them, its poor scalability (when increasing the number of objectives) is, perhaps, the most remarkable. The quick increase in the number of nondominated solutions as we increase the number of objectives, rapidly dilutes the effect of the selection mechanism of a MOEA [2]. This has triggered an important amount of research on the so-called "many-objective optimization", which refers to the study of problems having four or more objective functions.

Adriana Menchaca-Mendez and Carlos A. Coello Coello are with the Departamento de Computación, CINVESTAV-IPN, Av. IPN No. 2508, Col. San Pedro Zacatenco, México, D.F. 07360, MEXICO (email: ccoello@cs.cinvestav.mx). The second author is also with the UMI-LAFMIA 3175 CNRS. The first author acknowledges support from CONACyT to pursue graduate studies at the Computer Science Department of CINVESTAV-IPN. The second author acknowledges support from CONACyT project no. 103570.

In the current literature, we can identify three main approaches to cope with many-objectives problems, namely: (i) to adopt or propose a preference relation that induces a finer grain order on the solutions than that induced by the Pareto dominance relation [3], [4], [5], [6], (ii) to reduce the number of objectives of the problem during the search process [7] or, *a posteriori*, during the decision making process [8], [9], and, (iii) to adopt a selection scheme that does not rely on Pareto optimality (e.g., using compromise functions [10], alternative ranking schemes [11] or a selection mechanism based on a performance measure (from which hypervolume[1] has been a popular choice, in spite of its considerably high computational cost [13], [14]). Here, we study an approach from the third class, using differential evolution as our search engine. The main motivation of this work is to propose an alternative selection mechanism for MOEAs that can properly deal with many-objective optimization problems at a reasonably low computational cost.

The focus of our study is the *maximin fitness function* [15]. This technique assigns a fitness to each individual in the population without using the concept of Pareto optimality. This scheme encompasses a guidance mechanism based on very simple (and computationally efficient) operations. Our preliminary study of this approach has indicated its suitability as a selection operator in a MOEA whose search engine adopts differential evolution [16], even in the presence of a high number of objectives. However, its lack of an appropriate diversity maintenance mechanism makes it inappropriate with respect to state-of-the-art MOEAs, which led us to propose the incorporation of a clustering technique. The proposed approach, called Maximin-Clustering Differential Evolution (MCDE) is validated with several standard test problems and performance measures. As will be seen later on, our proposed MCDE is able to outperform NSGA-II [17] and is competitive with a state-of-the-art hypervolume-based MOEA (SMS-EMOA) [14], but requiring a much lower computational cost.

The remainder of this paper is organized as follows. Section II states the problem of our interest. The maximin fitness function is briefly described in Section III. Section IV describes in detail the selection operator that we propose and in Section V we present a full description of our approach. The experiments performed and the results obtained are shown in Section VI. Finally, we provide our conclusions and some possible paths for future work in Section VII.

[1]The **hypervolume** (also known as the S metric or the Lebesgue Measure) of a set of solutions measures the size of the portion of objective space that is dominated by those solutions collectively [12].

II. Problem Statement

The problem of our interest is the general multi-objective optimization problem (MOP) which is defined as follows:

Find $\vec{x}^* = [x_1^*, x_2^*, \ldots, x_n^*]^T$ which optimizes

$$\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x})]^T \quad (1)$$

such that $\vec{x}^* \in \Omega$, where $\Omega \subset R^n$ defines the feasible region of the problem. Assuming minimization problems, we have the following definitions.

Definition 1: We say that a vector $\vec{u} = [u_1, \ldots, u_n]^T$ dominates to vector $\vec{v} = [v_1, \ldots, v_n]^T$, denoted by $\vec{u} \leq_p \vec{v}$, if and only if $f_i(\vec{u}) \leq f_i(\vec{v})$ for all $i \in \{1, ..., k\}$ and there exists an $i \in \{1, \ldots, k\}$ such that $f_i(\vec{u}) < f_i(\vec{v})$.

Definition 2: A point $\vec{x}^* \in \Omega$ is Pareto optimal if and only if for all $\vec{x} \in \Omega$ we have that $\vec{x}^* \leq_p \vec{x}$ where $\vec{x}^* \neq \vec{x}$.

Definition 3: For a given MOP, $\vec{f}(\vec{x})$, the Pareto optimal set is defined as: $\mathcal{P}^* = \{\vec{x} \in \Omega | \neg\exists \vec{y} \in \Omega : \vec{f}(\vec{y}) \leq_p \vec{f}(\vec{x})\}$.

Definition 4: Let $\vec{f}(\vec{x})$ be a given MOP and $\mathcal{P}^*$ the Pareto optimal set. Then the Pareto Front is defined as: $\mathcal{PF}^* = \{\vec{f}(\vec{x}) \mid \vec{x} \in \mathcal{P}^*\}$.

