An Improved Version of a Reference-Based Multi-Objective Evolutionary Algorithm based on IGD$^+$

Edgar Manoatl Lopez
CINVESTAV-IPN (Evolutionary Computation Group)
MÉXICO, México D.F.
emanoatl@computacion.cs.cinvestav.mx

Carlos A. Coello Coello
CINVESTAV-IPN (Evolutionary Computation Group)
MÉXICO, México D.F.
ccoello@cs.cinvestav.mx

ABSTRACT

In recent years, the design of new selection mechanisms has become a popular trend in the development of Multi-Objective Evolutionary Algorithms (MOEAs). This trend has been motivated by the aim of maintaining a good balance between convergence and diversity of the solutions. Reference-based selection is, with no doubt, one of the most promising schemes in this area. However, reference-based MOEAs are known to have difficulties for solving multi-objective problems with complicated Pareto fronts, mainly because they rely on the consistency between the Pareto front shape and the distribution of the reference weight vectors. In this paper, we propose a reference-based MOEA, which uses the Inverted Generational Distance plus (IGD$^+$) indicator. The proposed approach adopts a novel method for approximating the reference set, based on an hypercube-based method. Our results indicate that our proposed approach is able to obtain solutions of a similar quality to those obtained by RVEA, MOEA/DD, NSGA-III and MOMBI-II in several test problems traditionally adopted in the specialized literature, and is able to outperform them in problems with complicated Pareto fronts.

CCS CONCEPTS

• **Computing methodologies** → *Continuous space search*;

KEYWORDS

Multi-objective Optimization; Performance Indicators; IGD$^+$ Indicator; Reference Search Methods; Degenerate Pareto Fronts

ACM Reference Format:
Edgar Manoatl Lopez and Carlos A. Coello Coello. 2018. An Improved Version of a Reference-Based Multi-Objective Evolutionary Algorithm based on IGD$^+$. In *GECCO '18: Genetic and Evolutionary Computation Conference, July 15–19, 2018, Kyoto, Japan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3205455.3205530

1 INTRODUCTION

A large number of real-world problems have several objectives (which are often in conflict with each other) that need to be optimized at the same time. These are the so-called Multi-objective Optimization Problems (MOPs) and their solution involves finding the best possible trade-offs among all the objectives. This set of trade-offs, when defined in decision variable space, is known as the *Pareto optimal set* ($\mathcal{PS}$). The image of the Pareto optimal set is called the *Pareto optimal front* ($\mathcal{PF}$). Recently, Multi-Objective Evolutionary Algorithms (MOEAs) have become an increasingly common approach for solving MOPs, mainly because of their conceptual simplicity, ease of use and efficiency.

For several years, MOEAs adopted selection mechanisms based on Pareto optimality. However, it has been found that Pareto-based MOEAs can not properly solve many-objective problems (problems with more than three objectives) [15]. This has motivated the development of new strategies for dealing with many-objective problems such as reference-based MOEAs [4, 7, 17, 18]. Reference-based MOEAs can be classified into two main groups: (1) decomposition-based MOEAs and (2) indicator-based MOEAs which rely on the use of reference sets. Decomposition-based MOEAs transform a MOP into a group of sub-MOPs, where each sub-MOP is defined by a reference weight point. Each sub-MOP is optimized simultaneously by a single-objective optimizer [26]. This sort of MOEAs have shown to be better than Pareto-based MOEAs both in traditional MOPs and in many-objective problems. However, the main drawback of decomposition-based MOEAs is that the diversity of its selection mechanism is controlled explicitly by the reference weight vectors. This is because each weight vector corresponds to one subproblem to be solved. On the other hand, MOEAs based on the hybridization of a reference set and an performance indicator have shown to be promising schemes for solving many-objective optimization problems [11, 12, 19]. When compared to hypervolume-based MOEAs[1] [1, 27], indicator-based reference-based MOEAs have a significantly lower computational cost and are able to obtain approximations of a similar quality to hypervolume-based MOEAs. Although effective and suitable for many-objective optimization, reference-based MOEAs in general require the generation of a set of reference weight vectors, analogously to decomposition-based MOEAs. In general, if the set of weight vectors and the Pareto front of a MOP share the same distribution, it is possible to obtain well-distributed approximations. There is, however, experimental evidence that indicates that the weight vectors most commonly used by these MOEAs adopt a simplex-like shape. This sort of scheme works well for Pareto fronts with regular shapes (e.g., a triangle or a sphere). Unfortunately, this scheme doesn't work properly with some complicated Pareto fronts (e.g., disconnected, degenerate, inverted simplex-like or badly-scaled). Empirical studies have

[1]The main drawback of hypervolume-based MOEAs is the high computational cost associated with the computation of the exact hypervolume contributions, which becomes unaffordable when trying to solve many-objective optimization problems.

GECCO '18, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5618-3/18/07...$15.00
https://doi.org/10.1145/3205455.3205530

shown that decomposition-based MOEAs and some indicator-based MOEAs have difficulties to solve these MOPs with complicated Pareto fronts. This motivated the work reported here, in which we propose a novel MOEA that transforms its selection mechanism into a Linear Assignment Problem (LAP), and adopts the Inverted Generational Distance plus (IGD$^+$) indicator as a cost function of the LAP. In order to compute the IGD$^+$ indicator, we also incorporate an adaptive method for building the reference point set. This method is based on the creation hypercubes. We show that the resulting MOEA has a competitive performance with respect to state-of-the-art MOEAs, and that is able to properly deal with MOPs having complicated Pareto fronts.

The remainder of this paper is organized as follows. Section 2 provides some basic concepts related to multi-objective optimization. The most relevant previous related work is described in Section 3. Our proposed approach is shown in Section 4. Our methodology is presented in Section 5. Section 6 presents our discussion results. Finally, our conclusions and some possible paths for future research are provided in Section 7.

