2013 IEEE Congress on Evolutionary Computation
June 20-23, Cancún, México

MOMBI: A New Metaheuristic for Many-Objective Optimization Based on the $R2$ Indicator

Raquel Hernández Gómez
Computer Science Department
CINVESTAV-IPN
Evolutionary Computation Group
México, D.F. 07360, MÉXICO
Email: rhernandez@computacion.cs.cinvestav.mx

Carlos A. Coello Coello
Computer Science Department
CINVESTAV-IPN
Evolutionary Computation Group
México, D.F. 07360, MÉXICO
Email: ccoello@cs.cinvestav.mx

***Abstract*—The incorporation of performance indicators as the selection mechanism of a multi-objective evolutionary algorithm (MOEA) is a topic that has attracted increasing interest in the last few years. This has been mainly motivated by the fact that Pareto-based selection schemes do not perform properly when solving problems with four or more objectives. The indicator that has been most commonly used for being incorporated in the selection mechanism of a MOEA has been the hypervolume. Here, however, we explore the use of the $R2$ indicator, which presents some advantages with respect to the hypervolume, the main one being its low computational cost. In this paper, we propose a new MOEA called Many-Objective Metaheuristic Based on the $R2$ Indicator (MOMBI), which ranks individuals using a utility function. The proposed approach is compared with respect to MOEA/D (based on scalarization) and SMS-EMOA (based on hypervolume) using several benchmark problems. Our preliminary experimental results indicate that MOMBI obtains results of similar quality to those produced by SMS-EMOA, but at a much lower computational cost. Additionally, MOMBI outperforms MOEA/D in most of the test instances adopted, particularly when dealing with high-dimensional problems having complicated Pareto fronts. Thus, we believe that our proposed approach is a viable alternative for solving many-objective optimization problems.**

I. Introduction

A wide variety of real-world problems have several (often conflicting) objectives that need to be optimized at the same time. They are called multiobjective optimization problems (MOPs) and their solution involves a different notion of optimality than the one used for global (single-objective) optimization. When solving a MOP, we normally aim to find the best possible trade-off among all the objectives. The notion of optimality most commonly used to deal with MOPs is Pareto optimality, and its use produces the so-called Pareto optimal set, which contains the decision variables that correspond to all the solutions that represent the best trade-offs among all the objectives (normally, there will be more than one). The image of the Pareto optimal set is called the Pareto front.

The use of evolutionary algorithms (as well as other bio-inspired metaheuristics) for solving MOPs has become increasingly popular in the last 15 years, giving rise to a wide variety of multi-objective evolutionary algorithms (MOEAs) [1]. The two key algorithmic components of a MOEA are: (1) a selection mechanism that preserves the best possible trade-offs among the objectives (i.e., the so-called nondominated solutions) and (2) a density estimator that allows us to spread solutions along the Pareto front in a uniform way, so that they are as diverse as possible.

For many years, MOEAs have adopted selection mechanisms based on Pareto optimality. These mechanisms preserve solutions that are Pareto optimal with respect to a set of reference (normally the current population), and assign a rank to each of these solutions, such that all the nondominated solutions are considered to be equally good. Pareto-based MOEAs have been very popular since the 1990s, but recent studies have shown that they do not perform properly when dealing with problems having four or more objectives (the so-called many-objective optimization problems) [2], [3]. This has motivated the development of new selection schemes from which the use of quality assessment indicators is probably the most popular [4]. The idea in this case, is to optimize a quality assessment indicator that provides a good ordering among sets that represent Pareto approximations. From the many indicators currently available, the Hypervolume [5] [1] is, with no doubt, the most popular nowadays. The main advantage of the hypervolume indicator is that it has been proved that its maximization is equivalent to finding the Pareto optimal set [7], and this has also been empirically corroborated [8]. In fact, maximizing the hypervolume also leads to sets of solutions whose spread along the Pareto front is maximized (although this does not necessarily mean that such solutions will be uniformly distributed along the Pareto front). Nevertheless, the high computational cost of the hypervolume (its computational cost grows exponentially on the number of objectives [9]) normally makes a selection mechanism based on such indicator prohibitive for problems having more than 5 objectives [10]. The nice mathematical properties of the hypervolume indicator has triggered an important amount of research, including work that focuses on computing it in a more efficient way [11], [12]. It is indeed possible to approximate the hypervolume contribution, significantly reducing its computational cost [11], but few studies of the performance of such approaches with respect to those using exact hypervolume calculations are currently available.

[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 [6].

978-1-4799-0454-9/13/$31.00 ©2013 IEEE

Here, we explore the use of another indicator that is known to have nice mathematical properties [13]: $R2$. In this paper, we propose a new MOEA, called Many-Objective Metaheuristic Based on the $R2$ Indicator (MOMBI) and analyze its performance with respect to that of two well-known approaches: MOEA/D [14], which is based on scalarization and SMS-EMOA [4], which is based on the hypervolume indicator (we use the approach to approximate the hypervolume contribution proposed in [11]).

The remainder of this paper is organized as follows. Section II provides some basic concepts related to multi-objective optimization. The previous related work is briefly discussed in Section III. Our proposed approach is described in detail (including some basic concepts on the $R2$ indicator) in Section IV. The results obtained by our proposed approach are compared with respect to those generated by two state-of-the-art MOEAs, using standard test problems and performance indicators taken from the specialized literature in Section V. Finally, our conclusions and some possible paths for future research are presented in Section VI.

II. Basic Concepts

We are interested in solving problems of the type[2]:

$$\text{minimize } \vec{f}(\vec{x}) := [f_1(\vec{x}), f_2(\vec{x}), \dots, f_m(\vec{x})] \quad (1)$$

subject to:

$$g_i(\vec{x}) \leq 0 \quad i = 1, 2, \dots, p \quad (2)$$

$$h_i(\vec{x}) = 0 \quad i = 1, 2, \dots, q \quad (3)$$

where $\vec{x} = [x_1, x_2, \dots, x_n]^T$ is the vector of decision variables, $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, ..., m$ are the objective functions and $g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, ..., p$, $j = 1, ..., q$ are the constraint functions of the problem.

To describe the concept of optimality in which we are interested, we will introduce next a few definitions.

Definition 1. Given two vectors $\vec{x}, \vec{y} \in \mathbb{R}^m$, we say that $\vec{x} \leq \vec{y}$ if $x_i \leq y_i$ for $i = 1, ..., m$, and that $\vec{x}$ **dominates** $\vec{y}$ (denoted by $\vec{x} \prec \vec{y}$) if $\vec{x} \leq \vec{y}$ and $\vec{x} \neq \vec{y}$.

