A Ranking Method Based on the R2 indicator for Many-Objective Optimization

Alan Díaz-Manríquez*, Gregorio Toscano-Pulido*, Carlos A. Coello Coello†, Ricardo Landa-Becerra*
*Information Technology Laboratory, CINVESTAV-Tamaulipas.
Parque Científico y Tecnológico TECNOTAM Km. 5.5 carretera Cd. Victoria-Soto La Marina.
Cd. Victoria, Tamaulipas 87130, MÉXICO
†CINVESTAV-IPN. Departamento de Computación. Av. IPN No. 2508.
Col. San Pedro Zacatenco, México D.F., 07360, MEXICO

***Abstract*—In recent years, the development of selection mechanisms based on performance indicators has become an important trend in algorithmic design. Hereof, the hypervolume has been the most popular choice. Multi-objective evolutionary algorithms (MOEAs) based on this indicator seem to be a good choice for dealing with many-objective optimization problems. However, their main drawback is that such algorithms are typically computationally expensive. This has motivated some recent research in which the use of other performance indicators has been explored. Here, we propose an efficient mechanism to integrate the $R2$ indicator to a modified version of Goldberg's non-dominated sorting method, in order to rank the individuals of a MOEA. Our proposed ranking scheme is coupled to two different search engines, resulting in two new MOEAs. These MOEAs are validated using several test problems and performance measures commonly adopted in the specialized literature. Results indicate that the proposed ranking approach gives rise to effective MOEAs, which produce results that are competitive with respect to those obtained by three well-known MOEAs. Additionally, we validate our resulting MOEAs in many-objective optimization problems, in which our proposed ranking scheme shows its main advantage, since it is able to outperform a hypervolume-based MOEA, requiring a much lower computational time.**

I. Introduction

Multiobjective Evolutionary Algorithms (MOEAs) based on *Pareto Dominance* (PD) have been successfully used to solve problems with two or three objectives, for several years. However, it is well-known that, as the number of objectives increases, the proportion of non-dominated solutions increases in an exponential way [1]–[3]. Therefore, very quickly, it becomes impossible to distinguish individuals for selection purposes and the selection pressure dilutes, since we are practically selecting solutions in a random way.

To overcome this shortcoming in the fitness assignment process [4], the evolutionary multiobjective optimization (EMO) community has developed several approaches that drive the search using a quality assessment indicator. This idea has become more popular in the last few years, mainly because of the growing interest in tackling multi-objective problems having 4 or more objectives (commonly called "many-objective optimization problems"), for which indicator-based MOEAs seem to be particularly suitable [5]. When using indicator-based selection, the idea is to identify the solutions that contribute the most to the improvement of the performance indicator adopted in the selection mechanism.

The most general version of an algorithm of this sort is the Indicator Based Evolutionary Algorithm (IBEA) proposed by Zitzler and Künzli [6]. In this case, the original problem is replaced by the minimization or maximization of a performance indicator. The authors proposed two different versions, one using the Hypervolume [7], and another one adopting the ϵ-indicator. In this work, the authors showed that with the use of an indicator to rank the solutions, the algorithm did not require any additional diversity preservation mechanism. Beume *et al.* proposed the S Metric Selection Evolutionary Multi-Objective Algorithm (SMS-EMOA) [8]. This algorithm uses the NSGA-II's selection procedure. However, the authors replaced the Crowding Distance with the Hypervolume indicator. Ishibuchi *et al.* proposed an approach divided in two phases [9]. In the first phase, each objective is optimized separately, while in the second phase, the algorithm searches for the solution with more contribution to the Hypervolume indicator. This approach was designed to search for a small number of non-dominated solutions along the entire Pareto front. Igel *et al.* proposed the Multi-Objective Covariance Matrix Adaptation Evolution Strategy (MO-CMA-ES) [10]. This algorithm uses a set of single-objective optimizers as its population. Each optimizer generates new solutions that can be accepted back into the population according to two criteria: 1) their ranking according to PD, or 2) their contribution to the Hypervolume. This work reveals that MO-CMA-ES is reliable and is faster than the standard CMA-ES.

As it turns out, the Hypervolume has become the most popular choice for implementing indicator-based MOEAs, mainly because of its nice theoretical properties [8]. The use of the Hypervolume raises, however, some issues, since it is very expensive (computationally speaking), and its computational cost considerably increases as we increase the number of objectives. One choice is to approximate the hypervolume (see for example [5]). Although this sort of scheme can considerably reduce the computational cost of the ranking process, one would expect that this will also decrease (perhaps in a significant manner) the accuracy of the selection mechanism. Rodriguez-Villalobos *et al.* recently proposed the DDE (Δ_p-Differential Evolution) [11] in which the authors adopted another performance indicator as an alternative to the Hypervolume. In this approach, the Δ_p indicator [12] is adopted. The fitness assignment for each solution is performed through the contribution to Δ_p. The Δ_p indicator requires a reference set to be calculated, and the authors used the nadir point and the ideal vector to create such a reference set. The authors reported that this approach could obtain competitive results with respect to other MOEAs (including SMS-EMOA), having as its main advantage its very low computational cost, even when dealing with many-objective problems.

Recent works [13, 14] have reported another suitable performance indicator: $R2$ [15]. This performance indicator has desirable properties (i.e., it is weakly monotonic, it produces well-distributed solutions and it can be computed in a fast manner) that make it a viable candidate to be used into an indicator-based MOEA. These works also compare the behavior of $R2$ with respect to that of the Hypervolume, and conclude that they behave in a similar way, but $R2$ has a considerably lower computational cost.