2 BASIC CONCEPTS

Formally a MOP in terms of minimization is defined as:

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

subject to:

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

$$h_j(\vec{x}) = 0, \quad j = 1, 2, \ldots, q \tag{3}$$

where $\vec{x} = [x_1, x_2, \ldots, x_n]$ is the vector of decision variables, $f_k : \mathbb{R}^n \to \mathbb{R}$, $k = 1, \ldots, m$ are the objective functions and $g_i, h_j : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, p$, $j = 1, \ldots, q$ are the constraint functions of the problem. Next, we introduce some definitions that will be used in the paper.

Definition 1. *Let $\vec{x}, \vec{y} \in \mathbb{R}^m$, we say that $\vec{x}$ "dominates" $\vec{y}$ (denoted by $\vec{x} \prec \vec{y}$), if and only if: i) $x_i \leq y_i$ for all $i \in \{1, \ldots, m\}$ and ii) $x_j < y_j$ for at least one $j \in \{1, \ldots, m\}$.*

Definition 2. *We say that a vector of decision variables $\vec{x} \in \mathcal{X} \subset \mathbb{R}^n$ is "non-dominated" with respect to $\mathcal{X}$ and $\vec{f}$, 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 $\vec{x} \in \mathcal{F} \subset \mathbb{R}^n$ (where $\mathcal{F}$ is the feasible region) is "Pareto optimal" if it is non-dominated with respect to $\mathcal{F}$ and $\vec{f}$.*

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

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

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

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

Since we adopt the IGD$^+$ indicator, we will proceed to define it. The IGD$^+$ indicator evaluates the quality of an objective vector using a reference set. According to [14], the IGD$^+$ indicator is defined as follows:

$$\text{IGD}^+(\mathcal{A}, \mathcal{Z}) = \frac{1}{|\mathcal{Z}|} \left(\sum_{j=1}^{|\mathcal{Z}|} \hat{d}_j + (\vec{z}, \vec{a})^p \right)^{1/p} \tag{4}$$

where $\hat{d}$ is the nearest modified Euclidean distance from $z_j \in \mathcal{Z}$ to $\mathcal{A}$ and the modified Euclidean distance $d^+(\vec{a}, \vec{z})$ is defined as:

$$d^+(\vec{z}, \vec{a}) = \sqrt{(\max\{a_1 - z_1, 0\})^2 + \cdots + (\max\{a_m - z_m, 0\})^2}. \tag{5}$$

Therefore, a lower IGD$^+$ value means that the set $\mathcal{A}$ constitutes a better approximation to the real $\mathcal{PF}$ if we consider the reference set to be $\mathcal{PF}_{True}$. This indicator was shown to be weakly Pareto compliant and presents some evident advantages with respect to the original Inverted Generational Distance (IGD) [6] (for more details about IGD$^+$, see [14]).

3 PREVIOUS RELATED WORK

Next, we will briefly discuss some reference-based MOEAs. Our discussion will focus specifically on approaches that adopt reference weight vectors for leading the optimization process. The approaches discussed are divided in two main groups: (a) decomposition-based MOEAs and (b) indicator-based MOEAs which rely on the use of reference sets.

The *Multi-objective Evolutionary Algorithm Based on Decomposition* (MOEA/D) [26] is the best well-known decomposition-based MOEA. MOEA/D divides the whole $\mathcal{PF}$ into a group of sub-spaces, each of which can be regarded as a sub-MOP. Another example of MOEA that belongs in this category is NSGA-III [7], which uses a distributed set of reference points to manage the diversity of the candidate solutions, whose aim is to improve convergence. There is also an extension of MOEA/D which includes the Pareto dominance relation to select candidate solutions. This MOEA is called MOEA/DD [17], and is able to outperform the original MOEA/D, particularly in many-objective problems (the authors of MOEA/DD validated it with unconstrained and constrained benchmark problems having up to 15 objectives). The *Reference Vector Guided Evolutionary Algorithm* (RVEA) [4] is a very promising MOEA that provides very competitive results in MOPs with complicated Pareto fronts. RVEA incorporates a novel method to preserve good candidate solutions, which consists of an adaptive technique for adjusting the reference vectors in order to balance the convergence and diversity of the solutions in high-dimensional objective spaces.

Regarding indicator-based MOEAs which rely on the use of reference sets, our discussion will focus specifically on approaches that adopt either IGD$^+$ or Δ_p[2] (both of which are based on the use of reference sets). The first MOEA based on Δ_p was DDE [21], which uses differential evolution (DE) as its search engine. Its authors showed that DDE was able to converge rapidly towards the true Pareto front and that it could properly solve many-objective problems. However, this approach had some limitations related to

[2] Δp is a Pareto non-compliant performance indicator which combines the indicators called Generational Distance (GD) and Inverted Generational Distance (IGD), which are able to assess both convergence and distribution of a Pareto front approximation given a reference point set. For details on Δ_p, see [6, 22].

the use of Δ_p (e.g., it does not properly work with multi-frontal and degenerate MOPs). On the other hand, Δ_p-EMOA [10] is another approach based on the use of Δ_p, which is inspired on SMS-EMOA [1] and incorporates a novel mechanism for building the reference set. This algorithm builds a reference point set by linearizing the non-dominated front of the current population. Δ_p-EMOA was designed to solve only bi-objective problems. Nevertheless, there is an extension of this approach, which is able to solve three-objective problems [23]. Another approach based on the Δ_p indicator is the *Reference Indicator-Based Evolutionary Multi-Objective Algorithm* (RIB-EMOA) [25]. This MOEA integrates a mechanism for building a reference set by using a family of curves. RIB-EMOA uses a weight vector set for approximating the reference point set. Although promising for many-objective optimization, this approach is not able to solve MOPs with complicated Pareto fronts. Another MOEA based on Δ_p was proposed in [20]. This algorithm uses ϵ-dominance in order to build a reference point set. This novel algorithm was validated using standard test functions, having from three to six objective functions. Its authors showed that this algorithm is able to solve convex, concave and disconnected MOPs. Additionally, they provided a promising solution to avoid the use of reference weight vectors for guiding the optimization process.

