

Received July 16, 2021, accepted July 30, 2021, date of publication August 2, 2021, date of current version August 10, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3101899

Preference-Based Evolutionary Multiobjective Optimization Through the Use of Reservation and Aspiration Points

SANDRA GONZÁLEZ-GALLARDO¹, RUBÉN SABORIDO²,
ANA B. RUIZ¹, AND MARIANO LUQUE¹

¹Department of Applied Economics (Mathematics), Campus EL Ejido, University of Málaga, 29071 Malaga, Spain

²ITIS Software, Campus Teatinos, University of Málaga, 29071 Malaga, Spain

Corresponding author: Mariano Luque (mluque@uma.es)

This work was supported in part by the Spanish Ministry of Science and Innovation under Grant PID2020-115429GB-I00, in part by the Andalusian Regional Ministry of Economy, Knowledge, Business and University [Andalusian Plan for Research, Development and Innovation (PAI Group)] under Grant SEJ-532 and Grant UMA18-FEDERJA-024, and in part by the University of Malaga under Project B1-2020_18 and Project B1-2020_1. The work of Sandra González-Gallardo was supported by the Sistema Nacional de Garantía Juvenil y del Programa Operativo de Empleo Juvenil 2014–2020–Fondos European Regional Development Fund (FEDER). The work of Rubén Saborido was supported by the Juan de la Cierva Grant through the Spanish State Research Agency under Grant FJC2018-038537-1.

ABSTRACT Preference-based Evolutionary Multiobjective Optimization (EMO) algorithms approximate the region of interest (ROI) of the Pareto optimal front defined by the preferences of a decision maker (DM). Here, we propose a preference-based EMO algorithm, in which the preferences are given by means of aspiration and reservation points. The aspiration point is formed by objective values which the DM wants to achieve, while the reservation point is constituted by values for the objectives not to be worsened. Internally, the first generations are performed in order to generate an initial approximation set according to the reservation point. Next, in the remaining generations, the algorithm adapts the search for new non-dominated solutions depending on the dominance relation between the solutions obtained so far and both the reservation and aspiration points. This allows knowing if the given points are achievable or not; this type of information cannot be known before the solution process starts. On this basis, the algorithm proceeds according to three different scenarios with the aim of re-orienting the search directions towards the ROI formed by the Pareto optimal solutions with objective values within the given aspiration and reservation values. Computational results show the potential of our proposal in 2, 3 and 5-objective test problems, in comparison to other state-of-the-art algorithms.

INDEX TERMS Evolutionary multiobjective optimization, preferences, aspiration and reservation points, weight vectors.

I. INTRODUCTION

Multiobjective optimization problems (MOPs) arising in many real-world applications contain several conflicting objectives $f_i : S \rightarrow \mathbb{R}$, with $i = 1, \dots, k$ ($k \geq 2$), that need to be optimized simultaneously over a *feasible set* in the decision space, $S \subset \mathbb{R}^n$, formed by *decision vectors* $\mathbf{x} = (x_1, \dots, x_n)^T$. Generally, they can be formulated as follows:

$$\begin{aligned} & \text{minimize } \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})\} \\ & \text{subject to } \mathbf{x} \in S. \end{aligned} \quad (1)$$

The associate editor coordinating the review of this manuscript and approving it for publication was Jagdish Chand Bansal.

All *objective vectors* $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_k(\mathbf{x}))^T$ constitute the *objective set* in the objective space $Z = \mathbf{f}(S) \subset \mathbb{R}^k$.

Usually, it is not possible to find a single solution where all the objectives can achieve their individual optima, given that they are in conflict. Instead, the so-called *Pareto optimal solutions* have to be found, in which none of the objectives can be improved without degrading, at least, one of the others. A solution $\mathbf{x} \in S$ is said to be *Pareto optimal* of problem (1) if and only if there is no other $\bar{\mathbf{x}} \in S$ such that $f_i(\bar{\mathbf{x}}) \leq f_i(\mathbf{x})$ for all $i = 1, \dots, k$, and $f_j(\bar{\mathbf{x}}) < f_j(\mathbf{x})$ for at least one index j . The corresponding objective vector $\mathbf{f}(\mathbf{x})$ is referred to as a *Pareto optimal objective vector*. The set of all Pareto optimal solutions is called the *Pareto optimal set* (PS), denoted by E ,

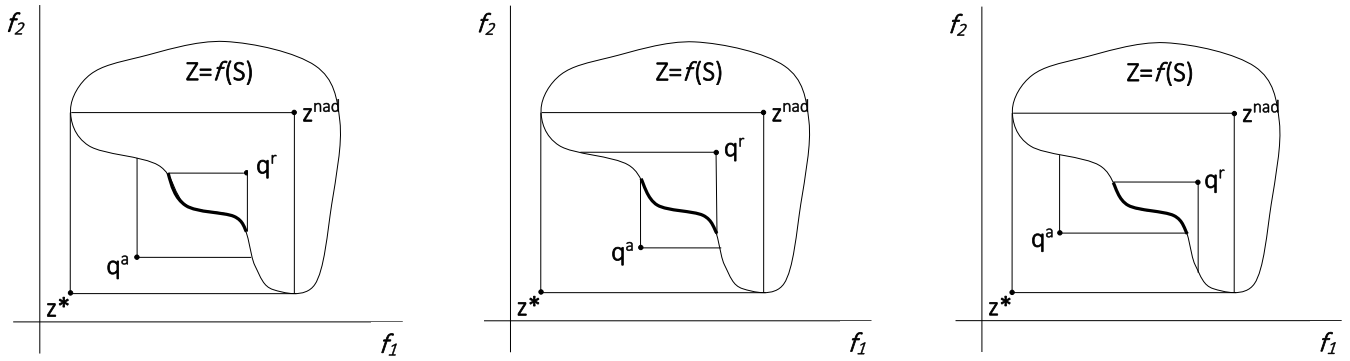


FIGURE 1. ROI defined by different reservation and aspiration points.

while all the Pareto optimal objective vectors form the *Pareto optimal front* (PF), denoted by $\mathbf{f}(E)$. In addition, for two vectors $\mathbf{z}, \bar{\mathbf{z}} \in \mathbb{R}^k$, we say that \mathbf{z} *dominates* $\bar{\mathbf{z}}$ if and only if $z_i \leq \bar{z}_i$ for all $i = 1, \dots, k$, and $z_j < \bar{z}_j$ for, at least, one index j . If \mathbf{z} and $\bar{\mathbf{z}}$ do not dominate each other, it is said that they are *non-dominated*. By *non-dominated set* we refer to a set of solutions in the decision space whose objective vectors are not dominated by the other solutions in the set.