In this work, we propose to use the $R2$ indicator to rank the individuals of a MOEA. The proposed approach is embedded into two different algorithms (a Genetic Algorithm and Differential Evolution). The new algorithms are then compared with respect to some state-of-the-art MOEAs taken from the specialized literature. Furthermore, we present a scalability study to analyze the behavior of our proposed approach as the number of objectives of the problem increases. The remainder of this work is organized as follows. Details of the $R2$ indicator are given in Section II. Section III presents our proposed ranking method as well as the two new resulting MOEAs[1]. A comparative study with respect to other algorithms is presented in Section IV. Finally, our conclusions and some possible paths for future work are provided in Section V.

II. $R2$ INDICATOR

The family of R indicators [15] is based on utility functions which map a vector $\vec{y} \in \mathcal{R}^k$ to a scalar value $u \in \mathcal{R}$ in order to measure the quality of two approximations of the Pareto front.

Definition 1: For a set U of utility functions, a probability distribution p on U, and a reference set R, the $R2$ indicator of a solution set A is defined as:

$$R2(R, A, U, p) = \int_{u \in U} \max_{r \in R}\{u(r)\}p(u)du - \int_{u \in U} \max_{a \in A}\{u(a)\}p(u)du \quad (1)$$

Definition 2: For a discrete and finite set U and a uniform distribution p over U, the $R2$ indicator can be defined as [16]:

$$R2(R, A, U) = \frac{1}{|U|}\sum_{u \in U}\left(\max_{r \in R}\{u(r)\} - \max_{a \in A}\{u(a)\}\right) \quad (2)$$

Since the first summand ($\max_{r \in R}\{u(r)\}$) is constant if we assume a constant R, the first summand can be deleted in order to have a unary indicator as a result (called R for simplicity) [14].

Definition 3: For a constant reference set, the $R2$ indicator can be defined as a unary indicator as follows:

$$R2(A, U) = -\frac{1}{|U|}\sum_{u \in U}\{u(a)\} \quad (3)$$

[1]The source code of these approaches is available for download from http://www.tamps.cinvestav.mx/~adiazm/

We use the Tchebycheff function as our utility function in our approach. This function works well when optimizing different types of Pareto fronts. It is worth noting that this aggregation function is not smooth for continuous multi-objective problems. However, our algorithm does not need to compute the derivative of the aggregation function. The Tchebycheff function can be defined as $u(z) = u_\lambda(\vec{z}) = -\max_{j \in \{1,\ldots,k\}} \lambda_j |z_j^* - z_j|$ where $\lambda = (\lambda_1, \ldots, \lambda_k) \in \Lambda$ is a weight vector and z^* is an utopian point[2].

Definition 4: The $R2$ indicator of a solution set A for a given set of weight vectors Λ and a utopian point z^* is defined as:

$$R2(A, \Lambda, z^*) = \frac{1}{\Lambda}\sum_{\lambda \in \Lambda} \min_{a \in A}\{\max_{j \in \{1,\ldots,k\}}\{\lambda_j |z_j^* - a_j|\}\} \quad (4)$$

Definition 5: Finally, we say that the contribution of one solution $a \in A$ to the $R2$ indicator can be defined as:

$$C_{R2}(a, A, \Lambda, z^*) = R2(A, \Lambda, z^*) - R2(A\backslash\{a\}, \Lambda, z^*) \quad (5)$$

III. OUR PROPOSED APPROACH

It is well known that Goldberg's non-dominated sorting scheme [17] has been adopted in a number of MOEAs (including indicator-based algorithms). It is worth noting, however, that the integration of this sort of scheme with an indicator-based selection mechanism can lead to very expensive algorithms (computationally speaking). Therefore, we decided to completely remove Pareto dominance of this sorting approach and replace it by the $R2$ indicator. The idea of this change was to produce a computationally efficient algorithm, particularly when dealing with many-objective problems. Below, we explain the procedure adopted to identify the solutions that contribute the most to $R2$.

A. Fast $R2$ Sorting Approach

Notwithstanding the straightforward way to compute the contribution of a solution $a \in A$ to $R2$ (where A is the approximated Pareto front) is the issue of how to measure $R2$ for A and then remove the solution a from A evaluating only the new value of the indicator $R2$ (this is known as the contribution of a). This is, indeed a computationally expensive procedure. Therefore, we looked for an alternative mechanism. First, we identify the individuals that contribute the most to the performance indicator. The $R2$ indicator relies on weight vectors, and the approach consists in computing the contribution of each weight vector $\lambda \in \Lambda$ to $R2$ as follows:

$$C_\lambda^{R2} = \min_{a \in A}\{\max_{j \in \{1,\ldots,k\}}\{\lambda_j |z_j^* - a_j|\}\} \quad (6)$$

Since $\sum_{\lambda \in \Lambda} C_\lambda^{R2}$ is equal to the $R2$ indicator, it is possible to identify which solutions contribute the most to $R2$, considering that it is also possible to obtain the individual

[2]An objective vector that is not dominated by any feasible search point

which contributes to each C_λ^{R2}. Thus, an individual will have a contribution equal to the sum of the C_λ^{R2} where the individual contributes. These findings led us to propose a pseudocode to compute the contribution of each solution to the indicator $R2$ which is shown in Algorithm 1 .

Algorithm 1 $C_{R2}(A, \Lambda, z^*)$

Input: A Approximated Pareto front
Output: C Contributions of the $|A|$ solutions
$\forall a \in A : C_a = 0$
for $\lambda \in \Lambda$ **do**
$[C_\lambda^{R2}\ a] = \min_{a \in A}\{\max_{j \in \{1,\ldots,k\}}\{\lambda_j |z_j^* - a_j|\}\}$
$\{a$ is the solution which contributes to $C_\lambda^{R2}\}$
$C_a = C_a + C_\lambda^{R2}$
end for

Having the contribution of each individual to the $R2$ indicator, we can easily rank the individuals with respect to these contributions. However, it is important to notice that, since we can have individuals with zero contribution to $R2$, then it is also possible to have fewer solutions than those needed to complete the required population size. In order to deal with this problem, we decided to adopt the notion of layers. The general idea of the proposed approach is the following: first, we apply the $R2$ indicator to the population. Then, we identify and isolate the individuals with a contribution greater than zero. These individuals will form our first layer (rank 1). Then, we apply again the $R2$ indicator to the remainder of the population (to those individuals that have a contribution of zero in the previous step), identifying and isolating again the individuals with a contribution greater than zero, and assigning them the next ranks. This process is repeated until the entire population obtains a rank. This approach is called the Fast $R2$ Sorting or $R2_{ranking}$ for short.