III. Maximin fitness function

The maximin fitness function presented in [15] is derived from the *Definition* 1 and, we can use it to solve MOPs. Let's consider a MOP with K objectives and let's assume that each objective is normalized. Let's consider an evolutionary algorithm whose population size is P. Let f_k^i be the normalized value of the k^{th} objective for the i^{th} individual in a particular generation. Assuming minimization, the j^{th} individual dominates the i^{th} individual if:

$$\begin{aligned} f_k^j \leq f_k^i \quad \forall k \in \{1, \ldots, K\} \quad \text{and} \\ \exists k \in \{1, \ldots, K\} \mid f_k^j < f_k^i \end{aligned} \quad (2)$$

Eq. (2) is equivalent to:

$$min_k(f_k^i - f_k^j) \geq 0 \quad (3)$$

The i^{th} individual in a particular generation will be dominated by another individual in the generation if:

$$max_{j \neq i}(min_k(f_k^i - f_k^j)) > 0 \quad (4)$$

Then, the maximin fitness function of individual i is defined as:

$$fitness^i = max_{j \neq i}(min_k(f_k^i - f_k^j)) \quad (5)$$

In eq. (5), the min is taken over all the objectives from 1 to K, and the max is taken over all the individuals in the population from 1 to P, except for the same individual i. From eq. (5), we know that any individual whose maximin fitness is greater than zero is a dominated solution and that, any individual whose maximin fitness is less than zero is a non-dominated solution. Finally, any individual whose maximin fitness is equal to zero is either a dominated solution or a duplicate non-dominated solution.

The algorithm to calculate the fitness of each individual of the population is shown in Algorithm 1. The complexity of this algorithm is $O(KP^2)$, where K is the number of objective fuctions and P is the population size. In Algorithm 1, we multiply the fitness of each individual by -1 in order to obtain a higher fitness for the individuals which are non-dominated and a lower fitness for the individuals which are dominated.

Algorithm 1: MaximinFitnessFunction

```
Input  : X (Current population with their normalized
         objective values), P (population size) and, K
         (number of objectives).
Output: X (Current population with the fitness updated
         of each individual).
for i ← 1 to P do
    maximal ← −∞;
    for j ← 1 to P do
        if j ≠ i then
            minimal ← ∞;
            for k ← 1 to K do
                if X[i].f[k] − X[j].f[k] < minimal
                then
                    minimal ← X[i].f[k] − X[j].f[k];
                end
            end
            if minimal > maximal then
                maximal ← minimal;
            end
        end
    end
    X[i].fitness ← −1 ∗ maximal;
end
```

IV. Selection Operator

When studying the maximin fitness function, we identified an important disadvantage of this approach when solving MOPs. If we take a look at the part where we get the minimum of the difference between normalized objective values of two given solutions, we can notice that if a particular objective is minimized more quickly than the others, then the remaining objectives are not considered. Let's assume that we have a MOP with two objective functions (f_1 and f_2) and that the objective f_1 is easier to optimize than objective f_2. In this case, when the maximin fitness function calculates the minimum, it will often obtain the component of objective f_1, without regarding f_2. Thus, if the maximin fitness function is incorporated into an evolutionary algorithm, it may occur that we only obtain solutions that minimize f_1, instead of finding the best possible trade-offs among the objectives, which is our aim. Evidently, we cannot assume that the objectives can be optimized separately, since that assumption would only be reasonable when the objectives have no conflict among themselves, which makes employing a multi-objective approach unnecessary (a single-objective optimizer would be sufficient in that case).

In order to address this problem, we propose to check if the individual that we want to select is similar in at least one objective to another (selected) individual. The process to verify similarity between individuals is shown in Algorithm 2. As can be seen, we need the parameter min_dif which indicates the minimum difference that must separate the two selected individuals in objective function space (this is similar to the niche radius adopted with fitness sharing [18]).

The complexity of Algorithm 2 is $O(KP)$, where K is the number of objectives and P is the population size. The full selection process based on a maximin fitness function is shown in Algorithm 3 and its complexity is $O(KP^2)$.

Algorithm 2: IsSimilarToAny

```
Input  : min_dif (Minimum difference), x
         (individual), Y (population), P (population
         size), K (number of objectives).
Output: Returns 1, if the individual x is similar to any
         individual in the population Y; otherwise,
         returns 0.
for i ← 1 to P do
    for k ← 1 to K do
        if |x.f[k] − Y[i].f[k]| < min_dif then
            Returns 1;
        end
    end
end
Returns 0;
```

Algorithm 3: MaximinSelection

```
Input  : X_sorted (Sorted population from high to low,
         according to the fitness values), P (population
         size), K (number of objectives) and S (the
         number of individuals to choose).
Output: Y (individuals selected)
s ← 1;
i ← 1;
/*Before selecting individuals, verify
  that there is not a similar one   */
while s ≤ S AND i ≤ P do
    while IsSimilarToAny(X_sorted[i], Y) = 1 AND
    i ≤ P do
        i ← i + 1;
    end
    if i ≤ P then
        Y[s] ← X_sorted[i];
        s ← s + 1;
    end
end
/*Select only according to fitness   */
i ← 1;
while s ≤ S do
    if X_sorted[i] has not been selected then
        Y[s] ← X_sorted[i];
        s ← s + 1;
    end
    i ← i + 1;
end
Returns Y;
```

As indicated before, the search engine of a MOEA has two main goals: (i) to find solutions which are as close as possible to the true Pareto front and, (ii) to produce solutions that are spread along the Pareto front as uniformly as possible.

The first of these goals is achieved by the mechanism described in Algorithm 3. However, we need one more mechanism, so that we can fulfill the second objective. Here, we propose to use the clustering technique described in Algorithm 4.

V. Maximin-Clustering Differential Evolution

The approach that we propose here is called Maximin-Clustering Differential Evolution (MCDE) and is described next.