The bounds of the objective functions in the objective space are defined by their worst and their best possible values, which constitute the *nadir point* $\mathbf{z}^{\text{nad}} = (z_1^{\text{nad}}, \dots, z_k^{\text{nad}})^T$ and the *ideal point* $\mathbf{z}^* = (z_1^*, \dots, z_k^*)^T$, respectively. To be more precise, the upper objective function bounds are calculated in the nadir point as $z_i^{\text{nad}} = \max_{\mathbf{x} \in E} f_i(\mathbf{x})$ ($i = 1, \dots, k$), while the lower bounds can be obtained with the ideal point as $z_i^* = \min_{\mathbf{x} \in E} f_i(\mathbf{x}) = \min_{\mathbf{x} \in S} f_i(\mathbf{x})$ ($i = 1, \dots, k$).

Pareto optimal solutions are incomparable in a mathematical sense and, to find the final solution of the MOP, a *decision maker* (DM) must be involved in the solution process in order to choose the Pareto optimal solution which best suits her/his preferences (the *most preferred solution*).

Evolutionary Multiobjective Optimization (EMO) algorithms [2], [3] are developed for solving MOPs. These algorithms generate a non-dominated set to approximate the entire PF, meaning that ideally the solutions generated should be uniformly distributed (diversity) and close enough to the PF (convergence). However, achieving convergence and diversity simultaneously is not always easy, especially when handling many-objective optimization problems ($k > 3$), since e.g. the percentage of non-dominated solutions increases with the number of objectives, or in problems with complicated PFs (such as e.g. non-convex, degenerated, or discontinuous). To cope with these issues, *decomposition-based* EMO algorithms transform the original MOP into a set of single-objective optimization sub-problems formulated using an aggregation function with one weight vector, and usually select the best individuals at each generation according to these sub-problems. Some examples of these algorithms are GWASF-GA [22], MOEA/D [14], [29], and NSGA-III [5].

The approximation found by an EMO algorithm can be useful to gain an insight into the objective functions (their ranges, their trade-offs, their degree of conflict, etc.). However, the task of identifying the most preferred Pareto optimal solution may be too demanding for a DM if (s)he has to analyze and compare all the solutions generated by an EMO algorithm, whose approximation set usually consists of a large number thereof. To facilitate this decision-making task, preferences can be included in the solution process, resulting in the so-called *preference-based EMO algorithms* [1]. Their main purpose consists of only approximating the subset of solutions which correspond to the preferences of the DM (the *region of interest* (ROI)) [6], [8], [25]. Within this group, note can be made of the WASF-GA algorithm [21], at which the preferences are elicited by means of a reference point $\mathbf{q} = (q_1, \dots, q_k)^T$ constituted by desirable values q_i for the objective functions f_i ($i = 1, \dots, k$). In practice, a reference point \mathbf{q} can be *achievable* if \mathbf{q} is equal or dominated by a Pareto optimal objective vector (that is, the desirable values can be simultaneously achieved or improved by a Pareto optimal solution), or *unachievable* in any other case.

Once a reference point is given as preference information, the ROI with the most interesting Pareto optimal solutions according to these preferences can be defined as suggested in [21]. To be more precise, we assume that the *ROI associated with a reference point* \mathbf{q} is defined as follows: when \mathbf{q} is achievable, the ROI is the subset of Pareto optimal objective vectors dominating \mathbf{q} , and if \mathbf{q} is unachievable, the Pareto optimal objective vectors dominated by \mathbf{q} constitute the ROI.

Besides using preferences, WASF-GA is an algorithm based on decomposition. It considers a set of weight vectors to internally formulate single-objective problems using the following achievement scalarizing function [27]:

$$s(\mathbf{q}, \mathbf{f}(\mathbf{x}), \mu) = \max_{i=1, \dots, k} \{ \mu_i(f_i(\mathbf{x}) - q_i) \} + \rho \sum_{i=1}^k \mu_i(f_i(\mathbf{x}) - q_i), \tag{2}$$

where \mathbf{q} is a reference point provided by the DM, $\mu = (\mu_1, \dots, \mu_k)^T$ with $\mu_i > 0$ for $i = 1, \dots, k$ denotes the

vector of weights, and $\rho > 0$ is a parameter to assure that the solution that minimizes (2) over the feasible set S is always a Pareto optimal solution to the original MOP (1). See [18] for further details.

Recently, IRA-EMO [23] has been proposed as an interactive EMO method based on WASF-GA. In this method, the preferences are provided in the form of preferred ranges for the objective functions. Given the multiobjective minimizing problem (1), for every $i = 1, \dots, k$, the DM specifies an aspiration level q_i^a (the lower desirable bound) and a reservation level q_i^r (the upper desirable bound) for the values of the objective f_i . It is implicitly assumed that $z_i^* \leq q_i^a < q_i^r \leq z_i^{\text{nad}}$ ($i = 1, \dots, k$). With this information, two reference points can be built: an *aspiration point* $\mathbf{q}^a = (q_1^a, \dots, q_k^a)^T$ formed by values regarded as desirable for the objective functions (i.e. values to be achieved, if possible), and a *reservation point* $\mathbf{q}^r = (q_1^r, \dots, q_k^r)^T$ comprised of acceptable objective levels beyond which their values should be rejected. This preference scheme allows the DM to freely investigate different parts of the PF by modifying the desirable bounds for the objective functions (i.e. the aspiration and reservation points).