B. Proposed MOEAs

The algorithm begins with an initial population size of N. This population is randomly initialized. The utopian point is formed with respect to the best obtained values for each objective (of this initial population). Since, in order to compute the $R2$ indicator, it is necessary to have a set of uniformly distributed weight vectors, we randomly initialized the weight vectors in such a way that the sum of each weight vector is equal to one [3]. The weights are changed at each iteration of the algorithm. After new solutions are generated through variation operators, the utopian point is updated. The actualization of the utopian point consists in verifying, for each newly evaluated solution, if they have an objective value better than the current utopian point. Finally, the Fast $R2$ Sorting approach is executed in order to select the best N solutions that will constitute the next generation. This is the general structure for our $R2$-based MOEA's proposals. We show the pseudocode of our general MOEA in Algorithm 2. Two different approaches are derived from this general MOEA. The first approach resembles a genetic algorithm similar to the NSGA-II since it uses its same variation operators: Simulated Binary Crossover (SBX) [19] and Polynomial-based Mutation [20]. This approach is called $R2$-MOGA. The second approach uses the Differential Evolution recombination operator to generate solutions, following the DE/rand/1/bin model [21]. This approach is called $R2$-MODE.

[3]The random weights vector was generated as in MOMHLib++ [18]

Algorithm 2 $R2$-MOEA

Input: $Iters$ - Number of iterations, $|\Lambda|$ - Number of weight vectors, N - Size of the population, k - Number of objectives
Output: P_{Iters}
P_1 - Initialize population of size N
$z^* \Leftarrow$ Upgrade_utopian_point(P_1)
for $i = 1$ a $Iters$ **do**
$\Lambda \Leftarrow$ Generate_Weights_Vectors(Λ, k)(Generate $|\Lambda|$ weight vector of size k)
$Q \Leftarrow$ New_Solutions(P_i)
$z^* \Leftarrow$ Upgrade_utopian_point(Q)
$P_{i+1} \Leftarrow R2_{ranking}(P_i \cup Q, N, \Lambda, z^*)$(Ranking with $R2_{ranking}$ and choose the best N to advance to the next generation)
end for

IV. Performance Assessment

The proposed approaches were evaluated using 13 test functions. Five functions were taken from the Zitzler-Deb-Thiele (ZDT) test suite [22], four functions were taken from the Deb-Thiele-Laumanns-Zitzler (DTLZ) test suite [23] and the remaining were taken from the Walking-Fish-Group (WFG) test suite [24]. The main features of these test problems are shown in Table I.

Test problem	# of decision variables	# of objectives
ZDT1-3	30	2
ZDT4,6	10	2
DTLZ1	7	3
DTLZ2-4	12	3
WFG1-4	24	3

TABLE I
Test problems adopted

In order to assess the performance of the proposed approaches, we decided to take two performance measures from the specialized literature.

1) **Hypervolume (HV)**. HV computes the area covered for all the solutions in the approximated Pareto front Q with the help of a reference point W. Equation (7) shows the mathematical definition of HV.

$$HV = volume(\bigcup_{i=1}^{|Q|} v_i) \tag{7}$$

where, for each solution $i \in Q$, a hypercube v_i is constructed using the reference point W. Therefore, HV is the union of the volume of all the hypercubes.

2) **Two set coverage ($C(A, B)$)**. This indicator computes the proportion of solutions in B which are dominated for the solutions in A. Equation (8) refers to the mathematical definition of the two set coverage.

$$C(A, B) = \frac{|\{b \in B | \exists a \in A : a \preceq b\}|}{|B|} \tag{8}$$

A value of $C(A, B) = 1$ indicates that all the solutions in B are dominated by any solution in A. Moreover, a value of $C(A, B) = 0$ indicates that there does not exist any solution in B which is dominated by any solution in A.

We decided to perform two experiments in order to evaluate the effectiveness of our proposals. The first experiment consists of evaluating the performance of our MOEAs when dealing with problems having two and three objectives. However, since two- and three-objectives problems have been widely studied and we think that many-objective problems will be the strength of our approaches, we only expect to have competitive results with respect to other approaches which are representative of the state-of-the-art, in this case. The second experiment is focused on evaluating the performance of our MOEAs in many-objective problems. For this sake, we decided to compare our results with respect to one of the most successful indicator-based approaches currently available: SMS-EMOA.

A. *Experiment 1 (two and three objectives)*

In order to make a comparative study we chose the three following approaches: 1) NSGA-II (this is, by far, the most popular Pareto-based MOEA), 2) MOEA/D (a more recent MOEA, based on decomposition, which has been found to be more effective than NSGA-II in a number of problems), and 3) SMS-EMOA (this is perhaps the most popular indicator-based MOEA in use today). All these MOEAs adopted a population size of 100 (except for MOEA/D in problems with 3 objectives where the population size was of 105) and ran for 200 generations for problems with 2 objectives and ran for 300 generations for problems with 3 objectives. The only exception was DTLZ3 for which the number of generations was set to 1000 (due to the difficulty of this problem). The remainder parameters for each algorithm were the ones suggested by their authors. Table II summarizes the parameters adopted in our comparative study. Finally, in order to have more confident results, each MOEA was executed 30 times.