Definition 2. We say that a vector of decision variables $\vec{x} \in \mathcal{X} \subset \mathbb{R}^n$ is **nondominated** with respect to $\mathcal{X}$, if there does not exist another $\vec{x}' \in \mathcal{X}$ such that $\vec{f}(\vec{x}') \prec \vec{f}(\vec{x})$.

Definition 3. We say that a vector of decision variables $\vec{x}^* \in \mathcal{F} \subset \mathbb{R}^n$ ($\mathcal{F}$ is the feasible region) is **Pareto-optimal** if it is nondominated with respect to $\mathcal{F}$.

Definition 4. The **Pareto Optimal Set** $\mathcal{P}^*$ is defined by:

$$\mathcal{P}^* = \{\vec{x} \in \mathcal{F} | \vec{x} \text{ is Pareto-optimal}\}$$

[2]Without loss of generality, we will assume only minimization problems.

Definition 5. The **Pareto Front** $\mathcal{PF}^*$ is defined by:

$$\mathcal{PF}^* = \{\vec{f}(\vec{x}) \in \mathbb{R}^m | \vec{x} \in \mathcal{P}^*\}$$

We thus wish to determine the Pareto optimal set from the set $\mathcal{F}$ of all the decision variable vectors that satisfy (2) and (3). Note however that in practice, not all the Pareto optimal set is normally desirable (e.g., it may not be desirable to have different solutions that map to the same values in objective function space) or achievable.

III. Related Work

In this section we review the previous related work on the use of indicators in the selection mechanism of a MOEA.

As indicated before, the performance indicator that has been most commonly used for the selection mechanism of a MOEA is the hypervolume [6]. This indicator has several advantages, from which the main one is that it is the only unary indicator which is known to be strictly monotonic [15]. However, computing the hypervolume is exponential in the number of objectives [16] and is sensitive to the choice of the reference point [4].

Currently, there are several MOEAs that incorporate the hypervolume in their selection mechanism (e.g., the S Metric Selection-Evolutionary Multi-Objective Optimization Algorithm (SMS-EMOA) [4] and the multi-objective covariance matrix adaptation evolution strategy (MO-CMA-ES) [17]. However, the high computational overload of these approaches motivated the development of alternative strategies. One of them is to estimate (by means of Monte Carlo simulations) the ranking of a set of individuals that would be induced by the hypervolume indicator, without having to compute the exact indicator values. This is the approach adopted by the Hypervolume Estimation algorithm for multi-objective optimization (HypE) [11].

More recently, a new performance indicator called Δ_p was proposed in [18]. This indicator can be seen as an "averaged Hausdorff distance" between the outcome set and the Pareto front. Δ_p is composed of slight modifications of two well-known performance indicators: generational distance (GD, see [19]) and inverted generational distance (IGD, see [20]). Δ_p is a pseudo-metric which simultaneously evaluates proximity to the Pareto front and spread of solutions along it. Although Δ_p is not Pareto compliant, its computation has a much lower computational cost than that of the hypervolume, and it can also handle outliers, which makes it attractive for assessing performance of MOEAs. It is worth noting, however, that for incorporating Δ_p into the selection mechanism of a MOEA, it is necessary to have an approximation of the true Pareto front at all times. This has motivated the development of techniques that can produce such an approximation in an efficient and effective way. For example, in [21], the authors linearize the nondominated (piecewise linear) front of the current population, and include this mechanism in the Δ_p-EMOA, which is used for solving bi-objective optimization problems. This algorithm is inspired by SMS-EMOA, and is assisted by a secondary population. Δ_p-EMOA performs better than NSGA-II [22], while consuming a lower number of function evaluations. An extension of this approach to three-objective problems is reported in [23]. In this case, the algorithm requires some previous mathematical steps which include reducing the dimensionality of the non-dominated solutions and calculating their convex hull. This version of Δ_p-EMOA achieves a better distribution of solutions than MOEA/D [14], SMS-EMOA and NSGA-II. However, this MOEA requires additional parameters and consumes a high computational time when dealing with many-objective optimization problems.

Another possible approach to incorporate Δ_p into a MOEA is to use an echelon form of the non-dominated individuals for the Pareto front. This is the mechanism adopted in Δ_p-DDE [10], in which Δ_p is used as the selection mechanism of a diferential evolution algorithm. Δ_p-DDE was able to outperform NSGA-II and provided competitive results with respect to SMS-EMOA, but at a considerably lower computational cost for many-objective optimization problems. The main limitation of this approach is that it produces a poor spread of solutions in high-dimensional search spaces. Also, it has some difficulties for dealing with discontinuous Pareto fronts.

Recently, some researchers have recommended to adopt the $R2$ indicator proposed in [24] to compare approximation sets on the basis of a set of utility functions [13]. A utility function is a model of the decision maker's preference that maps each point in the objective space into a utility value. It is worth noticing that the $R2$ indicator is weakly monotonic, and it is correlated with the hypervolume but has a lower computational overhead than such indicator [13]. Because of this, the $R2$ indicator is widely recommended for dealing with many-objective optimization problems and over large non-dominated sets [25]. It is worth emphasizing, however, that the main caveat when trying to use this performance indicator is that each utility function adopted, must be properly scaled.

The $R2$ indicator has been scarcely studied in the context of MOEAs. Here, we explore its potential use as a selection mechanism within a MOEA, emphasizing its possible usefulness in many-objective optimization problems.

IV. Our Proposed Approach

Since the proposed approach is based on the $R2$ indicator, we have to provide more details about this indicator before presenting our actual algorithm.

According to [13], the unary version of the $R2$ indicator for a constant reference set can be expressed as follows:

$$R2(A,U) = -\frac{1}{|U|}\sum_{u\in U}\max_{\vec{a}\in A}\{u(\vec{a})\}, \tag{4}$$

where A is the Pareto set approximation and U is a set of utility functions.