In IRA-EMO, Modified WASF-GA is the algorithm internally defined to generate a set of non-dominated solutions according to both points. However, this algorithm is designed to approximate the union set of the ROIs associated with \mathbf{q}^a and \mathbf{q}^r , using a small number of solutions. Nonetheless, when providing preferred objective function ranges, the DM may only be interested in the Pareto optimal solutions whose objective values are between the corresponding aspiration and the reservation levels (i.e. in the solutions lying in the intersection set of the ROIs defined by the two points).

In this paper, we focus on the use of preferences in EMO to reduce the search space to the ROI that best suits the desires of the DM. We assume that the preferences are expressed in the form of aspiration and reservation points, \mathbf{q}^a and \mathbf{q}^r . As mentioned above, when giving this information the DM wants to analyze solutions with objective function values within the desired bounds, i.e. among the aspiration and the reservation levels. Thus, the ROI to be approximated, denoted by R , includes the Pareto optimal solutions which belong, at the same time, to the ROIs associated with both points. If we denote by R^a and R^r the ROIs defined by the aspiration and the reservation points, respectively, the subset R with the most interesting Pareto optimal solutions for the DM can be obtained as $R = R^r \cap R^a$. Figure 1 shows different situations for a bi-objective minimization problem, where R is the region of the PF in bold.

To converge to this ROI, we propose a decomposition-based algorithm enabling the simultaneous use of both the aspiration and reservation points, which improves the convergence procedure of the algorithms previously cited. The main purpose is to directly focus the search for new solutions onto the ROI jointly defined by the two points, avoiding producing solutions outside this region, and to do so not only

with a few solutions but also with a sufficiently representative approximation set.

Our proposal is called *Evolutionary algorithm based on Reservation-Aspiration Levels* (ERAL). Using points \mathbf{q}^a and \mathbf{q}^r given by the DM and an initial set of weight vectors, ERAL produces an initial approximation of the ROI defined by \mathbf{q}^r in the first generations. Using the solutions obtained, the achievability of the reservation and aspiration points is investigated based on the relation of dominance between these solutions and the two points. Then, the algorithm procedure is adjusted accordingly to the scenario detected in order to converge to the subset R jointly defined by the aspiration and reservation points.

Moreover, it is well-known that the weight vectors' distribution directly affects the performance of any decomposition-based EMO algorithm, since the diversity of the population is somehow controlled by the set of weight vectors used. In problems with complicated PFs, using a well-distributed set of weight vectors does not guarantee the generation of a well-diversified approximation set of the PF. In general, the best results are obtained when the distribution of the weight vectors is consistent with the shape of the PF, but the PF's characteristics are usually unknown beforehand. To alleviate the effects of using a pre-fixed set of weight vectors, there are many proposals to update the distribution of the weights as the algorithm converges, such as e.g. [9], [11], [13], [16], [17], [20], [24]. In this line, the ERAL algorithm also incorporates the possibility of dynamically adapting the weight vectors according to the PF's shape in the last generations. Specifically, we have considered two procedures to adjust the weights: the dynamic adjustment of weight vectors proposed in [17] and the approach to update weights during an evolutionary process called AdaW [16].

The new algorithm ERAL is described below in Section II. Next, the computational experiment carried out to show the potential of our proposal is shown in Section III. Finally, the conclusions are drawn in Section IV.

II. EVOLUTIONARY ALGORITHM BASED ON RESERVATION AND ASPIRATION LEVELS

A. MOTIVATION

As mentioned above, according to the aspiration and reservation levels given by a DM, Modified WASF-GA [23] generates an approximation set of the region $R^r \cup R^a$, instead of directly approximating the intersection set $R = R^r \cap R^a$. This may not entail a significant waste of computational effort if the purpose is to generate a small number of solutions, as in IRA-EMO [23], although this also depends on the complexity of the problem considered. Nevertheless, there are more efficient ways to approximate R and avoid producing solutions outside it. This motivated us to propose the *Evolutionary algorithm based on Reservation and Aspiration Levels* (ERAL), as opposed to Modified WASF-GA, directly focuses the search for new non-dominated solutions on the subset R .

Our proposal is a decomposition-based algorithm that internally considers the ASF given in (2) to converge to the desired region R . Due to the practical meaning of minimizing this ASF over the feasible set,¹ it is important that the weight vectors considered are defined so that they produce projection directions that allow finding a well-diversified approximation of the PF [22], or of the subset of solutions to be approximated [21]. Therefore, in ERAL, a set of weight vectors are initially generated so that they define uniformly distributed projection directions.

The first generations of ERAL are performed using the weight vectors initially generated and the reservation point. Once an approximation set has been produced, the algorithm's working procedure is reconsidered depending on the dominance relation of the non-dominated solutions obtained so far with respect to both the reservation and aspiration points. This dominance information allows us to know the achievability of the aspiration and reservation points: whether both points are of the same type (achievable or unachievable) or if the aspiration point is unachievable and the reservation point is achievable. Let us remark that this information is unknown before the algorithm starts; indeed, even the DM is not able to foresee the achievability of both points when providing them. Then, the algorithm adopts different strategies to perform the remaining generations according to the knowledge gained in this respect, with the aim of guiding the search as best as possible towards the subset R to be approximated, which changes depending on the achievability of the two points.

In addition, the ERAL algorithm also allows the weight vectors to be dynamically adapted according to the PF's characteristics. As mentioned above, in decomposition-based EMO algorithms, the diversity of the final population is somehow controlled by the weight vectors, and a pre-fixed set of weight vectors may not lead to a uniformly distributed set of solutions in the region to be approximated [11], [13]. Several papers propose internal mechanisms to adapt the weight vectors along the convergence process [11], [16], [17], [20], [24]. Following a similar idea, the vectors of weights initially generated in ERAL can be internally updated. For this purpose, the weight vector adjustment procedures suggested in [16] and [17] can be executed in the last generations of the algorithm to better adjust the search directions towards R , according to the specific features of the PF associated with the problem being solved.