NSGA-II	MOEA/D	SMS-EMOA	$R2$-MOGA	$R2$-MODE
$pc = 1.0$ $pm = \frac{1}{\lvert n \rvert}$ $nc = 15$ $nm = 15$	$pc = 1.0$ $pm = \frac{1}{\lvert n \rvert}$ $nc = 15$ $nm = 20$ $T = 20$	$pc = 1.0$ $pm = \frac{1}{\lvert n \rvert}$ $nc = 15$ $nm = 20$	$pc = 1.0$ $pm = \frac{1}{\lvert n \rvert}$ $nc = 15$ $nm = 20$	F=0.5 CR=0.5

TABLE II
ADOPTED PARAMETERS FOR EACH MOEA

The application of the Hypervolume performance measure to the results obtained by the five approaches is shown in Table III. From these results, it is easy to see that all the approaches behaved similarly. The NSGA-II slightly outperformed the others for the ZDT test problems. However, this approach did not behave well when optimizing the three-objective test problems. In the DTLZ test problems, SMS-EMOA outperformed the other approaches, while our proposed $R2$-MODE behaved reasonably well in DTLZ4, WFG1 and WFG2. In this experiment, our proposed $R2$-MOGA proposal did not outperform any of its contenders. However, its results are competitive with respect to the results obtained by the other algorithms.

	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
		A=NSGA-II, B=$R2$-MOGA			
C(A,B)	0.438	0.482	**0.966**	0.509	0.164
C(B,A)	**0.513**	**0.601**	0.113	**0.877**	**0.985**
		A=MOEA/D, B=$R2$-MOGA			
C(A,B)	**0.840**	**0.960**	**0.800**	0.002	**0.999**
C(B,A)	0.142	0.134	0.047	**0.985**	0.037
		A=SMS-EMOA, B=$R2$-MOGA			
C(A,B)	**0.830**	**0.873**	**0.989**	0.529	0.006
C(B,A)	0.114	0.132	0.031	**0.793**	**0.985**
		A=NSGA-II, B=$R2$-MODE			
C(A,B)	0.064	0.130	**0.853**	**0.847**	0.000
C(B,A)	**0.776**	**0.865**	0.224	0.819	**0.980**
		A=MOEA/D, B=$R2$-MODE			
C(A,B)	**0.484**	**0.870**	**0.711**	0.832	0.000
C(B,A)	0.357	0.168	0.090	**0.996**	**0.870**
		A=SMS-EMOA, B=$R2$-MODE			
C(A,B)	0.324	**0.542**	**0.939**	**0.847**	0.000
C(B,A)	**0.436**	0.398	0.066	0.822	**0.977**

TABLE IV
COMPARISON OF THE RESULTS OBTAINED BY NSGA-II, MOEA/D AND SMS-EMOA VS. $R2$-MOGA AND $R2$-MODE USING THE TWO SET COVERAGE PERFORMANCE MEASURE FOR THE ZDT TEST SUITE: C(A,B)=% OF SOLUTIONS IN B DOMINATED BY A

Since performing a visual comparison can also be helpful sometimes, we decided to plot the median of the executions with respect to the Hypervolume performance measure. The obtained Pareto fronts are shown in Figures 1 and 2. From these figures, it can be noticed that our proposals did not completely cover the true Pareto front of some test problems (ZDT{2,3,4,6} and WFG{3,4}). However, that is not a problem related to our implementations, but to the $R2$ performance indicator itself. As Brockoff *et al.* [14] have stated, the $R2$ indicator has a tendency towards the knee of the Pareto front. However, and although this condition has a negative impact in our results, our approaches remained competitive with respect to the Hypervolume performance measure. In ZDT4 and WFG1, our $R2$-MODE had problems to converge to the true Pareto front. However, our proposed $R2$-MOGA achieved a good convergence. This led us to hypothesize that the main problems are caused by the search engine and not by the ranking method.

Thus, we decided to adopt a second performance measure so that we could be able to reach more general conclusions. Two Set Coverage is a binary performance measure that has been widely used in the specialized literature. Tables IV and V show the results of the applications of this performance measure to the algorithms' outputs. The results show that SMS-EMOA outperformed our $R2$-MODE in the test problems with three objectives. However, our $R2$-MOGA remained competitive.

B. *Experiment 2 (many-objective problems)*

Since one of the aims of using an indicator-based MOEA is its capability to perform well in the presence of many objective

Problem	NSGA-II	MOEA/D	SMS-EMOA	$R2$-MOGA	$R2$-MODE
ZDT1	**120.642±0.00**	120.331±0.46	120.617±0.01	120.097±0.15	120.221±0.18
ZDT2	**120.275±0.01**	119.903±0.78	120.239±0.03	119.622±0.22	119.899±0.23
ZDT3	128.642±0.59	128.629±0.58	**128.751±0.00**	128.250±0.24	128.420±0.18
ZDT4	**120.317±0.40**	118.436±1.36	117.643±1.65	119.778±0.56	117.931±1.88
ZDT6	116.092±0.04	**117.377±0.04**	115.945±0.11	115.818±0.13	115.929±0.13
DTLZ1	0.168±0.04	0.184±0.00	**0.189±0.00**	0.182±0.00	0.185±0.00
DTLZ2	0.694±0.01	0.710±0.00	**0.740±0.00**	0.712±0.00	0.729±0.00
DTLZ3	0.693±0.01	0.705±0.00	**0.741±0.00**	0.715±0.01	0.730±0.00
DTLZ4	0.644±0.15	0.712±0.00	0.654±0.16	0.715±0.05	**0.722±0.01**
WFG1	14.693±3.35	7.987±0.96	19.756±1.46	17.212±3.04	**24.670±1.44**
WFG2	34.009±4.14	31.783±3.13	36.220±4.64	33.937±4.21	**44.031±3.18**
WFG3	5.018±0.04	4.613±0.19	**5.054±0.03**	3.983±0.27	4.702±0.14
WFG4	15.220±0.81	20.469±0.15	**22.947±0.27**	19.203±0.33	19.780±0.34