MCDE adopts the operators of differential evolution to create new individuals but the selection process is modified as follows: If the size of the population is P, MCDE creates P new individuals. After that, it combines the population of parents and offspring to obtain a population of size $2P$. Then, MCDE uses Algorithms 3 and 4 to choose the P individuals that will take part of the following generation.

In order to speed up convergence, we also propose to modify the process in which parents are selected to participate in the process of mutation and crossover as follows. Instead of randomly selecting three individuals for becoming parents (as normally done in the DE algorithm), we use a binary tournament selection for choosing the three individuals needed. At each tournament, two individuals are randomly selected and the one with the higher fitness value is chosen. Finally, Algorithm 5 shows the full algorithm of our proposed MCDE approach.

VI. Experimental results

We compared our proposed MCDE with respect to two MOEAs representative of the state-of-the-art in the area:

- The Nondominated Sorting Genetic Algorithm II (NSGA-II) [17], which is a well-known MOEA whose selection mechanism is based on Pareto dominance. NSGA-II also incorporates a crowded comparison operator to produce well-distributed solutions along the Pareto front. This MOEA was chosen because is perhaps the most representative of the Pareto-based MOEAs.
- The S Metric Selection-Evolutionary Multiobjective Optimization Algorithm (SMS-EMOA) [14], which bases its selection mechanism on the hypervolume performance measure [12] combined with the non-dominated sorting procedure adopted in NSGA-II. This approach

Algorithm 4: ClusteringSelection

Input : X_{sorted} (Sorted population from high to low, according to fitness), P (population size), K (number of objectives) and S (number of individuals to choose).
Output: Y (individuals selected).

```
/*Choose the first S individuals as
  centers of the clusters C                    */
C_j = {X_sorted[j]};
/*Do clustering                                */
for i ← S+1 to P do
    if X_sorted[i] is closer to C_j then
        C_j ← C_j ∪ X_sorted[i];
    end
end
/*Obtain the new centers of the
  clusters                                     */
for j ← 1 to S do
    μ_j ← 1/|C_j| Σ_{X[i]∈C_j} X[i];
end
/*Select to individuals who are
  closest to the centers of the
  clusters                                     */
for j ← 1 to S do
    if X[i] | X[i] ∈ C_j is the nearest to the center μ_j
    then
        Y[j] ← X[i];
    end
end
Returns Y;
```

was chosen because is a state-of-the-art hypervolume-based MOEA.

It is important to emphasize that we chose these algorithms because our aim was to validate the selection mechanism of our proposed approach. Although there are several MOEAs based on DE (see for example [19], [20], [21], [22]), most of them adopt a Pareto-based selection mechanism and, therefore, were not considered for our comparative study (we decided to adopt NSGA-II instead, because of its widespread use and availability (its source code is available in the public domain)). We believe that it would be more interesting to compare results with respect to an scalarization method such as MOEA/D (there is a version based on DE) [23], but this was not done here because of time constraints.

We performed 30 independent runs for each test problem. The parameters adopted for our proposed MCDE are shown in Table I (these values were empirically derived after numerous experiments). For both NSGA-II and SMS-EMOA, we adopted the parameters suggested by the authors of NSGA-II: crossover probability $p_c = 0.9$, mutation probability $p_m = 1/n$, where n is the number of decision variables, for crossover and mutation operators, $\eta_c = 15$ and $\eta_m = 20$, respectively. Finally, we used the same population size and

Algorithm 5: Maximin-Clustering Differential Evolution

Input : P (population size), g_{max} (maximum number of generations), N (number of decision variables), K (number of objective functions) Cr (crossover probability) and min_dif (minimum difference between objectives).
Output: Last population.

```
contGen ← 0;
Create a random initial population;
repeat
    for i ← 1 to P do
        Select three parents (X[i_1], X[i_2] and X[i_3]) ;
        Obtain the new individual (X_new[i]) from DE's
        mutation and crossover operators;
    end
    Y ← X ∪ X_new;
    Y_sorted ← SortCurrentPopulation(X);
    if The number of nondominated individuals is
    greater to S then
        X ← ClusteringSelection(Y_sorted, P, K, S);
    else
        X ← MaximinSelection(Y_sorted, P, K, S);
    end
    contGen ← contGen + 1;
until contGen < g_max;
```

the same value for the maximum number of generations with the three algorithms compared. All three algorithms performed the same number of objective function evaluations (for the ZDT test problems they performed 10,000 evaluations, and for the DTLZ test problems they performed 20,000 evaluations, except for ZDT4 and DTLZ3 in which they performed 36,000 and 25,000 evaluations, respectively).

	F	Pr_{cr}	min_dif	P	G
ZDT1	0.5	0.9	0.00001	100	100
ZDT2	0.5	0.9	0.001	100	100
ZDT3	0.5	0.9	0.00001	100	100
ZDT4	0.5	0.23	0.001	120	300
ZDT6	0.5	0.9	0.001	100	100
DTLZ1	0.5	0.0001	0.0001	100	200
DTLZ2	0.5	0.01	0.001	100	200
DTLZ3	0.5	0.00001	0.001	100	250
DTLZ4	0.5	0.00001	0.001	100	200
DTLZ5	0.5	0.5	0.00001	100	200
DTLZ6	0.5	0.3	0.00001	100	200
DTLZ7	0.5	0.1	0.00001	100	200

TABLE I
PARAMETERS ADOPTED BY OUR PROPOSED MCDE ALGORITHM. F AND Pr_{cr} ARE PARAMETERS USED BY THE DE ALGORITHM; min_dif IS USED BY OUR PROPOSED SELECTION OPERATOR; P IS THE POPULATION SIZE AND G IS THE MAXIMUM NUMBER OF GENERATIONS.