B. THE ALGORITHM

Next, the ERAL algorithm is described in detail. Let \mathbf{q}^r and \mathbf{q}^a denote the reservation and the aspiration points given by the DM, respectively. As said before, it may happen that \mathbf{q}^r is achievable and \mathbf{q}^a is unachievable, but it is also possible that \mathbf{q}^r and \mathbf{q}^a are both achievable, or that both of them

¹Minimizing the ASF given in (2) over S means, in practice, to project \mathbf{q} onto the PF in the projection direction defined by the inverse components of the weight vector used (i.e. by $(\frac{1}{\mu_1}, \dots, \frac{1}{\mu_k})$).

are unachievable, but this information is unknown at the beginning.

As already said, a set of weight vectors is used to internally decompose the original MOP into various sub-problems. Let $W = \{\boldsymbol{\mu}^1, \dots, \boldsymbol{\mu}^{N_\mu}\}$ be the set of weight vectors in $(0, 1)^k$ used, where $N_\mu > 0$ is the number of vectors considered. These weight vectors are generated so that their inverse components (which define their search or projection directions in the objective space) are evenly distributed in the weight vector space. Besides, let us denote by N the population size and by G_T the total number of generations.

The first generations of ERAL are performed as in the original WASF-GA [21], using the weight vectors in W and \mathbf{q}^r as reference point. If G_p denotes the number of generations run using \mathbf{q}^r as reference point, we can define it as $G_p = p \cdot G_T$, where $0 < p < 1$ is a real number. In practice, the purpose of executing these G_p generations is to produce an initial approximation of the ROI only associated with \mathbf{q}^r . Thus, G_p cannot be too small in order not to stop the algorithm too prematurely, nor too large so that there are still generations to improve the convergence towards the subset R (delimited by both \mathbf{q}^r and \mathbf{q}^a) before the process finalizes.

Once these G_p generations have been completed, we check the dominance relation of the non-dominated solutions obtained with respect to both the aspiration and reservation points. Let P^{G_p} denote the population of non-dominated solutions at generation G_p . Internally, we build the following subsets:

- Subset of solutions whose objective vectors dominate the reservation point:

$$\mathbf{D}^r = \{\mathbf{x} \in P^{G_p} / \mathbf{f}(\mathbf{x}) \text{ dominates } \mathbf{q}^r\}. \quad (3)$$

- Subset of solutions whose objective vectors dominate the aspiration point:

$$\mathbf{D}^a = \{\mathbf{x} \in P^{G_p} / \mathbf{f}(\mathbf{x}) \text{ dominates } \mathbf{q}^a\}. \quad (4)$$

- Subset of solutions whose objective vectors dominate the reservation point and, at the same time, are dominated by the aspiration point:

$$\mathbf{D}^{r,a} = \{\mathbf{x} \in P^{G_p} / \mathbf{f}(\mathbf{x}) \text{ dominates } \mathbf{q}^r \text{ and is dominated by } \mathbf{q}^a\}. \quad (5)$$

Depending on whether there are solutions or not in \mathbf{D}^a , \mathbf{D}^r and $\mathbf{D}^{r,a}$, three different scenarios can be distinguished that tell us if \mathbf{q}^a and \mathbf{q}^r are achievable or not. Accordingly, the algorithm adapts its working procedure to perform the remaining number of generations (i.e. the last $G_{1-p} = (1-p) \cdot G_T = G_T - G_p$ generations as described next:

- **Scenario 1:** If $\mathbf{D}^r = \emptyset$, that is, none of the non-dominated solutions generated so far dominates the reservation point, then we can assume that both \mathbf{q}^r and \mathbf{q}^a are unachievable ($\mathbf{D}^r = \emptyset$ implies that $\mathbf{D}^a = \emptyset$). In this case, the ROI R to be approximated is defined just by the reservation point \mathbf{q}^r and the last G_{1-p} generations

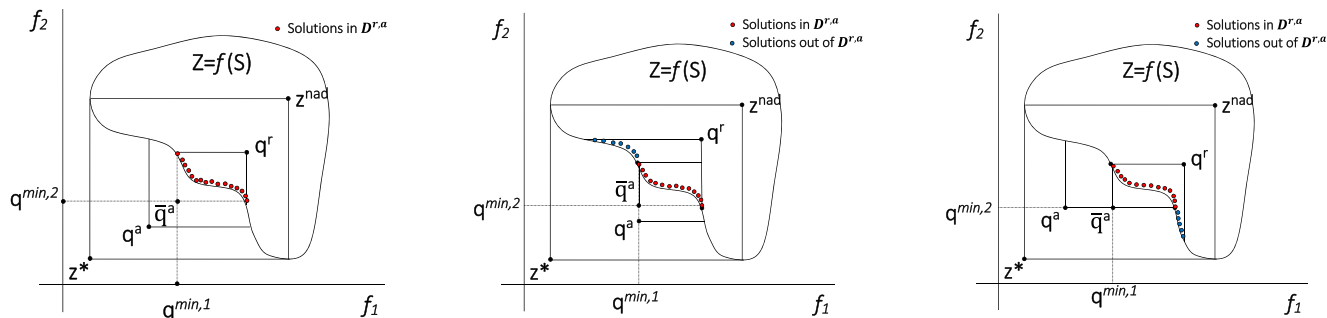


FIGURE 2. Scenario 3: q^a is unachievable and q^r achievable.

of ERAL are performed using in WASF-GA the same set of weight vectors W and q^r as a reference point.

- **Scenario 2:** If $D^a \neq \emptyset$, then, there exists at least one solution in P^{G_p} which dominates the aspiration point, meaning that both q^r and q^a are achievable ($D^a \neq \emptyset$ implies that $D^r \neq \emptyset$). In this situation, the subset R to be approximated is defined just by the aspiration point q^a and the last G_{1-p} generations of ERAL are executed using in WASF-GA the weight vectors in W and q^a as a reference point.
- **Scenario 3:** Whether $D^{r,a} \neq \emptyset$ (and this implies that $D^r \neq \emptyset$) and $D^a = \emptyset$, then, among the solutions generated at generation G_p , we find at least one solution dominating the reservation point that is also dominated by aspiration point q^a ; at the same time, there are no solutions dominating q^a . Thus, this implies that q^r is achievable and q^a is unachievable. In this case, the way to proceed with the remaining G_{1-p} generations is the following one. Let us consider that $q^{min,i}$ is the (best) minimum value achieved by the objective function f_i among the solutions in the subset $D^{r,a}$ ($i = 1, \dots, k$). That is, $q^{min,i} = \min_{x \in D^{r,a}} f_i(x)$. Then, a new aspiration point is defined as $\bar{q}^a = (q^{min,1}, q^{min,2}, \dots, q^{min,k})$ and the last G_{1-p} generations of ERAL are performed as in WASF-GA using the weight vectors in W and \bar{q}^a as a reference point. Figure 2 shows several situations corresponding to this scenario. As can be observed, point \bar{q}^a is calculated so that it adjusts to the delimited region R (in bold).