TABLE III

COMPARISON OF THE RESULTS OBTAINED BY NSGA-II, MOEA/D, SMS-EMOA, $R2$-MOGA AND $R2$-MODE WITH RESPECT TO THE HYPERVOLUME PERFORMANCE MEASURE, FOR THE ZDT, DTLZ AND WFG TEST SUITES

	DTLZ1	DTLZ2	DTLZ3	DTLZ4	WFG1	WFG2	WFG3	WFG4
	A=NSGA-II, B=$R2$-MOGA							
C(A,B)	0.002	0.062	0.029	0.109	0.402	0.035	0.027	0.030
C(B,A)	**0.397**	**0.323**	**0.219**	**0.376**	**0.821**	**0.503**	**0.617**	**0.327**
	A=MOEA/D, B=$R2$-MOGA							
C(A,B)	0.004	**0.040**	**0.028**	**0.053**	0.000	0.064	0.041	**0.104**
C(B,A)	**0.028**	0.000	0.018	0.000	**1.000**	**0.494**	**0.696**	0.026
	A=SMS-EMOA, B=$R2$-MOGA							
C(A,B)	**0.034**	**0.138**	**0.118**	**0.206**	**0.454**	**0.153**	0.134	0.107
C(B,A)	0.002	0.065	0.017	0.094	0.220	0.146	**0.229**	**0.108**
	A=NSGA-II, B=$R2$-MODE							
C(A,B)	0.028	0.107	0.108	0.159	**0.411**	0.030	0.105	**0.192**
C(B,A)	**0.386**	**0.275**	**0.150**	**0.277**	0.049	**0.287**	**0.637**	0.182
	A=MOEA/D, B=$R2$-MODE							
C(A,B)	0.024	**0.068**	0.049	**0.085**	0.000	0.037	0.075	**0.330**
C(B,A)	**0.231**	0.000	**0.294**	0.000	**1.000**	**0.258**	**0.639**	0.004
	A=SMS-EMOA, B=$R2$-MODE							
C(A,B)	**0.085**	**0.155**	**0.197**	**0.234**	**0.502**	**0.113**	**0.369**	**0.469**
C(B,A)	0.006	0.110	0.034	0.046	0.003	0.073	0.134	0.103

TABLE V

COMPARISON OF THE RESULTS OBTAINED BY NSGA-II, MOEA/D AND SMS-EMOA VS. $R2$-MOGA AND $R2$-MODE USING THE TWO SET COVERAGE PERFORMANCE MEASURE FOR THE DTLZ AND WFG TEST SUITES: C(A,B)=% OF SOLUTIONS IN B DOMINATED BY A.

functions, we decided to test the behavior of our proposed approach in such problems. For this experiment, we focused our efforts on solving the DTLZ{1-4} test problems in order to investigate the behavior of our proposed approaches with several objectives (we analyzed the behavior of our proposed approach from $M = 4$ to 10 objectives). Our results are compared with respect those obtained by SMS-EMOA (since this was the best approach in Experiment 1). Each MOEA was executed 30 times and their results were evaluated using the Hypervolume indicator. Since computational time is a usual drawback in indicator-based approaches, we decided to measure the execution time of the compared MOEAs as well. The reference point used for the adopted problems was of $r = [r_1 \dots r_M]$ where $r_i = 11$ for $1 \leq i \leq M$. The parameters were similar to those adopted in the previous experiment. Additionally, a second version of SMS-EMOA, called SMS-EMOA2 was added to the comparison or results. This version uses the approximation to the contribution to the Hypervolume (proposed in [5]) in order to decrease the execution time of the algorithm. The number of samples used in this latter algorithm is $100,000$. This addition was mandatory, since the computational time required by the original SMS-EMOA algorithm (using exact Hypervolume) becomes prohibitive very quickly as the number of objective raises (this has also been illustrated in other works [11]).

For this experiment, we adopted the Hypervolume performance measure. However, the values obtained by this performance measure tend to get bigger as we increase the number of objectives. Therefore, we decided to normalize these values in order to avoid problems when displaying them in our tables of results. The normalization procedure applied in each test problems was the following: First, we applied the Hypervolume to the obtained results by the compared approaches. Then, we searched for the highest Hypervolume value among all the executions. Finally, this highest value was used to normalize the Hypervolume results. Therefore, we prefer values which are closer to 1.

The results of the normalized Hypervolume and the running time (in brackets) for the compared approaches are shown in Tables VI, VII, VIII and IX. In some cases, the Hypervolume