With respect to the choice of the utility functions $u : \mathbb{R}^m \rightarrow \mathbb{R}$, there are several posibilities: weighted linear, weighted Tchebycheff or augmented Tchebycheff functions. We focus on the second one, but using a normalization, which allows us to deal with non-commensurable objective functions (i.e., measured in different units):

$$u_{\vec{w}}(\vec{a}) = -\max_{i\in\{1,\dots,m\}} w_i \left|\frac{a_i - z_i^*}{z_i^{nad} - z_i^*}\right|, \tag{5}$$

where $\vec{w} = \{w_1, ..., w_m\} \in W$ is a given weight vector, $\vec{z}^*$ and $\vec{z}^{nad}$ are the ideal[3] and nadir[4] vectors, respectively. Replacing equation (5) in equation (4) and applying the dual property[5], the $R2$ indicator is defined as:

$$R2(A,W) = \frac{1}{|W|}\sum_{\vec{w}\in W}\min_{\vec{a}\in A}\left\{\max_{i\in\{1,\dots,m\}} w_i \left|\frac{a_i - z_i^*}{z_i^{nad} - z_i^*}\right|\right\}. \tag{6}$$

Since we intend to use $R2$ in the selection mechanism of a MOEA, we need to design a scheme for that purpose. Our proposal here is to produce a nondominated sorting scheme based on the utility functions adopted. The core idea is to group solutions that optimize the set of utility functions chosen, and place such solutions on top, such that they get the first rank (the best). Such points will then be removed and a second rank will be identified in the same manner. The process will continue until all the solutions had been ranked. Clearly, this is a nondominated sorting scheme [26], except for the fact that Pareto dominance is not used in this case.

The formal definition of a rank, derived from equation (6), is presented in equation (7):

$$rank_k = \bigcup_{\vec{w}\in W}\min_{\vec{a}\in A\setminus B_k}\left\{\max_{i\in\{1,\dots,m\}} w_i \left|\frac{a_i - z_i^*}{z_i^{nad} - z_i^*}\right|\right\}, \tag{7}$$

where $B_k = \{\bigcup_x rank_x | k \geq 2, 1 \leq x < k\}$ is the union of solutions with the lowest ranks.

When two individuals contribute with the same Tchebycheff value for a weight vector, then we propose to choose the one with the lower Manhattan norm, defined by:

$$||\vec{a}||_1 = \sum_{i=1}^{m} |a_i|. \tag{8}$$

In order to illustrate our proposed ranking scheme, we present here a hypothetical example of a bi-objective problem. We assume an approximation of the Pareto optimal set, which consists of twelve solutions, as shown in Figure 1. The dashed lines represent the weight vectors $\{(10^{-4},1),(1/3,2/3),(2/3,1/3),(1,10^{-4})\}$, the reference points are set to $\vec{z}^* = (1.0, 1.2)$ and $\vec{z}^{nad} = (8.4, 7.8)$. In Table I, the objective functions and the optimum Tchebycheff value of each solution are shown. The first rank is formed with the solutions that are closest to the weights, according to the Tchebycheff metric, that is points $\{a,b,c,d\}$. The second rank consists of the the remainder solutions, which are now

[3] A lower bound of all the objective functions.

[4] An upper bound of each objective in the entire Pareto optimal set.

[5] $\min \vec{z} = -\max(-\vec{z})$

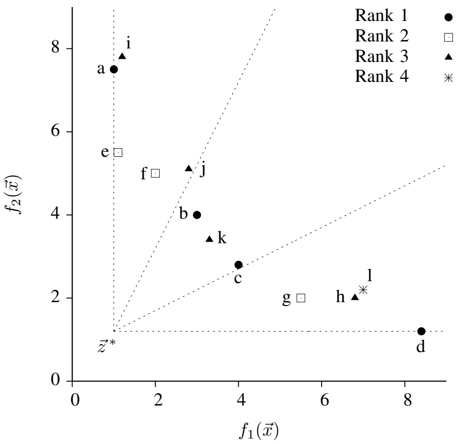

Fig. 1. Illustration of the ranking procedure based on the $R2$ indicator that we propose here.

TABLE I. A HYPOTHETICAL EXAMPLE FOR MOMBI.

Solution	f_1	f_2	$u^*(\vec{a})$
a	1.0	7.5	0.00009
b	3.0	4.0	0.18018
c	4.0	2.8	0.16161
d	8.4	1.2	0.00010
e	1.1	5.5	0.01351
f	2.0	5.0	0.13513
g	5.5	2.0	0.12121
h	6.8	2.0	0.12121
i	1.2	7.8	0.02702
j	2.8	5.1	0.19696
k	3.3	3.4	0.20720
l	7.0	2.2	0.15151

closest to the weights, i.e., points $\{e, f, g\}$. The third rank consists of points $\{h, i, j, k\}$. Finally, the farthest solution l belongs to the last rank. It is worth noticing that, in this case, solutions g and h contribute equally to the weight $(10^{-4}, 1)$. However g has a lower Manhattan norm than h (7.5 vs 8.8) and is, therefore, considered to be better than h.

A. Our Proposed Ranking Algorithm

In Algorithm 1, we present a naive approach to rank a population, based on equation (7). We assume that each individual p has the following structure:

$p.rank$	Hierarchy of the individual.
$p.u^*$	The best utility value obtained.
$p.\alpha$	The current utility value for a weight vector $\vec{w}$.
$p.\vec{f}$	Vector of objective functions.

Lines 1 to 4 initialize the variables $rank$ and u^* for each individual to the worst values. In lines 5 to 11, for every pair of weight vector $\vec{w}$ and objective function $\vec{f}$ of an individual p, the utility value is computed and stored in record $p.\alpha$. If the obtained value of an individual outperforms its previous one, it is updated in field $p.u^*$. In line 12, the population P is sorted with respect to the field $p.\alpha$ in increasing order. Lines 13 to 19 perform the ranking assignment of the sorted population.

As mentioned in the previous section, in line 12, when two individuals have the same utility function, we prefer the individual with the lowest Manhattan norm. This procedure does not guarantee having consecutive ranks. However, this is irrelevant for comparison purposes.