To some extent, the non-dominated solutions produced at each generation of ERAL are the best individuals produced so far for minimizing the ASF given in (2), using the weight vectors in W , and with respect to either q^r , q^a or \bar{q}^a , depending on the generation being executed, as previously described in the three scenarios.

Last but not least, the weight vectors in the set W can be updated, if desired, so that the convergence towards R is performed taking into account the PF's characteristics. In the last G_{1-p} generations (corresponding to the three scenarios), the procedures suggested in [16], [17] can be applied to re-calculate the weight vectors during the optimization process based on the non-dominated solutions already obtained,

which are likely to reflect the shape of the PF since they are gradually converging towards it. The idea underlying these weight vectors' adaptations consist of replacing the weight vectors generating solutions in overcrowded areas by new ones pushing the search towards the parts with a lack of solutions. The main differences between both procedures lie in the way to identify the vectors that will be removed and to calculate the new vectors that will replace the removed ones. For more details, see [16], [17]. In ERAL, these approaches are applied to adapt all the weight vectors in W only once, when 10% of the remaining generations have been performed (i.e. at generation $G_p + 0.1 \cdot G_{1-p}$). Note that any other procedure allowing us to update the weight vectors can also be used in ERAL.

To summarize, Algorithm 1 contains ERAL's main steps. We denote by h the current generation, P^h the population of individuals at generation h , Q^h the offspring population at generation h , Z^h the combined population of parents and offspring at generation h , F_n^h the n -th front at generation h and P_F the final set of non-dominated solutions generated. Algorithms 2, 3, 4 individually describe the steps followed in the three different scenarios.

Finally, we would like to say that, in case the DM is only interested in providing one of the two points, ERAL can also be applied just with the aspiration point or the reservation one. This option may be suitable if one needs to relax the cognitive burden required by using our algorithm to solve a specific MOP. When the DM only specifies an aspiration point, an approximation to the nadir point can be considered as the reservation point, while if only a reservation point is provided, an approximation to the ideal point can be used as the aspiration point.

III. COMPUTATIONAL EXPERIMENTS

A. EXPERIMENTAL DESIGN

To the best of our knowledge, none of the existing reference point-based EMO algorithms (see e.g. [6], [8], [25]) allows working with aspiration and reservation points as our proposal does (in the sense that these two points define desirable lower and upper bounds for the objective function values). Note that algorithms enabling the use of several reference points, such as e.g. [6], [25], can be executed using the

Algorithm 1 Main Steps of ERAL

Require: The aspiration and reservation points \mathbf{q}^a and \mathbf{q}^r , a set of weight vectors $W = \{\mu^1, \dots, \mu^{N_\mu}\}$, the population size N , the number of generations G_T and the percentage p of generations initially executed with \mathbf{q}^r .

Ensure: A approximation set P_F of the region R .

- 1: Set $G_p = p \cdot G_T$, $G_{1-p} = (1-p) \cdot G_T$ and $h = 0$.
- 2: Randomly generate N solutions to form the initial population P^0 .
- 3: **while** $h \leq G_p$ **do**
- 4: Set $Z^h = \emptyset$ and $n = 0$.
- 5: Apply the selection, crossover and mutation operators to P^h to create a population Q^h with N new solutions. Set $Z^h = P^h \cup Q^h$.
- 6: For every $\mathbf{x} \in Z^h$, calculate $s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}), \mu^j)$ with $j = 1, \dots, N_\mu$.
- 7: **while** $Z^h \neq \emptyset$ **do**
- 8: Set $n = n + 1$ and initialize $F_n^h = \emptyset$.
- 9: **for** $j = 1, \dots, N_\mu$ **do**
- 10: Find $\mathbf{x}_j^n \in Z^h$ such that $s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}_j^n), \mu^j) = \min_{\mathbf{x} \in Z^h} s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}), \mu^j)$.
- 11: Update $F_n^h = F_n^h \cup \{\mathbf{x}_j^n\}$.
- 12: Remove \mathbf{x}_j^n from Z^h .
- 13: **end for**
- 14: **end while**
- 15: Set $P^{h+1} = \emptyset$ and $m = 1$.
- 16: **while** $\#(P^{h+1} \cup F_m^h) \leq N$ **do**
- 17: Update $P^{h+1} = P^{h+1} \cup F_m^h$.
- 18: Set $m = m + 1$.
- 19: **end while**
- 20: Until having N solutions in P^{h+1} , include into P^{h+1} the solutions corresponding to the lowest values in $\{s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}), \mu^j)\}_{j=1, \dots, N_\mu, \mathbf{x} \in F_m^h}$.
- 21: Update $h = h + 1$.
- 22: **end while**
- 23: Using the population P^{G_p} , obtain the subsets \mathbf{D}^r , \mathbf{D}^a and $\mathbf{D}^{r,a}$ as in (3), (4) and (5), respectively.
- 24: **if** $\mathbf{D}^r = \emptyset$ **then**
- 25: Execute **Algorithm 2** using \mathbf{q}^r , the set of weight vectors W , the remaining number of generations G_{1-p} and P^{G_p} as the population for the next generation. Set P_F as the outcome of this algorithm.
- 26: **end if**
- 27: **if** $\mathbf{D}^a \neq \emptyset$ **then**
- 28: Execute **Algorithm 3** using \mathbf{q}^a , the set of weight vectors W , the remaining number of generations G_{1-p} and P^{G_p} as the population for the next generation. Set P_F as the outcome of this algorithm.
- 29: **end if**
- 30: **if** $\mathbf{D}^{r,a} \neq \emptyset$ and $\mathbf{D}^a = \emptyset$ **then**
- 31: Execute **Algorithm 4** using $\bar{\mathbf{q}}^a$ obtained from the set of non-dominated solutions $\mathbf{D}^{r,a}$, the set of weight vectors W , the remaining number of generations G_{1-p} and P^{G_p} as the population for the next generation. Set P_F as the outcome of this algorithm.
- 32: **end if**

aspiration and reservation points as two different reference points, but would approximate the ROIs determined by the two points independently. That is, their working procedures are not designed to converge to the region $R = R^r \cap R^a$, as our algorithm does. Therefore, the existing reference point-based