A. Test problems

To validate our proposed MCDE, we adopted two sets of problems. The first consists of five bi-objective test problems taken from the Zitzler-Deb-Thiele suite [24]. The second consists of seven problems having three or more objectives, taken from the Deb-Thiele-Laumanns-Zitzler (DTLZ) suite [25]. For the DTLZ test problems, we used $k = 5$ and, three, four and five objetive functions (i.e., $M = 3$,4, and 5).

B. Quality indicators

To assess performance, we adopted the following indicators:

1) *Generational Distance* (GD). This indicator was proposed by Van Veldhuizen in [26] and, it represents how far a set A is from the Pareto front $\mathcal{P}$. Formally, it is defined as:

$$GD = \frac{1}{|A|}\left(\sum_{i=1}^{|A|} d_i^2\right)^{\frac{1}{2}} \tag{6}$$

where d_i is the Euclidean distance, in objective function space, between a_i and the nearest member of $\mathcal{P}$. *Lower GD values represent better approximations*. For the calculation of the GD indicator, we used the following:

- Pareto optimal sets generated by an enumerative approach for the ZDT test problems [1].
- A set of equations that describe the Pareto fronts for the DTLZ test problems 1 to 6.
- A value of the function g for the test problem DTLZ7.

2) *Hypervolume indicator* (φ). It was originally proposed by Zitzler and Thiele in [27], and it's defined as the size of the space covered by the Pareto optimal solutions. φ rewards convergence towards the Pareto front as well as the maximum spread of the solutions obtained. If $\bigwedge$ denotes the Lebesgue measure, φ is defined as:

$$\varphi(A, y_{ref}) = \bigwedge\left(\bigcup_{y\in A}\{y' \mid y < y' < y_{ref}\}\right) \tag{7}$$

where $y_{ref} \in R_k$ denotes a reference point that should be dominated by all the Pareto optimal points. Fleischer proved in [28] that, given a finite search space and a reference point, *maximizing the hypervolume indicator is equivalent to finding the Pareto optimal set*. The disadvantage of this indicator is its high computational cost (the running time for calculating φ is exponential in the number of objective functions). To compute φ, we used the following reference points:

- For ZDT functions, we used $y_{ref} = [1, 1]$.
- For function DTLZ1, we used $y_{ref} = [y_1, \ldots, y_M] \mid y_i = 0.7\ \forall i = 1, \ldots, M$.
- For functions DTLZ(2-6), we used $y_{ref} = [y_1, \ldots, y_M] \mid y_i = 1.1\ \forall i = 1, \ldots, M$.
- For function DTLZ7, we used $y_{ref} = [y_1, \ldots, y_M]\ y_M = 6.1$ and $y_i = 1.1\ \forall i = 1, \ldots, M-1$.

C. Results

In Table II, we can observe that our proposed MCDE outperforms both NSGA-II and SMS-EMOA, in most of the ZDT test problems (the best results are shown in **boldface**). As shown in Tables III and V, in the DTLZ test problems having three and five objective functions, SMS-EMOA is better than MCDE in four test problems and MCDE is better than SMS-EMOA in three test problems. As shown in Table IV, for the test problems having four objective functions, SMS-EMOA is better than MCDE in five test problems and MCDE is better than SMS-EMOA only in two. It is important to note that our proposed MCDE presents a consistent behavior when increasing the number of objectives, unlike NSGA-II whose performance quickly degrades, reaching a value of zero for the hypervolume indicator when solving three of the DTLZ test problems.

To validate the results in our experiments, we performed statistical analysis using Wilcoxon's rank sum. Table VII shows the results. With respect to generational distance, we can say that our algorithm, MCDE, is significantly better than SMS-EMOA in DTLZ6 and DTLZ7, with three objective functions, and also in DTLZ7, with four objective functions because the hypothesis that the medians are equal can be rejected. In most of the remaining problems, we can say that SMS-EMOA and MCDE have a similar behavior, except for ZDT6, DTLZ5, with three objective functions, DTLZ1, DTLZ3 and DTLZ4, with five objective functions because the probability that the hypothesis is true is less than 0.5. Regarding the hypervolume indicator only in DTLZ2 with five objectives, we can say that SMS-EMOA is significantly better than MCDE. Finally, we can say that SMS-EMOA and MCDE have a similar behavior in most of the ZDT and DTLZ problems with three and five objective functions.

Based on the results shown before, we claim that the performance of MCDE is competitive with respect to the performance of SMS-EMOA. However, it is important to note that the computational cost of the SMS-EMOA algorithm is considerably larger than that of MCDE. Table VI shows the CPU time, per run, required by each algorithm. In this table, we can note that MCDE needs only 1 or 2 seconds in any of the test problems adopted, even for instances having five objectives. In contrast, SMS-EMOA needs up to 11 hours, per run, for the test problems having five objectives.

This difference is due to the fact that computing the maximim fitness function is an inexpensive process (its complexity is linear with respect to the number of objectives), whereas the computation of the hypervolume is exponential with respect to the number of objectives. Thus, we argue that MCDE can be a good alternative for dealing with many-objective optimization problems, unless we can afford a very high computational cost.