In recent years, some researchers have proposed other types of indicator-based reference-based MOEAs, such as R2-MOEA, which is based on the $R2$ indicator [3]. Same as when using Δ_p, the use of $R2$ requires a reference set of weights in order to compute its value. Another $R2$-based MOEA is the *Many Objective Meta-heuristic Based on the $R2$ indicator* (MOMBI) [11]. This algorithm adopts the use of weight vectors and the $R2$ indicator [3], and both mechanisms lead the optimization process. Although MOMBI is very competitive, it loses diversity in high dimensionality. This motivated the development of an improved version of this approach, called MOMBI-II [12], which uses a scalarizing function and statistical information for selecting the candidate solutions. Recently, a new MOEA based on the IGD$^+$ indicator, called IGD$^+$-EMOA, was proposed in [18]. This MOEA adopts a transformation to a Linear Assignment Problem (LAP) into its selection mechanism. Its authors proposed a novel technique for approximating the reference set, which is based on the use of γ-supersphere curves. In spite of the good performance of this approach in many-objective problems, it had difficulties for solving MOPs with degenerate Pareto fronts. Here, we extend that work by providing a novel method for approximating the reference point set using hypercubes. As we will see later on, this new version of IGD$^+$-EMOA is able solve many-objective optimization problems with complicated Pareto front shapes. Our approach has the following improvements. (1) An archiving process for preserving candidate solutions which will be form the reference set. (2) A method for adapting the reference set in order to sample uniformly the Pareto front. (3) A rule for updating the reference set.

4 OUR PROPOSED APPROACH

4.1 General Framework

The general framework of a MOEA adopted in this paper starts with a population $\mathcal{P}_0$ which contains N randomly generated individuals. A new offspring is created by choosing two different parents from $\mathcal{P}$. The parents are recombined using evolutionary operators.[3] After that, the resulting offspring are added to the new set. This process is executed until having a total of λ offspring. Thereafter, the algorithm combines the parents and the offspring populations to form the so-called $\mathcal{A}$ set. In order to select the next population, we apply our LAP-based selection mechanism.

4.2 Selection Mechanism

Since, we intend to use the IGD$^+$ indicator in the selection mechanism of our MOEA, we adopt the same selection mechanism proposed by IGD$^+$-EMOA [19]. This selection mechanism transforms the environmental selection mechanism into a Linear Assignment Problem (LAP). In order to explain this technique, we need to provide first more details about the LAP. The LAP is the problem of choosing an optimal assignment of n items (e.g., jobs) to m machines (or workers). Mathematically, the LAP can be expressed as: Given two sets, $A = \{a_1, \ldots, a_n\}$ and $T = \{t_1, \ldots, t_n\}$ with the same cardinality, and a cost function $C : A \times T \to \mathbb{R}$ and having $\Phi : A \to T$ as the set of all bijections between A and T. So, the LAP can be formulated as follows:

$$\min_{\phi \in \Phi} \sum_{a \in A} C(a, \Phi(a)) \tag{6}$$

Normally, the cost of the problem is also described as a squared matrix C, where each element $C_{i,j} = C(a_i, t_j)$ represents the relationship between a_i and t_j. Thus, the cost matrix in terms of a MOP is created as:

$$C_{i,j} = d^+(a_i, z_j), \quad i = 1, \ldots, |\mathcal{A}|, \quad j = 1, \ldots, |\mathcal{Z}|. \tag{7}$$

where $\vec{a}_i \in \mathcal{A}$ is the i^{th} vector point from the population $\mathcal{A}$, $\vec{z}_j \in \mathcal{Z}$ is the reference point and d^+ is the modified Euclidean distance. In order to solve the LAP, we use an extension of Kuhn-Munkres' Algorithm for rectangular matrices [2], which is able to solve LAP instances in polynomial time ($O(n^3)$) [16]. The extension to rectangular matrices allows the selection mechanism to operate in situations where the number of reference points and individuals from the population are not equal. The optimal solution to our assignment problem corresponds to the best relationship between the current points of the population and the elements of a reference set. In order to build the reference point set, the algorithm consists of two main procedures: a procedure to maintain non-dominated solutions into an archive and a mechanism to remove non-candidate solutions with a poor distribution from the archive. Next, we provide more detail about these two procedures.

4.3 Archiving Process

The archive has a pre-set capacity to store the non-dominated solutions, and the maximum number of solutions that are allowed in the archive is defined by a specific p value. When the archive reaches its maximum capacity, the approximation reference algorithm is executed for selecting candidate solutions (these candidate solutions will form the so-called *reference set*). After that, the archive is cleared and the archiving process continues until reaching a maximum number of generations. It is worth mentioning that the archiving process is applied at each generation of the MOEA.

[3] In our implementation, we adopted SBX (Simulated Binary Crossover) and Polynomial-based Mutation [8].

4.4 Building the Reference Set

In IGD$^+$-EMOA, we aim to select the best reference points whose directions are promising (i.e., directions with good distribution and spread). In order to do that, we adopted a greedy algorithm based on the hypercube contributions to select a certain number of reference points from the archive. This greedy algorithm executes a density estimator for computing the hypercubes. Its pseudo-code is shown in Algorithm 1. The algorithm is organized as three consecutive loops, and is invoked with a set of non-dominated candidate points (called $\mathcal{A}$ set) and the maximum number of reference points that we aim to find. In the first loop, we create a set of initial candidate solutions to form the so-called Q set. Thus, the solutions from $\mathcal{A}$ that form part of Q, will be removed from $\mathcal{A}$. After that, the greedy algorithm starts to find the best candidate solutions which will form the reference set $\mathcal{Z}$. In order to find the candidate reference points, the selection mechanism computes the hypercube contributions of the current reference set Q.