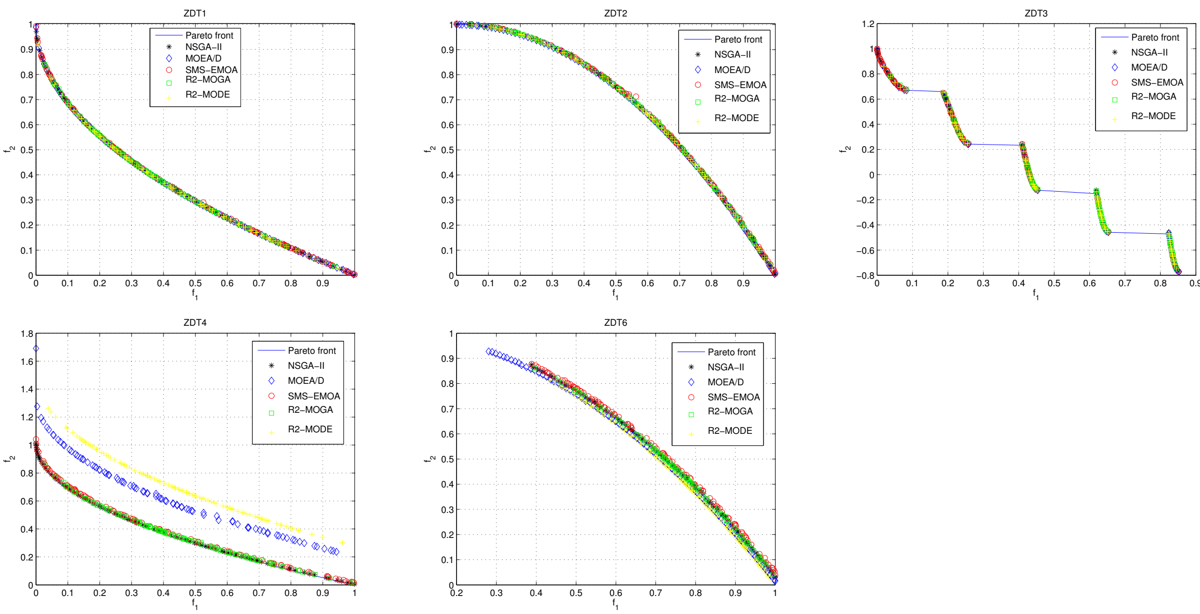

Fig. 1. Graphical results for the ZDT test suite (bi-objective)

has a value of zero. This value indicates that the algorithm did not not achieve any non-dominated solution with respect to the reference point. From these tables, it is also worth emphasizing that the time required by the original SMS-EMOA is greater than the time required by the rest of the approaches. Also, its execution for some problem sizes was prohibitive. That is the reason why we only report the results of SMS-EMOA for some instances. Furthermore, since the results of the original SMS-EMOA and its approximated version are very similar, we decided to discuss only the latter approach (since it required a lower computational time, and we have results for this approach for all the test problems instances).

For the easiest test problems (DTLZ2 and DTLZ4 shown in Tables VII and IX, respectively), the results of the compared approaches are similar. However, for DTLZ2 our $R2$-MOGA remains slightly behind SMS-EMOA2 and our proposed $R2$-MODE. On the other hand, for DTLZ1 and DTLZ3 (which are shown in Tables VI and VIII, respectively), our approaches clearly outperform SMS-EMOA2. For DTLZ1 the performance of SMS-EMOA2 decreased as we increased the number of objectives. The main problem of this algorithm is that the exact Hypervolume calculation was replaced with an approximation, and when the number of objectives raises it is necessary to increase the number of samples for the approximation, as well. However, if the number of samples is increased, the computational cost would also increase. Finally, for DTLZ3 our $R2$-MOGA clearly outperformed the others. This is a very challenging problem, and even our $R2$-MODE decreased its performance when we increased the number of objectives. SMS-EMOA2 was unable to converge in this problem.

Finally, Figure 3 shows the average time required by each of the tested approaches for each of the adopted problems. In this figure, we can see that our approaches required lower time than SMS-EMOA2. As a matter of fact, the computational time required of our approaches for the 10-objective instance of the adopted problems was lower than the 4-objective instance of the SMS-EMOA2 algorithm.

M	SMS-EMOA	SMS-EMOA2	R2-MOGA	R2-MODE
4	**1.0000**(71)	**1.0000**(9)	0.9992(2.8)	**1.0000**(3)
5	**1.0000**(1315)	0.9993(11)	0.9995(3.2)	**1.0000**(3.3)
6	——	0.9939(13)	0.9996(3.7)	**1.0000**(3.8)
7	——	0.9929(14.9)	0.9997(4.2)	**1.0000**(4.3)
8	——	0.9978(16.7)	0.9998(4.7)	**1.0000**(4.8)
9	——	0.9937(18.6)	0.9998(5.2)	**1.0000**(5.8)
10	——	0.9951(20.4)	0.9998(5.8)	**1.0000**(5.9)

TABLE VI
COMPARISON OF THE HYPERVOLUME'S AVERAGE AND (RUNNING TIME) OBTAINED BY SMS-EMOA, $R2$-MOGA AND $R2$-MODE FOR THE DTLZ1 TEST PROBLEM WITH DIFFERENT NUMBER OF OBJECTIVES (M)

M	SMS-EMOA	SMS-EMOA2	R2-MOGA	R2-MODE
4	**1.0000**(161)	**1.0000**(21.7)	0.9991(2.4)	**1.0000**(2.3)
5	**1.0000**(3230)	**1.0000**(20.6)	0.9995(2.8)	**1.0000**(2.8)
6	——	**1.0000**(20.8)	0.9997(3.3)	**1.0000**(3.3)
7	——	**1.0000**(21.7)	0.9998(3.7)	**1.0000**(3.7)
8	——	**1.0000**(23.2)	0.9998(4.2)	**1.0000**(4.2)
9	——	**1.0000**(24.9)	0.9998(4.6)	**1.0000**(4.7)
10	——	**1.0000**(27)	0.9999(5)	**1.0000**(5.2)

TABLE VII
COMPARISON OF THE HYPERVOLUME'S AVERAGE AND (RUNNING TIME) OBTAINED BY SMS-EMOA, $R2$-MOGA AND $R2$-MODE FOR THE DTLZ2 TEST PROBLEM WITH DIFFERENT NUMBER OF OBJECTIVES (M)