VII. Conclusions and future work

We have proposed a new selection operator to solve multiobjective optimization problems using a single-objective evolutionary algorithm (differential evolution in our case).

$\vec{f}$	I	NSGA-II	SMS-EMOA	MCDE
1	φ	0.843798 (0.003867)	0.865892 (0.001641)	**0.866497 (0.001268)**
	gd	0.001896 (0.000259)	0.000449 (0.000053)	**0.000161 (0.000063)**
2	φ	0.477477 (0.067691)	0.528328 (0.002507)	**0.529499 (0.002874)**
	gd	0.002925 (0.000679)	0.000581 (0.000110)	**0.000410 (0.000141)**
3	φ	1.294509 (0.006019)	0.711646 (0.010531)	**1.323276 (0.003000)**
	gd	**0.001384 (0.000222)**	0.020268 (0.001389)	0.006132 (0.030231)
4	φ	0.869855 (0.001417)	**0.869862 (0.002264)**	0.859478 (0.032763)
	gd	**0.000496 (0.000058)**	0.002413 (0.010458)	0.000894 (0.002144)
6	φ	0.233452 (0.031991)	0.230881 (0.035046)	**0.501168 (0.001371)**
	gd	0.022658 (0.003359)	0.021088 (0.003213)	**0.000112 (0.000013)**

TABLE II

RESULTS OBTAINED IN THE ZDT TEST PROBLEMS. WE SHOW AVERAGE VALUES OVER 30 INDEPENDENT RUNS. THE VALUES IN PARENTHESES CORRESPOND TO THE STANDARD DEVIATIONS.

$\vec{f}$	I	NSGA-II	SMS-EMOA	MCDE
1	φ	0.168104 (0.121049)	0.282027 (0.071556)	**0.301742 (0.050681)**
	gd	0.056767 (0.060737)	**0.012122 (0.024779)**	0.275396 (0.745076)
2	φ	0.695722 (0.007437)	**0.757970 (0.000048)**	0.721690 (0.011281)
	gd	0.000327 (0.000132)	**0.000000 (0.000000)**	0.000003 (0.000013)
3	φ	0.495070 (0.296949)	**0.663793 (0.230938)**	0.574081 (0.287628)
	gd	0.041826 (0.061893)	**0.013266 (0.033151)**	1.027464 (3.315866)
4	φ	0.688702 (0.011839)	**0.757973 (0.000044)**	0.707000 (0.014417)
	gd	0.000361 (0.000179)	**0.000000 (0.000000)**	0.000044 (0.000147)
5	φ	0.437974 (0.000258)	**0.439344 (0.000018)**	0.427400 (0.005107)
	gd	0.000066 (0.000037)	**0.000000 (0.000000)**	0.000498 (0.000084)
6	φ	0.263933 (0.023633)	0.401663 (0.021284)	**0.429099 (0.008223)**
	gd	0.017425 (0.002951)	0.003793 (0.001723)	**0.000000 (0.000000)**
7	φ	1.871739 (0.143621)	1.824769 (0.346389)	**1.956336 (0.012535)**
	gd	0.727283 (0.123080)	0.850543 (0.087943)	**0.000000 (0.000000)**

TABLE III

RESULTS OBTAINED IN THE DTLZ TEST PROBLEMS WITH THREE OBJECTIVE FUNCTIONS. WE SHOW AVERAGE VALUES OVER 30 INDEPENDENT RUNS. THE VALUES IN PARENTHESES CORRESPOND TO THE STANDARD DEVIATIONS.

$\vec{f}$	I	NSGA-II	SMS-EMOA	MCDE
1	φ	0.020431 (0.049314)	**0.216245 (0.047649)**	0.203586 (0.060700)
	gd	0.426400 (0.484402)	**0.012505 (0.030799)**	0.019485 (0.043234)
2	φ	0.864022 (0.019920)	**1.044446 (0.000062)**	0.957406 (0.019200)
	gd	0.000962 (0.000296)	**0.000002 (0.000001)**	0.000231 (0.000747)
3	φ	0.320878 (0.384497)	**0.827933 (0.414164)**	0.775603 (0.312009)
	gd	0.175820 (0.161358)	**0.020897 (0.039890)**	0.743255 (2.924236)
4	φ	0.852950 (0.020846)	**1.044515 (0.000087)**	0.961788 (0.010907)
	gd	0.001021 (0.000273)	**0.000001 (0.000001)**	0.000166 (0.000679)
5	φ	0.429414 (0.002911)	**0.439084 (0.000260)**	0.277086 (0.029985)
	gd	**0.021393 (0.004046)**	0.045849 (0.003666)	0.057070 (0.002998)
6	φ	0.000000 (0.000000)	0.220479 (0.016245)	**0.240899 (0.037808)**
	gd	0.155895 (0.014608)	**0.140056 (0.007726)**	0.156666 (0.014808)
7	φ	0.518986 (0.060112)	0.345574 (0.248708)	**0.602277 (0.037409)**
	gd	0.737255 (0.092109)	0.860942 (0.102134)	**0.000000 (0.000000)**

TABLE IV

RESULTS OBTAINED IN THE DTLZ TEST PROBLEMS WITH FOUR OBJECTIVE FUNCTIONS. WE SHOW AVERAGE VALUES OVER 30 INDEPENDENT RUNS. THE VALUES IN PARENTHESES CORRESPOND TO THE STANDARD DEVIATIONS.