Algorithm 1: ComputeReferenceSet($\mathcal{A}, z_{size}$)

```
Input: A current non-dominated set 𝒜 ⊂ ℝ^m and maximum
   number of reference points z_size.
Output: Reference point set 𝒵 ⊂ ℝ^m with |𝒵| = z_size
   y_ref ← FindMaxValue(𝒜) + ε;
   Q ← {};
   while |Q| < (z_size + 1) do
      a⃗ ← pop(𝒜);
      Q ⋃{a⃗} ;
   end while
   while 𝒜! = {} do
      i ← 0;
      maxHypercube ← HCB(Q, y_ref);
      for each q⃗ ∈ Q do
         ContHyperCube[i] ← maxHypercube−HCB(Q\{q⃗}, y_ref);

         i ← i + 1;
      end for
      i_min ← argmin ContHyperCube;
      Q\{q_{i_min}};
      a⃗ ← pop(𝒜);
      Q ⋃{a⃗};
   end while
   𝒵 ← {};
   for each q⃗ ∈ Q do
      𝒵 ⋃{q⃗ ∗ ε − l⃗};
   end for
   return 𝒵;
```

Once this is done, we remove the i^{th} solution that minimizes the hypercube value and we add a new candidate solution from $\mathcal{A}$ to Q. This process is executed until the cardinality of $\mathcal{A}$ is equal to zero. In the last loop, we apply the *expand* and *translate* operations. These operations transform the surface for spreading the reference set along objective function space.

A hypercube is generated by the union of all the maximum volumes covered by a reference point. The i^{th} maximum volume is described as "the maximum volume generated by a set of candidate points" (these candidate points are obtained from the archive and a reference point y_{ref}). The hypercube is computed using Algorithm 2.

Algorithm 2: HCB(Q, y_{ref})

```
Input: A current set Q ⊂ ℝ^m and a reference point y_ref
Output: Hypercube value
   if |Q| = 1 then
      return vol(Q, y_ref);
   end if
   VolList ← {};
   for each p⃗ ∈ Q′ do
      VolList ⋃{vol(p⃗, y_ref)};
   end for
   i_max ← argmax VolList;
   q⃗_max ← Q[i_max] ;
   𝒴 ← SplitReferencePoint (q⃗_max, y_ref);
   Q\{q⃗_max};
   hypercube ← 0;
   for each y⃗_new ∈ 𝒴 do
      Q_new ← CoverPoints (Q, y⃗_new);
      hypercube ← hypercube + HCB(Q_new, y⃗_new);
   end for
   return hypercube + max(VolList);
```

In the first part of Algorithm 2, we validate if Q contains one element. If that is the case, we compute the volume generated by y_{ref} and $\vec{q} \in Q$. Otherwise, we compute the union of all the maximum hypercubes. In order to apply this procedure, we find the vector $q_{\vec{max}}$ that maximizes the hypercube. Once this is done, we create m reference points which will form the so-called $\mathcal{Y}$. In order to create the set $\mathcal{Y}$, we combine the current reference point y_{ref} and the point $\vec{q}_{max}$. For each reference point from $\mathcal{Y}$, we reduce the set Q into a small subset in order to form the set Q_{new}, this makes that we have to split any point from Q, whose components are not covered by the $\vec{y}_{new}$[4]. Once this is done, we proceed to compute recursively the hypercube value of the new set formed by the subset Q_{new} and the new reference point y_{new}. Algorithm 2 provides an efficient way of estimating the hypercube value. It is worth noting that this value allows to measure the relationship among each element of a non-dominated set.

4.5 Update Frequency

The timing and frequency of updating the reference set plays an important role in this algorithm. The generated reference point set does not always contribute in a good way because a frequent updating can significantly affect the performance of the algorithm. Therefore, we propose two additional mechanisms for updating the reference set. The first one consists in updating the reference set if the variance of the hypercube contribution of the new reference set is lower than the variance of the previous reference set. In the second mechanism, if the hypercube value of the previous reference

[4]In order to split the ith point from set Q, we invoke the method "CoverPoints"

Table 1: Main properties of the 18 test problems adopted

Properties	Problems
Linear	DTLZ1
Convex and Concave	DTLZ2-3, MAF2-5, WFG1
Inverted Simplex-like	MAF1
Disconnected	DTLZ7, WFG2
Degenerate	DTLZ5-6, VNT2-3, WFG3
Badly-scaled	MAF4-5

set is less than the hypercube value of the new reference set, then the new reference set is replaced by the previous one. It is worth indicating that these two mechanisms are adopted in our proposed approach.

5 EXPERIMENTAL STUDY

We compare the performance of our IGD$^+$-EMOA with respect to that of four state-of-the-art MOEAs: NSGA-III [7], RVEA [4], MOMBI-II [12] and MOEA/DD [17]. These MOEAs had been found to be competitive on MOPs with a variety of Pareto front shapes.

5.1 Test problems

We aimed to study the performance of our proposed approach when solving MOPs with complicated Pareto front shapes. For this reason, we selected 18 test problems with a variety of representative Pareto front shapes from some well-known and recently proposed test suites (i.e., the DTLZ test suite [9], the WFG test suite [13], the MAF test suite [5] and the VNT test suite [24]). Based on the properties of their Pareto fronts, we categorized the test problems adopted into different groups: convex, concave, inverted simplex-like, disconnected, degenerate and badly-scaled (see Table 1).

5.2 Methodology

For our comparative study, we decided to adopt the hypervolume indicator, which assesses both convergence and maximum spread along the Pareto front. If Λ denotes the Lebesgue measure, the hypervolumen can be described as follows:

$$I_H(\mathcal{A}, \vec{y_{ref}}) = \Lambda\left(\bigcup_{\vec{y}\in\mathcal{A}} \{\vec{x} \mid \quad \vec{y} \prec \vec{x} \prec \vec{y_{ref}}\}\right) \tag{8}$$

where $\mathcal{A}$ is the approximation of the Pareto optimal set and $\vec{y_{ref}} \in \mathbb{R}^k$ denotes the reference point. To compute I_H, we used the reference points shown in Table 2.