Algorithm 2 Scenario 1

Require: A reservation point \mathbf{q}^r , a set of weight vectors $W = \{\mu^1, \dots, \mu^{N_\mu}\}$, the population size N , the remaining number of generations G_{1-p} , and an initial population P with N solutions.

Ensure: A approximation set P_F of the region R .

- 1: Set $h = 0$ and $P^0 = P$.
- 2: **while** $h \leq G_{1-p}$ **do**
- 3: Set $Z^h = \emptyset$ and $n = 0$.
- 4: Apply the selection, crossover and mutation operators to P^h and create a population Q^h with N new individuals. Set $Z^h = P^h \cup Q^h$.
- 5: For every $\mathbf{x} \in Z^h$, calculate $s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}), \mu^j)$ with $j = 1, \dots, N_\mu$.
- 6: **while** $Z^h \neq \emptyset$ **do**
- 7: Set $n = n + 1$ and initialize $F_n^h = \emptyset$.
- 8: **for** $j = 1, \dots, N_\mu$ **do**
- 9: Find $\mathbf{x}_j^n \in Z^h$ such that $s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}_j^n), \mu^j) = \min_{\mathbf{x} \in Z^h} s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}), \mu^j)$.
- 10: Update $F_n^h = F_n^h \cup \{\mathbf{x}_j^n\}$.
- 11: Remove \mathbf{x}_j^n from Z^h .
- 12: **end for**
- 13: **end while**
- 14: Set $P^{h+1} = \emptyset$ and $m = 1$.
- 15: **while** $\#(P^{h+1} \cup F_m^h) \leq N$ **do**
- 16: Update $P^{h+1} = P^{h+1} \cup F_m^h$.
- 17: Set $m = m + 1$.
- 18: **end while**
- 19: Until having N solutions in P^{h+1} , include into P^{h+1} the solutions corresponding to the lowest values in $\{s(\mathbf{q}^r, \mathbf{f}(\mathbf{x}), \mu^j)\}_{j=1, \dots, N_\mu, \mathbf{x} \in F_m^h}$.
- 20: Update $h = h + 1$.
- 21: **if** $h = 0.1 \cdot G_{1-p}$ **then**
- 22: If desired, perform an adaptation of all the weight vectors in W following [16], [17] and update W accordingly.
- 23: **end if**
- 24: **end while**
- 25: Set $P_F = F_1^{h-1}$.

EMO algorithms are not valid to perform a fair comparison against ERAL. In addition, we have not considered Modified WASF-GA in the computational tests because this approach only generates a few non-dominated solutions to approximate the region $R^r \cup R^a$, and not to directly converge to R , as already explained.

Owing to this, we have considered the EMO algorithms NSGA-III [5] and MOEA/D-DE [12] but, given that these algorithms directly approximate the whole PF, we have introduced the preference information in \mathbf{q}^a and \mathbf{q}^r as constraints bound to the objective functions in the problems' formulation in order to perform a fair comparison. That is, when running these algorithms (hereinafter denoted by cNSGA-III and cMOEA/D-DE, respectively), the following constraints are introduced into the formulation of each test problem considered: $q_i^a \leq f_i(\mathbf{x}) \leq q_i^r$, for each $i = 1, \dots, k$. This assures that they generate non-dominated solutions with objective function values within the desired values; that is, cNSGA-III and cMOEA/D-DE will approximate the region R defined by these two points.

Algorithm 3 Scenario 2

Require: An aspiration point \mathbf{q}^a , a set of weight vectors $W = \{\mu^1, \dots, \mu^{N_\mu}\}$, the population size N , the remaining number of generations G_{1-p} , and an initial population P with N solutions.

Ensure: A approximation set P_F of the region R .

```

1: Set  $h = 0$  and  $P^0 = P$ .
2: while  $h \leq G_{1-p}$  do
3:   Set  $Z^h = \emptyset$  and  $n = 0$ .
4:   Apply the selection, crossover and mutation operators to  $P^h$ 
   and create a population  $Q^h$  with  $N$  new individuals. Set  $Z^h = P^h \cup Q^h$ .
5:   For every  $\mathbf{x} \in Z^h$ , calculate  $s(\mathbf{q}^a, \mathbf{f}(\mathbf{x}), \bar{\mu}^j)$  with  $j = 1, \dots, N_\mu$ .
6:   while  $Z^h \neq \emptyset$  do
7:     Set  $n = n + 1$  and initialize  $F_n^h = \emptyset$ .
8:     for  $j = 1, \dots, N_\mu$  do
9:       Find  $\mathbf{x}_j^n \in Z^h$  such that  $s(\mathbf{q}^a, \mathbf{f}(\mathbf{x}_j^n), \mu^j) = \min_{\mathbf{x} \in Z^h} s(\mathbf{q}^a, \mathbf{f}(\mathbf{x}), \mu^j)$ .
10:      Update  $F_n^h = F_n^h \cup \{\mathbf{x}_j^n\}$ .
11:      Remove  $\mathbf{x}_j^n$  from  $Z^h$ .
12:    end for
13:  end while
14:  Set  $P^{h+1} = \emptyset$  and  $m = 1$ .
15:  while  $\#(P^{h+1} \cup F_m^h) \leq N$  do
16:    Update  $P^{h+1} = P^{h+1} \cup F_m^h$ .
17:    Set  $m = m + 1$ .
18:  end while
19:  Until having  $N$  solutions in  $P^{h+1}$ , include into  $P^{h+1}$ 
  the solutions corresponding to the lowest values in  $\{s(\mathbf{q}^a, \mathbf{f}(\mathbf{x}), \mu^j)\}_{j=1, \dots, N_\mu, \mathbf{x} \in F_m^h}$ .
20:  Update  $h = h + 1$ .
21:  if  $h = 0.1 \cdot G_{1-p}$  then
22:    If desired, perform an adaptation of all the weight vectors
    in  $W$  following [16], [17] and update  $W$  accordingly.
23:  end if
24: end while
25: Set  $P_F = F_1^{h-1}$ .