This operator takes into account the two main objectives of a MOEA: it uses a maximin technique to find solutions as close as possible to the true Pareto front and it uses a clustering technique to provide a good distribution of such solutions along the Pareto front. We chose these techniques in order to obtain a selection operator capable of solving problems of both low dimensionality (with two or three objective functions) and high dimensionality (more than three objective functions).

In our experimental study, we compared the performance of our proposed approach with respect to a state-of-the-art Pareto-based MOEA (NSGA-II) and with respect to a state-of-the-art hypervolume-based MOEA (SMS-EMOA) using standard test problems and performance measures taken from the specialized literature. Our results show that MCDE outperforms NSGA-II in all cases (low- and high-dimensionality test problems) and produces competitive results with respect to SMS-EMOA, but at a much lower computational cost.

As part of our future work, we plan to incorporate our selection operator into other evolutionary algorithms in order

$\vec{f}$	I	NSGA-II	SMS-EMOA	MCDE
1	φ	0.000000 (0.000000)	0.153918 (0.031912)	**0.155104 (0.020443)**
	gd	6.702562 (2.084582)	0.018024 (0.039251)	**0.012216 (0.020552)**
2	φ	0.978240 (0.024770)	**1.295810 (0.000108)**	1.147529 (0.026441)
	gd	0.003487 (0.000808)	**0.000007 (0.000002)**	0.000976 (0.002029)
3	φ	0.000000 (0.000000)	**1.245409 (0.231560)**	0.826641 (0.421045)
	gd	11.347256 (4.076464)	**0.004221 (0.018154)**	2.353595 (7.336803)
4	φ	0.970531 (0.031486)	**1.296010 (0.000114)**	1.164448 (0.027469)
	gd	0.002992 (0.001193)	**0.000002 (0.000001)**	0.000631 (0.001325)
5	φ	0.423968 (0.010354)	**0.449985 (0.000421)**	0.196839 (0.017682)
	gd	**0.043428 (0.006622)**	0.059179 (0.002101)	0.065526 (0.004445)
6	φ	0.000000 (0.000000)	0.156974 (0.016202)	**0.166601 (0.053506)**
	gd	0.272611 (0.024261)	0.165866 (0.012073)	**0.161178 (0.015361)**
7	φ	0.074212 (0.011193)	**0.084705 (0.065528)**	0.044723 (0.019550)
	gd	0.743493 (0.123118)	0.804493 (0.095978)	**0.030316 (0.026227)**

TABLE V

RESULTS OBTAINED IN THE DTLZ TEST PROBLEMS WITH FIVE OBJECTIVE FUNCTIONS. WE SHOW AVERAGE VALUES OVER 30 INDEPENDENT RUNS. THE VALUES IN PARENTHESES CORRESPOND TO THE STANDARD DEVIATIONS.

Function	Objectives	SMS-EMOA	MCDE
ZDT1	2	$\approx 3s$	$\approx 1s$
ZDT2	2	$\approx 3s$	$\approx 1s$
ZDT3	2	$\approx 4s$	$\approx 1s$
ZDT4	2	$\approx 9s$	$\approx 1s$
ZDT6	2	$\approx 2s$	$\approx 1s$
DTLZ1	3	$\approx 2m$	$\approx 1s$
DTLZ2	3	$\approx 4m$	$\approx 1s$
DTLZ3	3	$\approx 3m$	$\approx 1s$
DTLZ4	3	$\approx 4m$	$\approx 1s$
DTLZ5	3	$\approx 2m$	$\approx 1s$
DTLZ6	3	$\approx 2m$	$\approx 1s$
DTLZ7	3	$\approx 3m$	$\approx 1s$
DTLZ1	4	$\approx 40m$	$\approx 1s$
DTLZ2	4	$\approx 60m$	$\approx 2s$
DTLZ3	4	$\approx 40m$	$\approx 1s$
DTLZ4	4	$\approx 40m$	$\approx 1s$
DTLZ5	4	$\approx 20m$	$\approx 2s$
DTLZ6	4	$\approx 30m$	$\approx 2s$
DTLZ7	4	$\approx 20m$	$\approx 1s$
DTLZ1	5	$\approx 8h$	$\approx 1s$
DTLZ2	5	$\approx 11h$	$\approx 2s$
DTLZ3	5	$\approx 8h$	$\approx 2s$
DTLZ4	5	$\approx 11h$	$\approx 2s$
DTLZ5	5	$\approx 6h$	$\approx 2s$
DTLZ6	5	$\approx 7h$	$\approx 2s$
DTLZ7	5	$\approx 3h$	$\approx 1s$

TABLE VI

TIME REQUIRED BY SMS-EMOA AND MCDE, PER RUN, FOR THE TEST PROBLEMS ADOPTED. s = SECONDS, m = MINUTES, AND h = HOURS. BOTH ALGORITHMS WERE IMPLEMENTED IN THE C PROGRAMMING LANGUAGE AND THEY WERE EXECUTED ON PCS WITH THE SAME HARDWARE AND SOFTWARE CHARACTERISTICS.

to assess the impact of the search engine in the performance of our algorithm. We also plan to perform a statistical analysis of the sensitivity of our proposed approach to its parameters. Finally, we plan to compare our algorithm with other algorithms that use techniques not based on Pareto optimality or the hypervolume indicator, such as *MOEA/D* [23] and *maximinPSO* [29]. It is important to mention, however, that maximinPSO was tested in [29] only with bi-objective optimization problems. Its authors also reported a poor performance of this approach in ZDT2.