5.3 Parameterization

In the DTLZ and MAF test suites, the total number of decision variables is given by $n = m + k - 1$, where m is the number of objectives and k was set to 5 for DTLZ1, MAF1 and to 10 for DTLZ2-6, and MAF2-5. The number of decision variables in the WFG test problems was set to 24, and the position-related parameter was set to $m - 1$. The parameters of each MOEA adopted in our study were chosen in such a way that we could do a fair comparison among them. The distribution indexes for the SBX and polynomial-based mutation operators [8], used by all algorithms, were set to: $\eta_c = 20$ and $\eta_m = 20$, respectively. The crossover probability was set to $p_c = 0.9$ and the mutation probability was set to $p_m = 1/L$, where L is the number of decision variables. The total number of function evaluations was set in such a way that it did not exceed 60,000. In MOEA/DD, MOMBI-II and NSGA-III, the number of weight vectors was set to the same value as the population size. The population size N is dependent on H which specifies the number of divisions in objective space. H was set in such a way that N took a value not greater than 120. In RVEA, the rate of changing the penalty function and the frequency to conduct the reference vector adaptation were set to 2 and 0.1, respectively. The main characteristics of the hardware used for the experiments were the following: an Intel Core i7-3930k CPU running at 2.30 GHz, with 8GB of RAM.

Table 2: Reference points used for the hypervolume indicator

Problem	Reference point	Problem	Reference point
DTLZ1	(1,1,1)	VNT1	(5, 6, 5)
DTLZ2-6	(2,2,2)	VNT2	(5, -15, -11)
DTLZ7	(2, 2, 7)	VNT3	(9, 18, 5)
MAF1-3	(2,2,2)	WFG1	(3, 5, 7)
MAF4	(3,5, 9)	WFG2	(2, 4, 7)
MAF5	(9, 5, 3)	WFG3	(2, 3, 7)

6 DISCUSSION OF RESULTS

Table 3 provides the average hypervolume over the 30 independent executions of each compared MOEA for each instance of the DTLZ, WFG, VNT and MAF test suites. The best results are shown in **boldface** and grey-colored cells show the second best results. The variance is shown in parentheses. The Wilcoxon rank sum test was adopted to compare the results obtained by IGD$^+$-EMOA and its competitors at a significance level of 0.05, where the symbol "+" indicates that the compared algorithm is significantly outperformed by IGD$^+$-EMOEA. On the other hand, the symbol "-" means that our approach is significantly outperformed by its competitor. Finally, " $\approx$" means that there is no statistically significant difference between the results obtained by IGD$^+$-EMOA and the compared algorithm. It is clear that the winner in this experimental study is our proposed IGD$^+$-EMOA since it was able to outperform MOEA/DD, RVEA, MOMBI-II and NSGA-III in ten cases and in a few more, it obtained very similar results to those of the best performer. Figures 1, 2, 3 and 4 present a graphical representation of the approximations to the Pareto front obtained by each MOEA in some instances of the MAF and VNT test problems adopted with 3 objectives. On the MOPs with inverted Simplex-like Pareto fronts, IGD$^+$-EMOA showed a clear advantage over its competitors (see Figure 1). Figures 1.a to 1.e show that the solutions produced by all the MOEAs adopted have a good coverage of the Pareto front. However, the solutions of MOMBI-II and NSGA-III are not distributed very uniformly, while the solutions of RVEA and MOEA/DD are distributed uniformly but their number is apparently less than their population size. On MOPs with degenerate Pareto fronts, our proposed approach had also a good performance. Table 3 indicates that IGD$^+$-EMOA was able to outperform its competitors in this type of MOPs since its solutions are distributed more uniformly (see Figure 3 which shows the results obtained for VNT2). MOMBI-II, RVEA and IGD$^+$-EMOA are able to obtain solutions of a similar quality when they solve MOPs with badly-scaled Pareto fronts, but our approach was not able to outperform MOMBI-II in MAF4. However, IGD$^+$-EMOA was better than its competitors when solving MAF5. On the other hand, it is well-known that MOMBI-II, RVEA and NSGA-III can solve efficiently MOPs with simplex-like Pareto fronts. In this regard, it is worth mentioning that in these MOPs, our proposed IGD$^+$-EMOA was able to obtain approximations of a similar quality to those obtained by its competitors. For DTLZ7, IGD$^+$-EMOA did not perform better than the other MOEAs. The reason is probably that the Pareto front shape of this problem is disconnected, which makes the approximations produced by our approach to converge to a single region. We can see in Table 3 that the variance obtained by IGD$^+$-EMOA increases significantly in this MOP. We can conclude that the construction of our reference point set is very sensitive to this sort of scenarios, which is a clear weakness of our proposed approach.

Table 4 shows a preliminary study on degenerate many-objective problems by considering DTLZ5 with 3 up to 10 objectives. We can see that our proposed approach significantly outperformed its competitors, since MOMBI-II, RVEA, MOEA/DD and NSGA-III were not able to converge to the true Pareto front as the number of objectives was increased. We can see in Table 4 that the performance of MOMBI-II, RVEA, MOEA/DD and NSGA-III is not consistent when the number of objectives is greater than 8. Figure 6 presents a graphical representation (using parallel coordinates plots) of the approximations of the Pareto front obtained by each MOEA solving DTLZ5 with 10 objectives. We can see that our proposed IGD$^+$-EMOA was able to obtain the best results in terms of Hypervolume indicator, which makes it that our approach can solve MOPs with degenerate shapes even in many-objective instances.

7 CONCLUSIONS

We have proposed a reference-based MOEA for solving many-objective problems with a particular emphasis on those having complicated Pareto front shapes. The core idea of our proposed approach is to adopt the IGD$^+$ performance indicator in its selection mechanism. Additionally, our proposal introduces a novel method for building the reference set which is based on the use of hypercubes [5]. Our results indicate that the use of hypercubes significantly improves the performance of IGD$^+$-EMOA. As can be observed, the reference set is of utmost importance since our approach guides its search process using a set of reference points. Our preliminary results indicate that IGD$^+$-EMOA is very competitive with respect to MOMBI-II, RVEA, MOEA/DD and NSGA-III, being able to outperform them in more than 50 percent of the 18 benchmark problems adopted. Based on such results, we claim that our proposed approach is a competitive alternative to deal with MOPs having complicated Pareto front shapes, even in high-dimensional objective spaces. As part of our future work, we are interested in studying the sensitivity of our proposed approach to the reference set, with the aim of addressing the main weakness that was identified in our comparative study (i.e., disconnected Pareto fronts). Furthermore, we are also interested in incorporating a local search mechanism into our proposed MOEA, with the aim of improving its performance.