Algorithm 1 $R2$ Ranking Algorithm

Require: Population P, set of weight vectors W, reference points $\vec{z}^*$ and $\vec{z}^{nad}$
Ensure: Ranking of the population
1: **for all** $p \in P$ **do**
2: $\quad p.rank \leftarrow \infty$
3: $\quad p.u^* \leftarrow \infty$
4: **end for**
5: **for all** $\vec{w} \in W$ **do**
6: $\quad$ **for all** $p \in P$ **do**
7: $\quad\quad p.\alpha \leftarrow u_{\vec{w}}(p.\vec{f}, \vec{z}^*, \vec{z}^{nad})$
8: $\quad\quad$ **if** $p.\alpha < p.u^*$ **then**
9: $\quad\quad\quad p.u^* \leftarrow p.\alpha$
10: $\quad\quad$ **end if**
11: $\quad$ **end for**
12: $\quad$ Sort the pupulation P with respect to the field α in increasing order
13: $\quad rank \leftarrow 1$
14: $\quad$ **for all** $p \in P$ **do**
15: $\quad\quad$ **if** $rank < p.rank$ **then**
16: $\quad\quad\quad p.rank \leftarrow rank$
17: $\quad\quad$ **end if**
18: $\quad\quad rank \leftarrow rank + 1$
19: $\quad$ **end for**
20: **end for**

In Algorithm 2 we introduce our proposed approach, called MOMBI. This approach is based on a Genetic Algorithm, and it first initializes the population by randomly selecting N solutions from $\mathcal{F}$ (using a uniform distribution). In lines 3 to 5, we obtain the objective function values, the reference points, and the ranking of the population. At each generation, the algorithm performs a binary tournament selection, using the rank of each solution (line 7). In line 8, we use mutation and crossover operators to produce an offspring of N individuals. The reference points are updated with the minimum and maximum objective function values in line 10. The parent and offspring population are ranked in line 11. In line 12, the reduction of the population is performed by selecting the best N candidates according to their rank, the best utility value obtained, and the Manhattan norm.

It is worth indicating that this approach produces a finer-grained ranking (with fewer ties) than the nondominated sorting procedure adopted by NSGA-II. Re-taking our previous example, if we wanted to select a half of the solutions, using MOMBI, we would keep the individuals $\{a, b, c, d, e, g\}$, since e, f and g are in the same rank. In this case, we choose the solutions with the lowest Tchebycheff values. According to Pareto dominance, the nondominated solutions are $\{a, b, c, d, e, f, g, k\}$. Although f and k are nondominated, in this case $R2$ removes them with the aim of preserving diversity.

The set of weights W is controlled by a parameter H. Each weight vector takes a value from:

Algorithm 2 Main Loop of MOMBI

Require: MOP, set of weight vectors W
Ensure: Approximation set A to the Pareto front
$i \leftarrow 0$
Initialize population P_i
Evaluate population P_i
Calculate reference points $\{\vec{z}^{*}, \vec{z}^{nad}\}$
Execute $R2$ ranking algorithm $(P_i, W, \vec{z}^{*}, \vec{z}^{nad})$
repeat
  Perform tournament selection
  Generate offspring P_i' using variation operators
  Evaluate population P_i'
  Update reference points $\{\vec{z^{*}}, \vec{z}^{nad}\}$
  Execute $R2$ ranking algorithm $(P_i \bigcup P_i', W, \vec{z}^{*}, \vec{z}^{nad})$
  Reduce population $P_{i+1} \leftarrow \{P_i \bigcup P_i'\}$
  $i \leftarrow$i + 1
until termination condition fullfilled

$$\left\{\varepsilon, \frac{1}{H}, \frac{2}{H}, ..., \frac{H}{H}\right\}, \tag{9}$$

where ε is a value close to zero (10^{-4} is recommended), in order to prevent cancellation in the calculations. The total amount of vectors is represented by the combinatorial number C_{m-1}^{H+m-1}.

V. Experimental Results

We compare the performance of our proposed MOMBI[6] with respect to that of two state-of-the-art MOEAs. The first is the multi-objective evolutionary algorithm based on decomposition (MOEA/D)[7][14], which transforms an optimization problem into a number of scalar optimization subproblems that are simultaneously optimized. The second approach is the S Metric Selection-Evolutionary Multiobjective Optimization Algorithm (SMS-EMOA) [4], which is a popular hypervolume-based MOEA. SMS-EMOA adopts non-dominated sorting as its primary selection criterion and the hypervolume contribution as its secondary criterion. Since SMS-EMOA requires a considerably large computational time in problems of high dimensionality (i.e., MOPs having 4 or more objectives) [10], we use here a version that incorporates the algorithm proposed in [11] for estimating the hypervolume using Monte Carlo sampling, instead of the exact hypervolume calculations adopted in the original algorithm [4].

We decided not to compare results with respect to NSGA-II [22], because several studies currently available indicate that MOEA/D is able to outperform it ([27], [28]), and our first experiments corroborated such results.

All the experiments reported here were conducted on identical PCs having Intel(R) Core(TM) i7 processors running at 2.67GHz and with 3.8 GBytes in RAM. The three MOEAs adopted in our comparative study were implemented in C/C++ under Linux, using real-numbers encoding.

[6]The source code and the complete study of MOMBI is available at: http://computacion.cs.cinvestav.mx/~rhernandez/mombi

[7]We used the implementation from 2007 for continuous search spaces: http://dces.essex.ac.uk/staff/zhang/webofmoead.htm

TABLE II. Properties of the test problems.

Problem	Separability	Modality	Geometry
DTLZ1	separable	multi	linear
DTLZ2	separable	uni	concave
DTLZ3	separable	multi	concave
DTLZ4	separable	uni	concave
DTLZ5	unknown	uni	arc, degenerated
DTLZ6	unknown	uni	arc, degenerated
DTLZ7	not applicable separable	$f_{1:m-1}$ uni f_m multi	disconnected, mixed
WFG1	separable	uni	convex, mixed
WFG2	non-separable	$f_{1:m-1}$ uni f_m multi	convex, disconnected
WFG3	non-separable	uni	linear, degenerate
WFG4	separable	multi	concave
WFG5	separable	deceptive	concave
WFG6	non-separable	uni	concave
WFG7	separable	uni	concave
WFG8	non-separable	uni	concave
WFG9	non-separable	multi, deceptive	concave

A. Test problems

For comparison purposes, we adopted the Deb-Thiele-Laumanns-Zitzler [29] and the Walking-Fish-Group [30] test suites. All the minimization problems adopted are scalable with respect to the number of objectives and have a variety of geometries for the Pareto front, such as linear, mixed (concave/convex), degenerate and disconnected. They also include some aspects such as separability and multifrontality which make them more difficult to solve. In Table II we summarize the main features of these test problems [30].

B. Parameters Settings

We performed 30 independent runs of each of three MOEAs compared, in each of the test instances adopted. In DTLZ, the total number of variables is given by $n = m + k - 1$, where m is the number of objectives. k was set to 5 for DTLZ1, 10 for DTLZ2-6 and 20 for DTLZ7. The number of decision variables in WFG was set to 24, and the position-related parameter was set to $m - 1$.

The variation operators adopted in our implementations were: simulated binary crossover (SBX) and polynomial-based mutation [31]. The crossover rate was set to 0.9, while the mutation rate was set to $1/n$. The distribution indexes for both SBX and the polynomial-based mutation were set to 20. The total number of function evaluations was set in such a way that it did not exceed 50,000. The population size and the maximum number of generations adopted in our experiments are shown in Table III, and varied according to the value of m (i.e., number of objectives) adopted.