REFERENCES

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

[2] M. Farina and P. Amato, "On the Optimal Solution Definition for Many-criteria Optimization Problems," in *Proceedings of the NAFIPS-FLINT International Conference'2002*, (Piscataway, New Jersey), pp. 233–238, IEEE Service Center, June 2002.

[3] F. di Pierro, S.-T. Khu, and D. A. Savić, "An Investigation on Preference Order Ranking Scheme for Multiobjective Evolutionary Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 11, pp. 17–45, February 2007.

[4] M. Farina, "A Neural Network Based Generalized Response Surface Multiobjective Evolutionary Algorithm," in *Congress on Evolutionary Computation (CEC'2002)*, vol. 1, (Piscataway, New Jersey), pp. 956–961, IEEE Service Center, May 2002.

[5] A. Sülflow, N. Drechsler, and R. Drechsler, "Robust Multi-objective Optimization in High Dimensional Spaces," in *Evolutionary Multi-Criterion Optimization, 4th International Conference, EMO 2007* (S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, eds.), (Matshushima, Japan), pp. 715–726, Springer. Lecture Notes in Computer Science Vol. 4403, March 2007.

[6] H. Sato, H. E. Aguirre, and K. Tanaka, "Controlling Dominance Area of Solutions and Its Impact on the Performance of MOEAs," in *Evolutionary Multi-Criterion Optimization, 4th International Conference, EMO 2007* (S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, eds.), (Matshushima, Japan), pp. 5–20, Springer. Lecture Notes in Computer Science Vol. 4403, March 2007.

[7] D. Brockhoff and E. Zitzler, "Improving Hypervolume-based Multiobjective Evolutionary Algorithms by Using Objective Reduction Methods," in *2007 IEEE Congress on Evolutionary Computation (CEC'2007)*, (Singapore), pp. 2086–2093, IEEE Press, September 2007.

[8] D. Brockhoff, T. Friedrich, N. Hebbinghaus, C. Klein, F. Neumann, and E. Zitzler, "Do Additional Objectives Make a Problem Harder?," in *2007 Genetic and Evolutionary Computation Conference (GECCO'2007)* (D. Thierens, ed.), vol. 1, (London, UK), pp. 765–772, ACM Press, July 2007.

[9] A. López Jaimes, C. A. Coello Coello, and D. Chakraborty, "Objective Reduction Using a Feature Selection Technique," in *2008 Genetic and Evolutionary Computation Conference (GECCO'2008)*, (Atlanta, USA), pp. 674–680, ACM Press, July 2008. ISBN 978-1-60558-131-6.

[10] P. J. Bentley and J. P. Wakefield, "Finding Acceptable Solutions in the Pareto-Optimal Range using Multiobjective Genetic Algorithms," in *Soft Computing in Engineering Design and Manufacturing* (P. K. Chawdhry, R. Roy, and R. K. Pant, eds.), Part 5, (London), pp. 231–240, Springer Verlag London Limited, June 1997. (Presented at the 2nd On-line World Conference on Soft Computing in Design and Manufacturing (WSC2)).

[11] A. López Jaimes, L. V. Santana Quintero, and C. A. Coello Coello, "Ranking methods in many-objective evolutionary algorithms,"

Function	K	I	nsga-ii & mcde $P(H)$	sms-emoa & mcde $P(H)$
ZDT1	2	φ	0.888303 (0)	0.830255 (0)
		dg	0.695215(0)	1.000000(0)
ZDT2	2	φ	0.610008 (0)	0.841801 (0)
		dg	0.935192(0)	0.678892(0)
ZDT3	2	φ	0.795846 (0)	0.935192 (0)
		dg	0.888303(0)	0.501144(0)
ZDT4	2	φ	0.876631 (0)	0.706164 (0)
		dg	0.695207(0)	0.864990(0)
ZDT6	2	φ	0.569220 (0)	0.853382 (0)
		dg	0.122348(0)	0.180887(0)
DTLZ1	3	φ	0.958731 (0)	0.750587 (0)
		dg	0.830253(0)	0.982307(0)
DTLZ2	3	φ	0.795846 (0)	0.888300 (0)
		dg	0.216819(0)	0.841101(0)
DTLZ3	3	φ	0.395207 (0)	0.845779 (0)
		dg	0.539510(0)	0.911708(0)
DTLZ4	3	φ	0.251881 (0)	0.096263 (0)
		dg	0.277076(0)	0.727340(0)
DTLZ5	3	φ	0.395267 (0)	0.923441 (0)
		dg	0.841799(0)	0.145017(0)
DTLZ6	3	φ	0.970516 (0)	0.864994 (0)
		dg	0.007912(1)	0.046591(1)
DTLZ7	3	φ	0.610008 (0)	0.251881 (0)
		dg	0.000551(1)	0.000551(1)
DTLZ1	4	φ	0.075362 (0)	0.695211 (0)
		dg	0.304177(0)	0.988204(0)
DTLZ2	4	φ	0.510598 (0)	0.122353 (0)
		dg	0.332848(0)	0.646468(0)
DTLZ3	4	φ	0.066284 (0)	0.185767 (0)
		dg	0.539510(0)	0.599684(0)
DTLZ4	4	φ	0.864994 (0)	0.371077 (0)
		dg	0.216979(0)	0.711262(0)
DTLZ5	4	φ	0.482517 (0)	0.347828 (0)
		dg	0.911709(0)	0.739399(0)
DTLZ6	4	φ	0.000048 (1)	0.185767 (0)
		dg	0.739399(0)	0.684323(0)
DTLZ7	4	φ	0.297272 (0)	0.471406 (0)
		dg	0.000551(1)	0.000166(1)
DTLZ1	5	φ	0.000048 (1)	0.340282 (0)
		dg	0.185767(0)	0.347828(0)
DTLZ2	5	φ	0.610008 (0)	0.043584 (1)
		dg	0.510592(0)	0.756142(0)
DTLZ3	5	φ	0.000048 (1)	0.539510 (0)
		dg	0.090490(0)	0.211555(0)
DTLZ4	5	φ	0.559231 (0)	0.706171 (0)
		dg	0.464273(0)	0.239667(0)
DTLZ5	5	φ	0.569220 (0)	0.473347 (0)
		dg	0.876635(0)	0.911709(0)
DTLZ6	5	φ	0.000048 (1)	0.750587 (0)
		dg	0.773120(0)	0.673495(0)
DTLZ7	5	φ	0.807275 (0)	0.818746 (0)
		dg	0.510598(0)	0.599689(0)