[5] The source code of our proposed approach is available from the first author, upon request.

REFERENCES

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

[2] François Bourgeois and Jean-Claude Lassalle. 1971. An Extension of the Munkres Algorithm for the Assignment Problem to Rectangular Matrices. *Commun. ACM* 14, 12 (Dec. 1971), 802–804. https://doi.org/10.1145/362919.362945

[3] Dimo Brockhoff, Tobias Wagner, and Heike Trautmann. 2012. On the Properties of the $R2$ Indicator. In *2012 Genetic and Evolutionary Computation Conference (GECCO'2012)*. ACM Press, Philadelphia, USA, 465–472. ISBN: 978-1-4503-1177-9.

[4] Ran Cheng, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. 2016. A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation* 20, 5 (October 2016), 773–791.

[5] Ran Cheng, Miqing Li, Ye Tian, Xingyi Zhang, Shengxiang Yang, Yaochu Jin, and Xin Yao. 2017. A benchmark test suite for evolutionary many-objective optimization. *Complex & Intelligent Systems* 3, 1 (01 Mar 2017), 67–81. https://doi.org/10.1007/s40747-017-0039-7

[6] Carlos A. Coello Coello and Margarita Reyes Sierra. 2004. A Study of the Parallelization of a Coevolutionary Multi-Objective Evolutionary Algorithm. In *Proceedings of the Third Mexican International Conference on Artificial Intelligence (MICAI'2004)*, Raúl Monroy, Gustavo Arroyo-Figueroa, Luis Enrique Sucar, and Humberto Sossa (Eds.). Springer Verlag. Lecture Notes in Artificial Intelligence Vol. 2972, 688–697.

[7] Kalyanmoy Deb and Himanshu Jain. 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation* 18, 4 (August 2014), 577–601.

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

[9] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. 2005. Scalable Test Problems for Evolutionary Multiobjective Optimization. In *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, Ajith Abraham, Lakhmi Jain, and Robert Goldberg (Eds.). Springer, USA, 105–145.

[10] K. Gerstl, G. Rudolph, O. Schütze, and H. Trautmann. 2011. 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)*. IEEE Press, Mérida, Yucatán, México, 975–980.

[11] Raquel Hernández Gómez and Carlos A. Coello Coello. 2013. MOMBI: A New Metaheuristic for Many-Objective Optimization Based on the R2 Indicator. In *2013 IEEE Congress on Evolutionary Computation (CEC'2013)*. IEEE Press, Cancún, México, 2488–2495. ISBN 978-1-4799-0454-9.

[12] Raquel Hernández Gómez and Carlos A. Coello Coello. 2015. Improved Metaheuristic Based on the R2 Indicator for Many-Objective Optimization. In *2015 Genetic and Evolutionary Computation Conference (GECCO 2015)*. ACM Press, Madrid, Spain, 679–686. ISBN 978-1-4503-3472-3.

[13] Simon Huband, Phil Hingston, Luigi Barone, and Lyndon While. 2006. A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. *IEEE Transactions on Evolutionary Computation* 10, 5 (October 2006), 477–506.

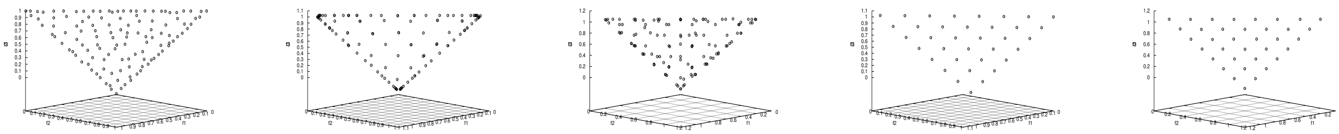
[14] Hisao Ishibuchi, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. 2015. Modified Distance Calculation in Generational Distance and Inverted Generational Distance. In *Evolutionary Multi-Criterion Optimization, 8th International Conference, EMO 2015*, António Gaspar-Cunha, Carlos Henggeler Antunes, and Carlos Coello Coello (Eds.). Springer. Lecture Notes in Computer Science Vol. 9019, Guimarães, Portugal, 110–125.

[15] Hisao Ishibuchi, Noritaka Tsukamoto, and Yusuke Nojima. 2008. Evolutionary many-objective optimization: A short review. In *2008 Congress on Evolutionary Computation (CEC'2008)*. IEEE Service Center, Hong Kong, 2424–2431.

[16] H. W. Kuhn and Bryn Yaw. 1955. The Hungarian method for the assignment problem. *Naval Res. Logist. Quart* (1955), 83–97.

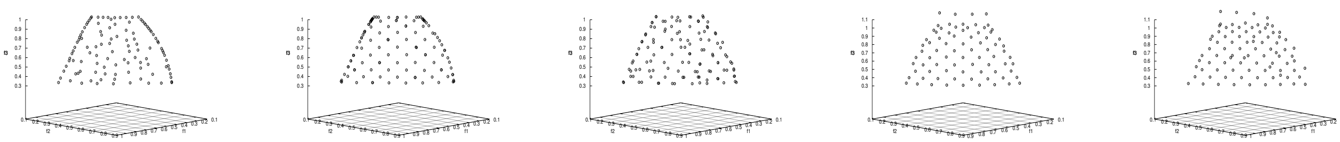
[17] Ke Li, Kalyanmoy Deb, Qingfu Zhang, and Sam Kwong. 2015. An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition. *IEEE Transactions on Evolutionary Computation* 19, 5 (October 2015), 694–716.