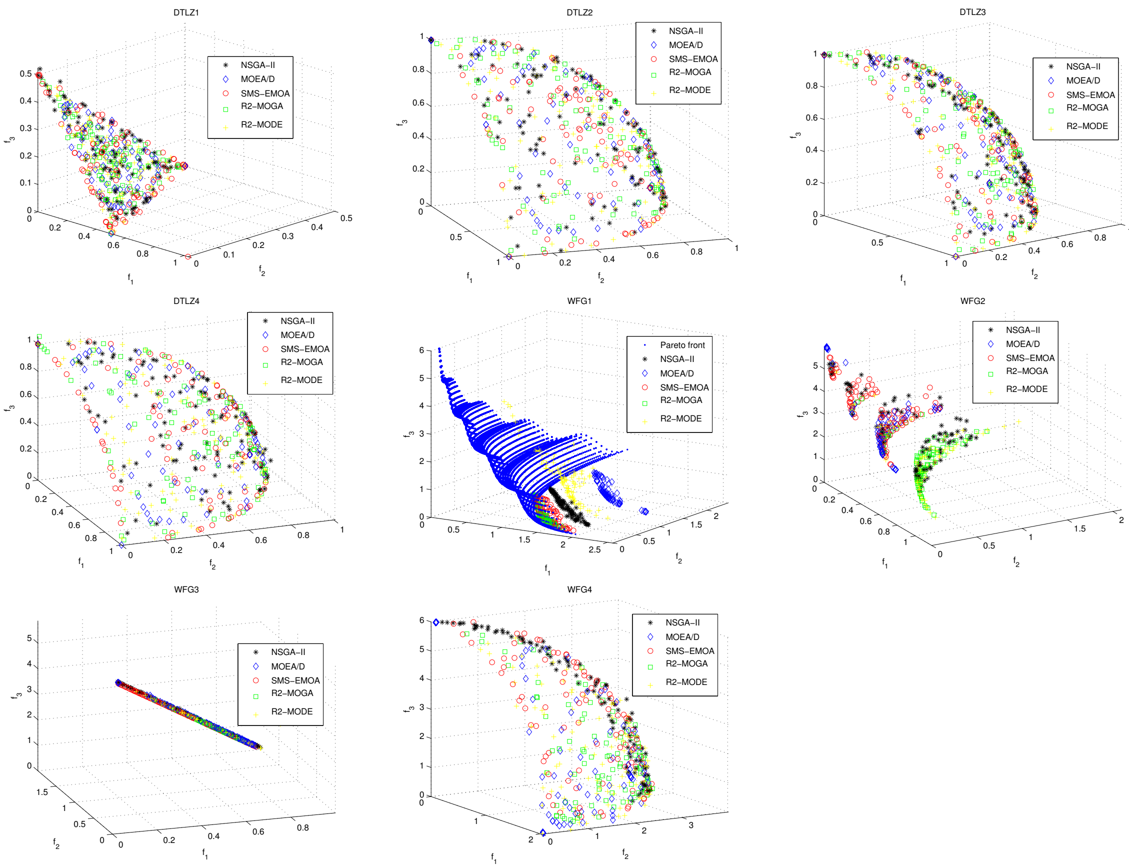

Fig. 2. Graphical results for the DTLZ and WFG test suites (three-objective)

M	SMS-EMOA	SMS-EMOA2	R2-MOGA	R2-MODE
4	0.9601(36)	0.0055(9.6)	**0.9984**(4)	0.8712(6.3)
5	0.9956(650)	0.0000(11.8)	**0.9994**(4.1)	0.7802(6.4)
6	——	0.0000(14.3)	**0.9992**(4.5)	0.6306(6.9)
7	——	0.0000(16.8)	**0.9996**(5)	0.5803(7.3)
8	——	0.0000(18.8)	**0.9998**(5.6)	0.3945(8)
9	——	0.0000(20.8)	**0.9995**(6.2)	0.2421(8.6)
10	——	0.0000(22.9)	**0.9997**(6.9)	0.1464(9)

TABLE VIII
COMPARISON OF THE HYPERVOLUME'S AVERAGE AND (RUNNING TIME) OBTAINED BY SMS-EMOA, $R2$-MOGA AND $R2$-MODE FOR THE DTLZ3 TEST PROBLEM WITH DIFFERENT NUMBER OF OBJECTIVES (M)

M	SMS-EMOA	SMS-EMOA2	R2-MOGA	R2-MODE
4	0.9996(160)	0.9989(28.2)	0.9998(3.7)	**1.0000**(3.3)
5	**1.0000**(3271)	0.9997(23.2)	0.9999(4.9)	**1.0000**(3.6)
6	——	**1.0000**(23.7)	**1.0000**(5.7)	**1.0000**(4.1)
7	——	**1.0000**(23.2)	**1.0000**(6.2)	**1.0000**(4.4)
8	——	**1.0000**(24.6)	**1.0000**(6.1)	**1.0000**(4.8)
9	——	**1.0000**(26.5)	**1.0000**(6.8)	**1.0000**(5.3)
10	——	**1.0000**(28.1)	**1.0000**(7.5)	**1.0000**(5.7)

TABLE IX
COMPARISON OF THE HYPERVOLUME'S AVERAGE AND (RUNNING TIME) OBTAINED BY SMS-EMOA, $R2$-MOGA AND $R2$-MODE FOR THE DTLZ4 TEST PROBLEM WITH DIFFERENT NUMBER OF OBJECTIVES (M)

V. CONCLUSIONS AND FUTURE WORK

In this work, we proposed the Fast $R2$ sorting which is a new approach to rank the individuals of a MOEA. The proposed ranking method uses the contribution to the $R2$ indicator in order to select solutions. The $R2$ contribution was successfully coupled to the notion of layers in order to produce a fast $R2$ sorting method. This sorting method was incorporated in two different search engines: a genetic algorithm and a differential evolution algorithm. Our proposed approaches were validated using two- and three-objective function problems. These approaches obtained competitive results when compared to NSGA-II, MOEA/D and SMS-EMOA using several test problems and performance indicators taken from the specialized literature. Therefore, we can say that our sorting method was successful. However, since our main target were many-objective problems, we decided to study our approaches in this sort of problems. For this sake, we adopted scalable test functions and we compared our results with respect to a well-known Hypervolume-based approach (SMS-EMOA) and a variation of it (SMS-EMOA2) that approximates the Hypervolume contributions, which results in a more efficient performance. Our results indicate that our proposed approaches outperform the others with respect to which it