In MOEA/D and MOMBI, the number of weight vectors is the same as the population size. Following the proposal described in [14], MOEA/D used the Tchebycheff approach with a neighborhood size of 20. The number of samples in SMS-EMOA was set to 10^5.

C. Performance Assessment

For comparing results, we selected the hypervolume indicator, which is equal to the sum of all the rectangular areas, bounded by some reference point. Since this reference point is important, we provide the values that we adopted for each test problem in Table IV. Mathematically, the hypervolume

TABLE III. PARAMETERS.

m	H	Population Size	Generations	Function Evaluations
2	119	120	416	49920
3	14			
4	7			
5	5	126	396	49896
6	4			
7	3	84	595	49980
8		120	416	49920

TABLE IV. REFERENCE POINTS FOR THE TEST INSTANCES.

Test Problem	Reference Point
DTLZ1	$(1, 1, 1, \ldots)$
DTLZ2, DTLZ4	$(2, 2, 2, \ldots)$
DTLZ3	$(7, 7, 7, \ldots)$
DTLZ5	$(4, 4, 4, \ldots)$
DTLZ6	$(11, 11, 11, \ldots)$
DTLZ7	$(21, 21, 21, \ldots)$
WFG	$(3, 5, 7, \ldots, 2m + 1)$

can be described using equation (10) (it is worth noting that higher hypervolume values are preferred):

$$HV(A) = \left\{ \bigcup volume(v) | v \in A \right\}. \quad (10)$$

Additionally, we also considered the running time of each algorithm, measured in seconds (s) or minutes (m). Running times are particularly relevant in this case, since we are interested in analyzing the way in which each of the three MOEAs behaves when increasing the number of objectives, and this includes measuring their computational cost.

D. Discussion of Results

Table V provides the average hypervolume and the average runtime (in parentheses) of each compared MOEA for each instance of the DTLZ test suite. The best results are presented in **boldface** and the second best ones in *italics*. The clear winner in this case is SMS-EMOA, since it was able to outperform the other two MOEAs in 46.9% of the instances adopted. However, this comes at the expense of a computational cost which is considerably higher than the one required by the two other compared MOEAs. On the other hand, MOEA/D presents the lowest running times, but was the winner in only 18.4% of the problems. Our proposed MOMBI represents some sort of intermediate solution, since it was able to outperform the other two approaches in 34.7% of the instances, while requiring reasonably low running times (considerably lower than those required by SMS-EMOA and not much higher than those required by MOEA/D). Remarkably, our proposed MOMBI outperformed the other two compared MOEAs in all instances of DTLZ7. Figure 2 presents a graphical representation of the approximations of the Pareto front obtained by our proposed MOMBI in some of the DTLZ test problems adopted.

In Tables VI and VII, we show the comparison of results for the WFG test suite. Here, again SMS-EMOA outperformed the other two MOEAs in 71.4% of the problems, but requiring a considerably higher CPU time. Our proposed MOMBI ranks second, but requiring less than 3 seconds to solve any of the test instances considered in this case. Remarkably, in this case, our proposed MOMBI outperformed MOEA/D in 95.2% of the test instances considered, while requiring only slightly higher CPU times than this other MOEA. Also, MOMBI outperformed the other two compared MOEAs in all instances of WFG1.

TABLE V. COMPARISON OF RESULTS FOR THE DTLZ TEST PROBLEMS. AVERAGE HYPERVOLUME AND AVERAGE RUNTIME (IN PARENTHESES).