```

Moreover, in the comparison, we also execute two variants of ERAL at which dynamic adaptations of the weight vectors are carried out while the algorithm is converging. These two variants are referred to as Adapt-ERAL and AdaW-ERAL, depending on whether the weight vectors' adaptation is done following [17] or [16], respectively.

The test problems involved in our study are from the families DTLZ [7], UF [30], WFG [10], and ZDT [31]. We have considered problems with two, three and five objective functions. These test problems have various properties, such as having a linear, multi-modal, concave, discontinuous, or degenerate PF, having a multi-modal, biased, or deceptive search space, and/or having strong-linkage decision variables [15]. Therefore, approximating their PFs represents a challenge for any of the algorithms included in the comparison. In Table 1, we give a detailed description of the main features of these problems, and we also indicate the number of objectives (k) considered for each problem. In addition, we use $k + 4$, $k + 9$, and $k + 19$ decision variables for the DTLZ1, DTLZ2-DTLZ4, and DTLZ7

Algorithm 4 Scenario 3

Require: A set of non-dominated solutions $\mathbf{D}^{r,a}$, a set of weight vectors $W = \{\mu^1, \dots, \mu^{N_\mu}\}$, the population size N , the remaining number of generations G_{1-p} , and an initial population P with N solutions.

Ensure: A approximation set P_F of the region R .

```

1: for  $i = 1, \dots, k$  do
2:   Calculate  $q^{min,i} = \min_{\mathbf{x} \in \mathbf{D}^{r,a}} f_i(\mathbf{x})$ .
3: end for
4: Define  $\bar{\mathbf{q}}^a = (q^{min,1}, q^{min,2}, \dots, q^{min,k})$ 
5: Set  $h = 0$  and  $P^0 = P$ .
6: while  $h \leq G_{1-p}$  do
7:   Set  $Z^h = \emptyset$  and  $n = 0$ .
8:   Apply the selection, crossover and mutation operators to  $P^h$ 
   and create a population  $Q^h$  with  $N$  new individuals. Set  $Z^h = P^h \cup Q^h$ .
9:   For every  $\mathbf{x} \in Z^h$ , calculate  $s(\bar{\mathbf{q}}^a, \mathbf{f}(\mathbf{x}), \bar{\mu}^j)$  with  $j = 1, \dots, N_\mu$ .
10:  while  $Z^h \neq \emptyset$  do
11:    Set  $n = n + 1$  and initialize  $F_n^h = \emptyset$ .
12:    for  $j = 1, \dots, N_\mu$  do
13:      Find  $\mathbf{x}_j^n \in Z^h$  such that  $s(\bar{\mathbf{q}}^a, \mathbf{f}(\mathbf{x}_j^n), \mu^j) = \min_{\mathbf{x} \in Z^h} s(\bar{\mathbf{q}}^a, \mathbf{f}(\mathbf{x}), \mu^j)$ .
14:      Update  $F_n^h = F_n^h \cup \{\mathbf{x}_j^n\}$ .
15:      Remove  $\mathbf{x}_j^n$  from  $Z^h$ .
16:    end for
17:  end while
18:  Set  $P^{h+1} = \emptyset$  and  $m = 1$ .
19:  while  $\#(P^{h+1} \cup F_m^h) \leq N$  do
20:    Update  $P^{h+1} = P^{h+1} \cup F_m^h$ .
21:    Set  $m = m + 1$ .
22:  end while
23:  Until having  $N$  solutions in  $P^{h+1}$ , include into  $P^{h+1}$ 
  the solutions corresponding to the lowest values in  $\{s(\bar{\mathbf{q}}^a, \mathbf{f}(\mathbf{x}), \mu^j)\}_{j=1, \dots, N_\mu, \mathbf{x} \in F_m^h}$ .
24:  Update  $h = h + 1$ .
25:  if  $h = 0.1 \cdot G_{1-p}$  then
26:    If desired, perform an adaptation of all the weight vectors
    in  $W$  following [16], [17] and update  $W$  accordingly.
27:  end if
28: end while
29: Set  $P_F = F_1^{h-1}$ .

```

problems, respectively. For the WFG problems, we set the position- and distance-related parameters to $k - 1$ and 10, respectively.

Our experiment has been implemented in Java using the jMetal framework [19]. For each problem, we have randomly generated six pairs of reservation and aspiration points that have been used to test each algorithm. Note that using such number of reservation and aspiration points ensures that the three scenarios described in Section II are considered in our computational study. Let us remark that, before running the algorithms, it is not possible to know if the generated points are achievable or unachievable. Besides, to have statistically significant results, we have executed 10 independent runs for each algorithm, each test problem and each pair of aspiration and reservation points. Thus, a total of 60 independent runs have been executed for each problem and each algorithm.

TABLE 1. Test problems and their properties.