TABLE VII

RESULTS OF STATISTICAL ANALYSIS APPLIED TO EXPERIMENTS THAT WE DID. K IS THE NUMBER OF OBJECTIVE FUNCTIONS, P IS THE PROBABILITY OF OBSERVING THE GIVEN RESULT (THE NULL HYPOTHESIS IS TRUE). SMALL VALUES OF P CAST DOUBT ON THE VALIDITY OF THE NULL HYPOTHESIS. $H = 0$ INDICATES THAT THE NULL HYPOTHESIS ("MEDIANS ARE EQUAL") CANNOT BE REJECTED AT THE 5% LEVEL. $H = 1$ INDICATES THAT THE NULL HYPOTHESIS CAN BE REJECTED AT THE 5% LEVEL.

in *Nature-Inspired Algorithms for Optimisation* (R. Chiong, ed.), pp. 413–434, Berlin: Springer, 2009. ISBN 978-3-642-00266-3.

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

[13] E. Zitzler and S. Künzli, "Indicator-based Selection in Multiobjective Search," in *Parallel Problem Solving from Nature - PPSN VIII* (X. Y. et al., ed.), (Birmingham, UK), pp. 832–842, Springer-Verlag. Lecture Notes in Computer Science Vol. 3242, September 2004.

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

[15] R. Balling and S. Wilson, "The Maximin Fitness Function for Multi-objective Evolutionary Computation: Application to City Planning," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)* (L. Spector, E. D. Goodman, A. Wu, W. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M. H. Garzon, and E. Burke, eds.), (San Francisco, California), pp. 1079–1084, Morgan Kaufmann Publishers, 2001.

[16] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution. A Practical Approach to Global Optimization*. Berlin: Springer, 2005. ISBN 3-540-20950-6.

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

[18] K. Deb and D. E. Goldberg, "An Investigation of Niche and Species Formation in Genetic Function Optimization," in *Proceedings of the Third International Conference on Genetic Algorithms* (J. D. Schaffer, ed.), (San Mateo, California), pp. 42–50, George Mason University, Morgan Kaufmann Publishers, June 1989.

[19] H. A. Abbass, "The Self-Adaptive Pareto Differential Evolution Algorithm," in *Congress on Evolutionary Computation (CEC'2002)*, vol. 1, (Piscataway, New Jersey), pp. 831–836, IEEE Service Center, May 2002.

[20] N. K. Madavan, "Multiobjective Optimization Using a Pareto Differential Evolution Approach," in *Congress on Evolutionary Computation (CEC'2002)*, vol. 2, (Piscataway, New Jersey), pp. 1145–1150, IEEE Service Center, May 2002.

[21] F. Xue, A. C. Sanderson, and R. J. Graves, "Pareto-based Multi-Objective Differential Evolution," in *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, vol. 2, (Canberra, Australia), pp. 862–869, IEEE Press, December 2003.

[22] T. Robič and B. Filipič, "DEMO: Differential Evolution for Multiobjective Optimization," in *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005* (C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, eds.), (Guanajuato, México), pp. 520–533, Springer. Lecture Notes in Computer Science Vol. 3410, March 2005.

[23] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, pp. 712–731, December 2007.

[24] E. Zitzler, K. Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," *Evolutionary Computation*, vol. 8, pp. 173–195, Summer 2000.

[25] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multiobjective Optimization," in *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications* (A. Abraham, L. Jain, and R. Goldberg, eds.), pp. 105–145, USA: Springer, 2005.

[26] 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.

[27] E. Zitzler and L. Thiele, "Multiobjective Optimization Using Evolutionary Algorithms—A Comparative Study," in *Parallel Problem Solving from Nature V* (A. E. Eiben, ed.), (Amsterdam), pp. 292–301, Springer-Verlag, September 1998.

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

[29] X. Li, "Better spread and convergence: Particle swarm multiobjective optimization using the maximin fitness function," in *Genetic and Evolutionary Computation GECCO 2004* (K. Deb, ed.), vol. 3102 of *Lecture Notes in Computer Science*, pp. 117–128, Springer Berlin / Heidelberg, 2004.