m	MOMBI	MOEA/D	SMS-EMOA
	DTLZ1		
2	**8.737838e-01** (1.50s)	*8.737470e-01* (0.60s)	8.735790e-01 (7.42m)
3	*9.693597e-01* (1.55s)	9.689446e-01 (0.69s)	**9.738787e-01** (7.38m)
4	9.854892e-01 (1.64s)	*9.884462e-01* (0.78s)	**9.924153e-01** (5.02m)
5	*9.919373e-01* (1.83s)	**9.932330e-01** (0.92s)	9.878291e-01 (3.83m)
6	*9.944634e-01* (1.95s)	**9.955037e-01** (0.97s)	9.656273e-01 (4.73m)
7	*9.752993e-01* (1.36s)	**9.865474e-01** (1.03s)	9.533103e-01 (6.20m)
8	**9.845829e-01** (2.13s)	*9.805640e-01* (1.11s)	9.354406e-01 (6.52m)
	DTLZ2		
2	*3.210785e+00* (1.52s)	**3.210866e+00** (0.72s)	3.210667e+00 (20.21m)
3	*7.388812e+00* (1.57s)	7.383274e+00 (0.85s)	**7.427998e+00** (51.74m)
4	1.542186e+01 (1.69s)	*1.542219e+01* (0.90s)	**1.558163e+01** (59.29m)
5	*3.153492e+01* (1.89s)	3.153351e+01 (1.03s)	**3.168830e+01** (74.86m)
6	*6.299770e+01* (2.01s)	6.287641e+01 (1.15s)	**6.375897e+01** (78.05m)
7	*1.226035e+02* (1.44s)	1.219099e+02 (1.15s)	**1.277801e+02** (58.98m)
8	*2.453361e+02* (2.21s)	2.438129e+02 (1.23s)	**2.558339e+02** (93.74m)
	DTLZ3		
2	**4.820292e+01** (1.54s)	*4.820122e+01* (0.67s)	4.819756e+01 (4.32m)
3	*3.423640e+02* (1.58s)	**3.423744e+02** (0.83s)	3.391415e+02 (2.99m)
4	**2.400275e+03** (1.68s)	*2.400109e+03* (0.91s)	1.524083e+03 (3.06m)
5	**1.680506e+04** (1.89s)	*1.680405e+04* (1.00s)	2.303193e+03 (3.27m)
6	*1.175943e+05* (2.01s)	**1.176068e+05** (1.08s)	4.276067e+03 (4.24m)
7	**8.228736e+05** (1.44s)	*8.224488e+05* (1.10s)	4.508892e+04 (4.91m)
8	*5.752984e+06* (2.21s)	**5.756130e+06** (1.21s)	9.710261e+04 (6.15m)
	DTLZ4		
2	**3.089723e+00** (1.54s)	2.565073e+00 (0.66s)	*3.008898e+00* (17.17m)
3	**7.292802e+00** (1.61s)	6.479912e+00 (0.83s)	*7.126184e+00* (57.30m)
4	**1.518143e+01** (1.76s)	1.420544e+01 (0.91s)	*1.509114e+01* (63.68m)
5	**3.128310e+01** (2.01s)	2.868783e+01 (1.03s)	*3.076816e+01* (69.85m)
6	*6.278644e+01* (2.22s)	6.022725e+01 (1.16s)	**6.337518e+01** (74.71m)
7	*1.221849e+02* (1.64s)	1.171185e+02 (1.22s)	**1.272727e+02** (63.35m)
8	*2.443096e+02* (2.52s)	2.419017e+02 (1.38s)	**2.554984e+02** (79.34m)
	DTLZ5		
2	*1.521078e+01* (1.52s)	**1.521085e+01** (0.73s)	1.521065e+01 (20.81m)
3	*5.984359e+01* (1.58s)	5.984289e+01 (0.85s)	**5.986865e+01** (40.89m)
4	*2.392736e+02* (1.68s)	2.387649e+02 (0.92s)	**2.392747e+02** (33.37m)
5	*9.494039e+02* (1.88s)	9.452332e+02 (0.99s)	**9.579889e+02** (47.84m)
6	*3.768958e+03* (2.00s)	3.746660e+03 (1.05s)	**3.834024e+03** (53.24m)
7	*1.494773e+04* (1.43s)	1.492053e+04 (1.09s)	**1.529317e+04** (58.51m)
8	*5.983042e+04* (2.21s)	5.946643e+04 (1.18s)	**6.129900e+04** (73.26m)
	DTLZ6		
2	*1.201014e+02* (1.56s)	1.200361e+02 (0.63s)	**1.201015e+02** (16.00m)
3	*1.317979e+03* (1.61s)	1.316642e+03 (0.77s)	**1.318087e+03** (25.12m)
4	*1.447895e+04* (1.74s)	**1.448415e+04** (0.90s)	1.447694e+04 (24.23m)
5	1.559276e+05 (1.92s)	*1.582229e+05* (1.02s)	**1.592911e+05** (42.63m)
6	1.697309e+06 (2.08s)	*1.732222e+06* (1.07s)	**1.752823e+06** (53.79m)
7	1.869272e+07 (1.47s)	*1.907810e+07* (1.10s)	**1.932404e+07** (48.30m)
8	2.037302e+08 (2.25s)	*2.099493e+08* (1.23s)	**2.121759e+08** (68.94m)
	DTLZ7		
2	**3.915723e+02** (2.34s)	3.872996e+02 (1.28s)	*3.915701e+02* (29.69m)
3	**8.029312e+03** (2.23s)	7.796863e+03 (1.40s)	*7.975963e+03* (40.50m)
4	**1.647751e+05** (2.28s)	1.592594e+05 (1.40s)	*1.622055e+05* (53.12m)
5	**3.368343e+06** (2.64s)	*3.139535e+06* (1.52s)	3.093805e+06 (74.06m)
6	**6.683026e+07** (2.84s)	*6.276078e+07* (1.85s)	5.937294e+07 (86.32m)
7	**1.197319e+09** (2.07s)	*1.177091e+09* (1.72s)	1.107867e+09 (77.17m)
8	**2.461092e+10** (2.88s)	*2.314593e+10* (1.91s)	2.007093e+10 (102.93m)

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have introduced a new multi-objective evolutionary algorithm whose selection mechanism is based on the $R2$ indicator. It is worth emphasizing that the proposed approach is entirely based on the $R2$ indicator, since it does not incorporate Pareto dominance anywhere. Our preliminary experimental results show that our proposed approach is able

TABLE VI. COMPARISON OF RESULTS FOR THE WFG TEST PROBLEMS.

m	MOMBI	MOEA/D	SMS-EMOA
		WFG1	
2	**6.294468e+00** (1.73s)	5.836052e+00 (1.07s)	*6.253419e+00* (6.14m)
3	**5.494206e+01** (1.82s)	*5.169631e+01* (1.22s)	5.069579e+01 (31.06m)
4	**4.700020e+02** (1.92s)	*4.565852e+02* (1.29s)	4.153136e+02 (38.56m)
5	**4.992548e+03** (2.11s)	*4.665130e+03* (1.41s)	4.247997e+03 (44.15m)
6	**6.546410e+04** (2.23s)	*5.654459e+04* (1.41s)	5.448022e+04 (52.01m)
7	**1.050149e+06** (1.64s)	8.222872e+05 (1.43s)	*8.667918e+05* (35.36m)
8	**2.027665e+07** (2.37s)	1.353975e+07 (1.50s)	*1.509819e+07* (43.54m)
		WFG2	
2	*1.096112e+01* (1.64s)	1.045158e+01 (0.98s)	**1.099152e+01** (7.65m)
3	**9.377662e+01** (1.70s)	8.526391e+01 (1.16s)	*9.242151e+01* (13.52m)
4	**9.023867e+02** (1.81s)	7.729761e+02 (1.17s)	*8.664688e+02* (16.27m)
5	**9.927867e+03** (1.99s)	8.599778e+03 (1.26s)	*9.685033e+03* (20.05m)
6	*1.292477e+05* (2.11s)	1.104339e+05 (1.31s)	**1.293052e+05** (15.93m)
7	*1.798632e+06* (1.52s)	1.550014e+06 (1.31s)	**1.826704e+06** (16.16m)
8	*3.115276e+07* (2.32s)	2.824171e+07 (1.41s)	**3.206849e+07** (22.95m)
		WFG3	
2	**1.090903e+01** (1.65s)	1.084768e+01 (1.06s)	*1.088469e+01* (13.50m)
3	**7.533956e+01** (1.69s)	7.343757e+01 (1.13s)	*7.521931e+01* (28.47m)
4	*6.521278e+02* (1.78s)	5.922803e+02 (1.20s)	**6.725993e+02** (37.40m)
5	*6.536085e+03* (1.96s)	5.868854e+03 (1.25s)	**7.388436e+03** (52.33m)
6	*8.169419e+04* (2.07s)	7.015066e+04 (1.24s)	**9.622059e+04** (56.04m)
7	**1.139849e+06** (1.49s)	9.341369e+05 (1.27s)	*1.106727e+06* (85.61m)
8	*1.919048e+07* (2.24s)	1.571596e+07 (1.31s)	**2.346759e+07** (115.11m)
		WFG4	
2	**8.663414e+00** (1.69s)	*8.638790e+00* (1.13s)	8.622096e+00 (16.62m)
3	*7.428724e+01* (1.76s)	7.367783e+01 (1.26s)	**7.655320e+01** (32.23m)
4	*6.899212e+02* (1.88s)	6.683063e+02 (1.30s)	**7.565462e+02** (40.65m)
5	*7.963947e+03* (2.06s)	7.477385e+03 (1.37s)	**8.613500e+03** (40.79m)
6	*9.172554e+04* (2.18s)	8.358673e+04 (1.42s)	**1.116341e+05** (52.17m)
7	*1.114922e+06* (1.68s)	9.895873e+05 (1.41s)	**1.551352e+06** (47.11m)
8	*1.840730e+07* (2.36s)	1.563104e+07 (1.51s)	**2.832567e+07** (63.59m)
		WFG5	
2	**8.208059e+00** (1.65s)	8.135799e+00 (1.05s)	*8.165422e+00* (21.52m)
3	*7.111670e+01* (1.71s)	6.980779e+01 (1.15s)	**7.337062e+01** (29.56m)
4	*6.676905e+02* (1.92s)	6.374582e+02 (1.20s)	**7.323117e+02** (30.40m)
5	*7.690169e+03* (2.00s)	7.402440e+03 (1.34s)	**8.421103e+03** (29.38m)
6	*9.592303e+04* (2.11s)	9.477511e+04 (1.39s)	**1.097473e+05** (38.55m)
7	1.061202e+06 (1.54s)	*1.085262e+06* (1.44s)	**1.576659e+06** (40.72m)
8	*1.742656e+07* (2.32s)	1.720052e+07 (1.50s)	**2.749614e+07** (49.43m)