[18] Edgar Manoatl Lopez and Carlos A. Coello Coello. 2016. IGD$^+$-EMOA: A Multi-Objective Evolutionary Algorithm based on IGD$^+$. In *2016 IEEE Congress on Evolutionary Computation (CEC'2016)*. IEEE Press, Vancouver, Canada, 999–1006. ISBN 978-1-5090-0623-9.



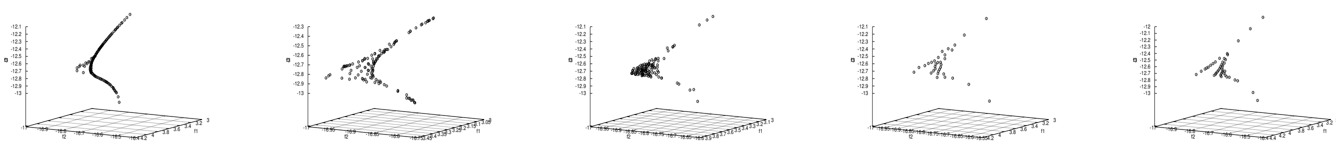
(a) IGD$^+$-EMOA (b) MOMBI-II (c) NSGA-III (d) RVEA (e) MOEA/DD

Figure 1: Graphical representation of the final set of solutions obtained by each MOEA on MAF1 with 3 objectives



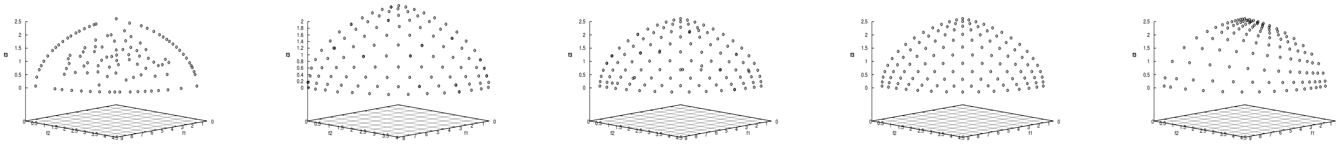
(a) IGD$^+$-EMOA (b) MOMBI-II (c) NSGA-III (d) RVEA (e) MOEA/DD

Figure 2: Graphical representation of the final set of solutions obtained by each MOEA on MAF2 with 3 objectives



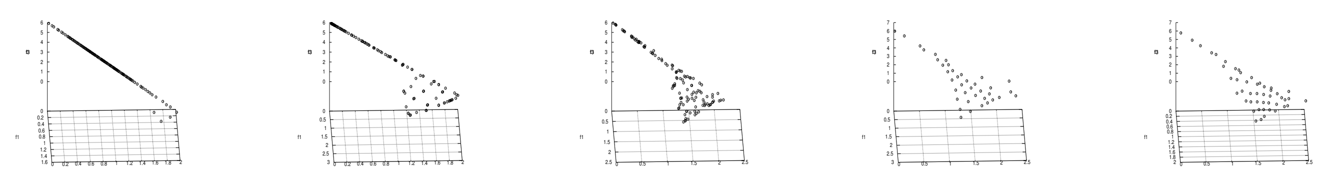
(a) IGD$^+$-EMOA (b) MOMBI-II (c) NSGA-III (d) RVEA (e) MOEA/DD

Figure 3: Graphical representation of the final set of solutions obtained by each MOEA on VNT2 with 3 objectives



(a) IGD$^+$-EMOA (b) MOMBI-II (c) NSGA-III (d) RVEA (e) MOEA/DD

Figure 4: Graphical representation of the final set of solutions obtained by each MOEA on MAF5 with 3 objectives



(a) IGD$^+$-EMOA (b) MOMBI-II (c) NSGA-III (d) RVEA (e) MOEA/DD

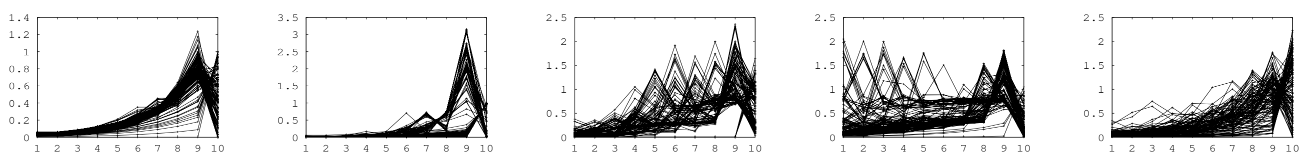
Figure 5: Graphical representation of the final set of solutions obtained by each MOEA on WFG3 with 3 objectives

Table 3: Performance comparison among several MOEAs using the average hypervolume indicator obtained from 30 independent executions solving 18 benchmark problems.