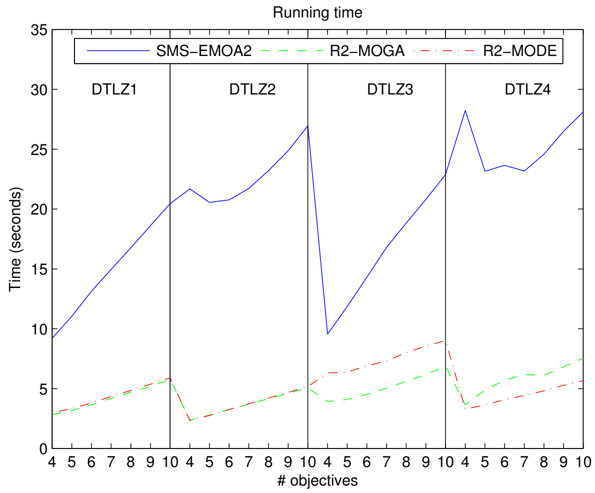

Fig. 3. Computational time required by SMS-EMOA2, $R2$-MOGA, $R2$-MODE for the DTLZ adopted problems

was compared, not only with respect to the Hypervolume, but also in terms of the required computational time. Therefore, we can argue that our proposed $R2$ sorting approach was successfully integrated to a MOEA, and we believe that this type of approach should be particularly useful for dealing with many-objective problems. Finally, an important feature of this proposal is that it does not adopt Pareto dominance at all.

As part of our future work, we would like to further investigate the incorporation of the new sorting method into other search engines. Furthermore, we are also interested in the use of our proposed approach for restricting the size of external archives.

References

[1] R. Purshouse and P. Fleming, "Evolutionary many-objective optimisation: An exploratory analysis," in *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, vol. 3. IEEE, 2003, pp. 2066–2073.

[2] V. Khare, X. Yao, and K. Deb, "Performance scaling of multi-objective evolutionary algorithms," in *Proceedings of the 2nd international conference on Evolutionary multi-criterion optimization*, ser. EMO'03. Berlin, Heidelberg: Springer-Verlag, 2003, pp. 376–390. [Online]. Available: http://dl.acm.org/citation.cfm?id=1760102.1760134

[3] E. Hughes, "Evolutionary many-objective optimisation: many once or one many?" in *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 1, sept. 2005, pp. 222 –227 Vol.1.

[4] E. J. Hughes, "Fitness assignment methods for many-objective problems," in *Multiobjective Problem Solving from Nature*, ser. Natural Computing Series, J. Knowles, D. Corne, K. Deb, and D. Chair, Eds. Springer Berlin Heidelberg, 2008, pp. 307–329.

[5] J. Bader and E. Zitzler, "Hype: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, Mar. 2011. [Online]. Available: http://dx.doi.org/10.1162/EVCO_a_00009

[6] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Parallel Problem Solving from Nature - PPSN VIII*, ser. Lecture Notes in Computer Science, X. Yao, E. Burke, J. Lozano, J. Smith, J. Merelo-Guervós, J. Bullinaria, J. Rowe, P. Tino, A. Kabán, and H.-P. Schwefel, Eds. Springer Berlin / Heidelberg, 2004, vol. 3242, pp. 832–842.

[7] E. Zitzler, "Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications," Ph.D. dissertation, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, November 1999.

[8] N. Beume, B. Naujoks, and M. Emmerich, "Sms-emoa: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653 – 1669, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221706005443

[9] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Iterative approach to indicator-based multiobjective optimization," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, sept. 2007, pp. 3967 –3974.

[10] C. Igel, N. Hansen, and S. Roth, "Covariance matrix adaptation for multi-objective optimization," *Evol. Comput.*, vol. 15, no. 1, pp. 1–28, Mar. 2007.

[11] C. A. Rodríguez Villalobos and C. A. Coello Coello, "A new multi-objective evolutionary algorithm based on a performance assessment indicator," in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference*, ser. GECCO '12. New York, NY, USA: ACM, 2012, pp. 505–512.

[12] O. Schütze, X. Esquivel, A. Lara, and C. A. C. Coello, "Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization," *IEEE Trans. Evolutionary Computation*, vol. 16, no. 4, pp. 504–522, 2012.

[13] S. Wessing and B. Naujoks, "Sequential parameter optimization for multi-objective problems," in *Evolutionary Computation (CEC), 2010 IEEE Congress on*, july 2010, pp. 1 –8.

[14] D. Brockhoff, T. Wagner, and H. Trautmann, "On the properties of the r2 indicator," in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference*, ser. GECCO '12. New York, NY, USA: ACM, 2012, pp. 465–472.

[15] M. Hansen and A. Jaszkiewicz, *Evaluating the quality of approximations to the non-dominated set*. IMM, Department of Mathematical Modelling, Technical Universityof Denmark, 1998.

[16] E. Zitzler, J. Knowles, and L. Thiele, "Quality assessment of pareto set approximations," *Multiobjective Optimization*, pp. 373–404, 2008.

[17] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989.

[18] A. Jaszkiewicz, "Momh multiple-objective metaheuristics," Apr. 2013. [Online]. Available: http://home.gna.org/momh/index.html

[19] K. Deb and R. B. Agrawal, "Simulated Binary Crossover for Continuous Search Space," Departement of Mechanical Enginering, Indian Institute of Technology, Kanpur, India, Tech. Rep., 1994. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.26.8485

[20] K. Deb and M. Goyal, "A combined genetic adaptive search (geneas) for engineering design," *Computer Science and Informatics*, vol. 26, pp. 30–45, 1996.

[21] R. Storn and K. Price, "Differential evolution – a fast and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[22] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, Jun. 2000. [Online]. Available: http://dx.doi.org/10.1162/106365600568202

[23] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," in *Evolutionary Multiobjective Optimization*, ser. Advanced Information and Knowledge Processing, A. Abraham, L. Jain, and R. Goldberg, Eds. Springer Berlin Heidelberg, 2005, pp. 105–145.

[24] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 5, pp. 477–506, 2006.