TABLE VII. COMPARISON OF RESULTS FOR THE WFG TEST PROBLEMS (CONTINUATION).

m	MOMBI	MOEA/D	SMS-EMOA
		WFG6	
2	**8.366185e+00** (1.75s)	*8.339773e+00* (1.20s)	8.335931e+00 (15.04m)
3	*7.180386e+01* (1.78s)	7.090983e+01 (1.28s)	**7.396142e+01** (27.41m)
4	*6.905283e+02* (1.87s)	6.335617e+02 (1.27s)	**7.273866e+02** (26.21m)
5	*7.992895e+03* (2.04s)	7.166289e+03 (1.31s)	**8.453112e+03** (24.98m)
6	*9.657006e+04* (2.15s)	8.711648e+04 (1.35s)	**1.114382e+05** (26.89m)
7	1.069256e+06 (1.61s)	*1.094029e+06* (1.37s)	**1.484532e+06** (32.95m)
8	*1.720793e+07* (2.38s)	1.568331e+07 (1.42s)	**2.804717e+07** (36.11m)
		WFG7	
2	**8.676485e+00** (1.64s)	*8.665153e+00* (1.07s)	8.659994e+00 (25.32m)
3	*7.495940e+01* (1.72s)	7.376198e+01 (1.22s)	**7.689421e+01** (33.91m)
4	*7.207621e+02* (1.84s)	6.801568e+02 (1.29s)	**7.626003e+02** (35.63m)
5	*8.374343e+03* (2.04s)	7.683882e+03 (1.42s)	**8.907564e+03** (30.54m)
6	*1.013138e+05* (2.17s)	9.488079e+04 (1.47s)	**1.156273e+05** (34.36m)
7	*1.126400e+06* (1.66s)	1.011440e+06 (1.52s)	**1.653699e+06** (43.30m)
8	*1.844738e+07* (2.42s)	1.533142e+07 (1.64s)	**2.943970e+07** (48.29m)
		WFG8	
2	**8.081846e+00** (1.79s)	*8.070813e+00* (1.33s)	8.058860e+00 (7.15m)
3	*6.842167e+01* (1.83s)	6.806646e+01 (1.44s)	**7.021008e+01** (19.32m)
4	*5.851432e+02* (1.93s)	5.489519e+02 (1.47s)	**6.862165e+02** (27.23m)
5	*5.010892e+03* (2.10s)	4.748502e+03 (1.49s)	**7.782572e+03** (38.25m)
6	*5.680393e+04* (2.19s)	4.633819e+04 (1.53s)	**9.848530e+04** (56.00m)
7	*7.871659e+05* (1.66s)	5.748249e+05 (1.53s)	**1.318850e+06** (57.39m)
8	*1.299050e+07* (2.38s)	9.495123e+06 (1.51s)	**2.461203e+07** (77.00m)
		WFG9	
2	*8.234320e+00* (1.99s)	8.065763e+00 (1.64s)	**8.252136e+00** (23.99m)
3	6.712947e+01 (1.99s)	*6.785035e+01* (1.74s)	**7.108843e+01** (39.76m)
4	*5.889761e+02* (2.08s)	5.654998e+02 (1.76s)	**6.930176e+02** (40.48m)
5	*5.852350e+03* (2.25s)	5.585894e+03 (1.82s)	**7.684327e+03** (41.68m)
6	*6.393249e+04* (2.48s)	5.933684e+04 (1.85s)	**1.017342e+05** (48.38m)
7	*6.670261e+05* (1.84s)	6.585370e+05 (1.84s)	**1.519268e+06** (43.50m)
8	*1.078788e+07* (2.61s)	1.002607e+07 (1.93s)	**2.583169e+07** (63.17m)

to outperform MOEA/D in most cases and that it requires a considerably lower computational cost than SMS-EMOA in all cases. Also, we hypothesize that our proposed MOMBI outperforms SMS-EMOA and MOEA/D in problems in which the Pareto front is mixed or disconnected.

Evidently, our results are only preliminary and much more work is required. We are interested, for example, in studying the sensitivity of our proposed approach to the reference set and in incorporating a mechanism to handle constraints. It would also be interesting to combine this indicator with another one (e.g., Δ_p), with the aim of combining their advantages and compensate for their possible limitations. Since in many-objective optimization the target is not only convergence and distribution, but also pertinency[8] [32], we are intent on integrating preference information into MOMBI.