Problems	MOMBI-II	RVEA	MOEA/DD	IGD$^+$-EMOA	NSGA-III
DTLZ1	0.96622 (0.000001) +	0.66911 (0.000152) +	0.97379 (0.000000) ≈	**0.97381 (0.000000)**	0.96256 (0.001064) +
DTLZ2	7.36755 (0.000028) +	7.42224 (0.000000) +	7.42225 (0.000000) +	**7.42736 (0.000016)**	7.41893 (0.000000) +
DTLZ3	7.38843 (0.000084) -	7.40582 (0.000084) -	**7.4118 (0.000047)** -	7.2211 (0.006505)	7.38048 (0.000258) -
DTLZ4	7.3593 (0.036144) -	**7.42226 (0.000000)** -	7.42224 (0.000000) -	6.76942 (0.236529)	7.10506 (0.227356) ≈
DTLZ5	6.00978 (0.000000) +	5.9632 (0.000369) +	6.02456 (0.000062) +	**6.1033 (0.000000)**	5.84002 (0.05518) +
DTLZ6	5.79608 (0.00523) +	5.13815 (0.016264) +	5.6037 (0.006442) +	**5.92189 (0.005444)**	5.49135 (0.023354) +
DTLZ7	**13.37473 (0.000091)** -	13.0605 (1.283746) ≈	12.99409 (0.015542) ≈	12.37989 (2.217964)	13.32733 (0.002554) ≈
VIE1	61.44939 (0.000533) +	60.51323 (0.011862) +	60.55111 (0.021176) +	**61.98616 (0.000514)**	61.19214 (0.011932) +
VIE2	7.79702 (0.000001) +	7.7712 (0.000368) +	7.80468 (0.000037) +	**7.84583 (0.000012)**	7.77446 (0.000935) +
VIE3	15.11767 (0.000262) +	15.03082 (0.000422)+	15.06016 (0.000114) +	**15.16248 (0.000258)**	15.12629 (0.000502) +
MAF1	5.44926 (0.000019) +	5.37408 (0.000659) +	5.37139 (0.00009) +	**5.50322 (0.000193)**	5.4129 (0.000875) +
MAF2	5.08952 (0.000056) +	**5.1583 (0.000058)** -	5.11373 (0.000003) +	5.13305 (0.000019)	5.09758 (0.000043) +
MAF3	7.90637 (0.000043) -	**7.91154 (0.004847)** -	7.64261 (1.915744) +	7.79256 (0.020172)	7.89441 (0.00452) -
MAF4	**84.87316 (0.151259)** ≈	83.53436 (29.511151) ≈	51.80943 (1120.296924) +	84.46979 (2.762969)	83.73257 (1.377427) ≈
MAF5	95.97704 (52.294491) +	96.66782 (53.122845) +	96.95207 (0.017991) +	**98.27521 (0.000549)**	88.72762 (237.475764) +
WFG1	50.38691 (7.353216) ≈	**51.68413 (5.001739)** ≈	41.77398 (7.334821) +	48.54235 (50.414084)	44.95726 (10.36034) +
WFG2	48.72516 (12.06217) -	**51.14414 (0.045119)** -	44.23925 (3.146579) +	45.58634 (7.395813)	48.14747 (12.622738) ≈
WFG3	24.28138 (0.007298) ≈	22.12339 (0.086504) +	21.04349 (0.178677) +	**24.34142 (0.015209)**	23.54542 (0.037132) +

Table 4: Performance comparison among several MOEAs using the average hypervolume indicator obtained from 30 independent executions solving DTLZ5 with 3 up to 10 objectives.

m	MOMBI-II	RVEA	MOEA/DD	IGD$^+$-EMOA	NSGA-III
3	6.00978 (0.000000) +	5.9632 (0.000369) +	6.02456 (0.000062) +	**6.1033 (0.000000)**	5.84002 (0.05518) +
5	21.79297 (0.001846) +	23.30876 (0.003765) +	20.09359 (0.702607) +	**23.48784 (0.006147)**	20.73826 (0.862548) +
8	167.75169 (1.3931) +	166.28627 (311.052362) +	144.38434 (32.923411) +	**180.36716 (24.078975)**	108.75498 (416.448268) +
10	686.39987 (203.794132) +	551.2682 (9825.611555) +	489.97082 (805.137545) +	**718.77776 (8.701444)**	358.05289 (19458.130567) +



(a) IGD$^+$-EMOA **(b) MOMBI-II** **(c) NSGA-III** **(d) RVEA** **(e) MOEAD/DD**

Figure 6: Graphical representation of the final set of solutions obtained by the five MOEAs used in our study on DTLZ5 with 10 objectives.

[19] Edgar Manoatl Lopez and Carlos A. Coello Coello. 2017. Improving the Integration of the IGD+ Indicator into the Selection Mechanism of a Multi-objective Evolutionary Algorithm. In *2017 IEEE Congress on Evolutionary Computation (CEC'2017)*. IEEE Press, San Sebastián, Spain, 2683–2690. ISBN 978-1-5090-4601-0.

[20] Adriana Menchaca-Mendez, Carlos Hernández, and Carlos A. Coello Coello. 2016. Δ_p-MOEA: A New Multi-Objective Evolutionary Algorithm Based on the Δ_p Indicator. In *2016 IEEE Congress on Evolutionary Computation (CEC'2016)*. IEEE Press, Vancouver, Canada, 3753–3760. ISBN 978-1-5090-0623-9.

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

[22] Oliver Schütze, Xavier Esquivel, Adriana Lara, and Carlos A. Coello Coello. 2012. Using the Averaged Hausdorff Distance as a Performance Measure in Evolutionary Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation* 16, 4 (August 2012), 504–522.

[23] Heike Trautmann, Günter Rudolph, Christian Dominguez-Medina, and Oliver Schütze. 2012. Finding Evenly Spaced Pareto Fronts for Three-Objective Optimization Problems. In *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*, Oliver Schütze, Carlos A. Coello Coello, Alexandru-Adrian Tantar, Emilia Tantar, Pascal Bouvry, Pierre Del Moral, and Pierrick Legrand (Eds.). Springer, Advances in Intelligent Systems and Computing Vol. 175, Berlin, Germany, 89–105. ISBN 978-3-642-31519-0.

[24] David A. Van Veldhuizen. 1999. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations.* Ph.D. Dissertation. Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, USA.

[25] Saúl Zapotecas Martínez, Víctor A. Sosa Hernández, Hernán Aguirre, Kiyoshi Tanaka, and Carlos A. Coello Coello. 2014. Using a Family of Curves to Approximate the Pareto Front of a Multi-Objective Optimization Problem. In *Parallel Problem Solving from Nature - PPSN XIII, 13th International Conference*, Thomas Bartz-Beielstein, Jürgen Branke, Bogdan Filipič, and Jim Smith (Eds.). Springer. Lecture Notes in Computer Science Vol. 8672, Ljubljana, Slovenia, 682–691.

[26] Qingfu Zhang and Hui Li. 2007. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation* 11, 6 (December 2007), 712–731.

[27] Eckart Zitzler and Simon Künzli. 2004. Indicator-based Selection in Multiobjective Search. In *Parallel Problem Solving from Nature - PPSN VIII*, Xin Yao et al. (Ed.). Springer-Verlag. Lecture Notes in Computer Science Vol. 3242, Birmingham, UK, 832–842.