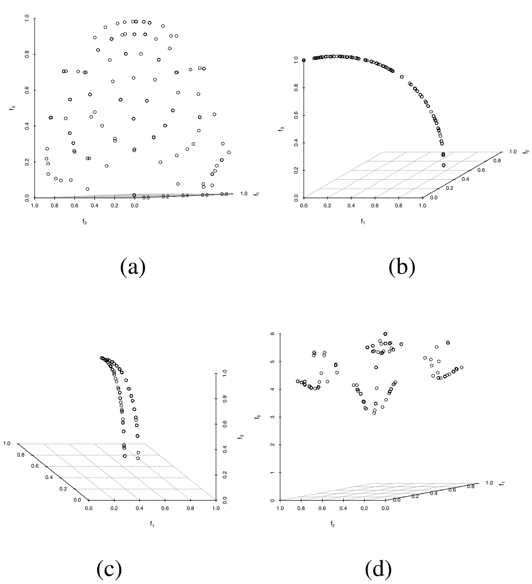

Fig. 2. Plots of the approximations obtained by our proposed MOMBI for $m = 3$ in: (a) DTLZ2, (b) DTLZ5, (c) DTLZ6 and (d) DTLZ7. These plots correspond to the mean hypervolume value from 30 independent runs.

ACKNOWLEDGMENTS

The first author acknowledges support from CINVESTAV-IPN and CONACyT to pursue graduate studies in Computer Science at CINVESTAV-IPN. The second author gratefully acknowledges support from CONACyT project no. 103570.

[8]Property in which a MOEA focuses on solutions that are inside the designer's region of interest

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] J. Knowles and D. Corne, "Quantifying the Effects of Objective Space Dimension in Evolutionary Multiobjective Optimization," 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. 757–771, Springer. Lecture Notes in Computer Science Vol. 4403, March 2007.

[3] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *2008 Congress on Evolutionary Computation (CEC'2008)*, (Hong Kong), pp. 2424–2431, IEEE Service Center, June 2008.

[4] 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.

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

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

[7] M. Fleischer, "The Measure of Pareto Optima. Applications to Multiobjective 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.

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

[9] K. Bringmann and T. Friedrich, "Don't be Greedy when Calculating Hypervolume Contributions," in *FOGA '09: Proceedings of the tenth ACM SIGEVO workshop on Foundations of genetic algorithms*, (Orlando, Florida, USA), pp. 103–112, ACM, January 2009.

[10] C. A. Rodríguez Villalobos and C. A. Coello Coello, "A New Multi-Objective Evolutionary Algorithm Based on a Performance Assessment Indicator," in *2012 Genetic and Evolutionary Computation Conference (GECCO'2012)*, (Philadelphia, USA), pp. 505–512, ACM Press, July 2012. ISBN: 978-1-4503-1177-9.

[11] J. Bader and E. Zitzler, "HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization," *Evolutionary Computation*, vol. 19, no. 1, pp. 45–76, Spring, 2011.

[12] K. Bringmann and T. Friedrich, "Convergence of Hypervolume-Based Archiving Algorithms II: Competitiveness," in *2012 Genetic and Evolutionary Computation Conference (GECCO'2012)*, (Philadelphia, USA), pp. 457–464, ACM Press, July 2012. ISBN: 978-1-4503-1177-9.

[13] D. Brockhoff, T. Wagner, and H. Trautmann, "On the Properties of the $R2$ Indicator," in *2012 Genetic and Evolutionary Computation Conference (GECCO'2012)*, (Philadelphia, USA), pp. 465–472, ACM Press, July 2012. ISBN: 978-1-4503-1177-9.

[14] 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.

[15] E. Zitzler, D. Brockhoff, and L. Thiele, "The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicator Via Weighted Integration," 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. 862–876, Springer. Lecture Notes in Computer Science Vol. 4403, March 2007.

[16] L. While, "A New Analysis of the LebMeasure Algorithm for Calculating Hypervolume," in *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005* (C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, eds.), (Guanajuato, México), pp. 326–340, Springer. Lecture Notes in Computer Science Vol. 3410, March 2005.

[17] C. Igel, N. Hansen, and S. Roth, "Covariance Matrix Adaptation for Multi-objective Optimization," *Evolutionary Computation*, vol. 15, pp. 1–28, Spring 2007.

[18] O. Schütze, X. Esquivel, A. Lara, and C. A. Coello Coello, "Using the Averaged Hausdorff Distance as a Performance Measure in Evolutionary Multiobjective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, pp. 504–522, August 2012.

[19] 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.

[20] C. A. Coello Coello and N. Cruz Cortés, "Solving Multiobjective Optimization Problems using an Artificial Immune System," *Genetic Programming and Evolvable Machines*, vol. 6, pp. 163–190, June 2005.

[21] K. Gerstl, G. Rudolph, O. Schütze, and H. Trautmann, "Finding Evenly Spaced Fronts for Multiobjective Control via Averaging Hausdorff-Measure," in *The 2011 8th International Conference on Electrical Engineering, Computer Science and Automatic Control (CCE'2011)*, (Mérida, Yucatán, México), pp. 975–980, IEEE Press, October 2011.

[22] 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.

[23] H. Trautmann, G. Rudolph, C. Dominguez-Medina, and O. Schütze, "Finding Evenly Spaced Pareto Fronts for Three-Objective Optimization Problems," in *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II* (O. Schütze, C. A. Coello Coello, A.-A. Tantar, E. Tantar, P. Bouvry, P. Del Moral, and P. Legrand, eds.), pp. 89–105, Berlin, Germany: Springer, Advances in Intelligent Systems and Computing Vol. 175, 2012. ISBN 978-3-642-31519-0.

[24] M. P. Hansen, *Metaheuristics for multiple objective combinatorial optimization*. PhD thesis, Institute of Mathematical Modelling, Technical University of Denmark, March 1998.

[25] J. Knowles and D. Corne, "On Metrics for Comparing Nondominated Sets," in *Congress on Evolutionary Computation (CEC'2002)*, vol. 1, (Piscataway, New Jersey), pp. 711–716, IEEE Service Center, May 2002.

[26] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, Massachusetts: Addison-Wesley Publishing Company, 1989.

[27] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 13, pp. 284–302, April 2009.

[28] K. Li, S. Kwong, J. Cao, M. Li, J. Zheng, and R. Shen, "Achieving balance between proximity and diversity in multi-objective evolutionary algorithm," *Information Sciences*, vol. 182, pp. 220–242, January 1 2012.

[29] 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.

[30] S. Huband, P. Hingston, L. Barone, and L. While, "A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, pp. 477–506, October 2006.

[31] K. Deb and R. B. Agrawal, "Simulated Binary Crossover for Continuous Search Space," *Complex Systems*, vol. 9, pp. 115–148, 1995.

[32] P. Fleming, R. C. Purshouse, and R. J. Lygoe, "Many-Objective Optimization: An Engineering Design Perspective," in *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005* (C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, eds.), (Guanajuato, México), pp. 14–32, Springer. Lecture Notes in Computer Science Vol. 3410, March 2005.