Objectives (k)	Problem	Properties
2	ZDT1	Convex
	ZDT2	Concave
	ZDT3	Convex, discontinuous
	ZDT4	Convex, multi-modal
	ZDT6	Concave, multi-modal, biased
	UF1	Convex, complex PS
	UF2	Convex, complex PS
	UF3	Convex, complex PS
	UF4	Concave, complex PS
	UF5	Linear, discrete, complex PS
3	UF6	Linear, discontinuous, complex PS
	UF7	Linear, complex PS
	UF8	Concave, complex PS
	UF9	Linear, discontinuous, complex PS
2, 3, 5	UF10	Concave, complex PS
	DTLZ1	Linear, multi-modal, simplex-like PF
	DTLZ2	Concave
	DTLZ3	Concave, multi-modal
	DTLZ4	Concave, biased
	DTLZ7	Mixed, discontinuous, multi-modal
	WFG1	Mixed, biased
	WFG2	Convex, discontinuous, non-separable
	WFG3	Linear, degenerate, non-separable
	WFG4	Concave, multi-modal
	WFG5	Concave, deceptive
	WFG6	Concave, non-separable
	WFG7	Concave, biased
	WFG8	Concave, non-separable, biased
	WFG9	Concave, non-separable, deceptive, biased

For the two-, three-, and five-objective problems, respectively, the population sizes used are $N = 50, 91$ and 210 , and the maximum number of generations considered (stopping criterion) is $G_T = 400, 600$ and 800 .

In ERAL, Adapt-ERAL, AdaW-ERAL and cNSGA-III, we use the SBX crossover operator [4] (distribution index $\eta_c = 30$ and probability $P_c = 0.9$), and the polynomial mutation operator [4] (distribution index $\eta_m = 20$ and probability $P_m = 1/n$, where n is the number of variables). In the three ERAL variants, the value p has been set to 0.6 , which means that $G_p = 240, 360$ and 480 for the two-, three-, and five-objective problems, respectively. In cMOEA/D-DE, the neighborhood size used is $0.1 \cdot N$ and the probability of choosing the mate sub-problem from the neighborhood is 0.9 . The crossover ratio and the scale factor are both set to 0.5 for the DE operator. For a fair comparison, all algorithms initially use the same set of weight vectors.

The algorithms' performances have been evaluated with the hypervolume (HV) metric, which is obtained using the WFG calculation method suggested in [26] for all test problems. For the HV calculation, the reference point needed has been set depending on the achievability of the aspiration and reservation points considered in each case, so that it is suited as best possible to the ROI to be approximated. Specifically, the nadir point, the aspiration point and the reservation point have been used as the reference point required, respectively, in scenarios 1, 2 and 3 described in Section II.

Note that the HV values obtained by each algorithm have been normalized to have comparable values. For each problem and for each pair of aspiration and reservation points

(each region of interest), the HV value of each algorithm has been divided by the sum of the HV values of the five algorithms considered. Then, we have calculated the mean of the normalized HV values obtained in all the runs for each problem. Hereinafter, for simplicity, when we talk about the HV mean value, we refer to the mean of the HV values normalized in this way.

In addition, we apply a Wilcoxon rank-sum test [28] to check if the HV mean values achieved by ERAL, Adapt-ERAL and AdaW-ERAL are significantly different from the ones of the other algorithms. For each problem, the null hypothesis is that the distribution of their HV mean values in the 60 runs differ by a value α , assuming that the difference is significant if the p -value obtained is lower than $\alpha = 0.05$. We use the `wilcox.test` function from the R software.²

B. RESULTS

Tables 2, 3 and 4 show the HV mean values obtained in the 60 independent runs, for the two-, three- and five-objective problems, respectively. For each problem, we emphasize the algorithm with the best HV value in dark gray color, and the one with the second best value in light gray color. Besides, in Table 5, we indicate the numbers of problems for which the HV mean values of ERAL, Adapt-ERAL or AdaW-ERAL are significantly better than (\blacktriangle), equal to (\odot), or worse than (∇) the HV mean value of the other algorithms, according to the Wilcoxon test with a significance level $\alpha = 0.05$.

In relation to the two-objective problems (see Table 2), our results indicate that the three variants of ERAL perform better than cMOEA/D-DE and cNSGA-III in all problems. Observe that ERAL reaches the best HV mean value in 15 out of the 26 problems (hereafter indicated as 15/26), Adapt-ERAL is the best in 1/26 problems, AdaW-ERAL achieves the best results in 10/26 cases, while cMOEA/D-DE and cNSGA-III do not win in any problem. Moreover, the Wilcoxon test (see Table 5) reveals that, in the problems with two objectives, ERAL shows a statistically better performance in 24/26 cases compared to cNSGA-III, and it produces significantly better results in 16/26 cases with respect to cMOEA/D-DE. Similar results are found by the two ERAL variants. Indeed, Adapt-ERAL is significantly better in 25/26 cases compared to both cNSGA-III and cMOEA/D-DE, and AdaW-ERAL obtains significantly better results in 24/26 and 25/26 problems when compared to cNSGA-III and cMOEA/D-DE, respectively. The ROI approximation of the ZDT4 problem is shown in Figure 3. This representation corresponds to the ROI obtained for each algorithm in ten independent runs for a fixed level of preferences (an aspiration point and reservation point).

Concerning the problems with three objectives (see Table 3), in general, ERAL, Adapt-ERAL and AdaW-ERAL also outperform the other two algorithms. Specifically, each one wins in 6/16, 1/16 and 4/16 problems, respectively.

²<https://stat.ethz.ch/R-manual/R-devel/library/stats/html/wilcox.test.html>

TABLE 2. Normalized HV mean values in 60 independent runs for the two-objective problems.

	ERAL	Adapt-ERAL	AdaW-ERAL	cNSGA-III	cMOEA/D-DE
ZDT1	0.312165	0.308950	0.312097	0.033540	0.033247
ZDT2	0.315462	0.301031	0.315303	0.035532	0.032672
ZDT3	0.264606	0.263677	0.263629	0.100879	0.107209
ZDT4	0.262439	0.251519	0.261923	0.135729	0.088389
ZDT6	0.321833	0.282529	0.321832	0.040547	0.033258
DTLZ1	0.268410	0.248618	0.251158	0.184671	0.047143
DTLZ2	0.308894	0.308782	0.308891	0.041209	0.032225
DTLZ3	0.281903	0.256901	0.278311	0.165407	0.017478
DTLZ4	0.279099	0.277408	0.307951	0.068421	0.067121
DTLZ7	0.264279	0.264270	0.264272	0.110798	0.096381
UF1	0.256288	0.247120	0.239971	0.128314	0.128308
UF2	0.273412	0.268791	0.273006	0.117618	0.067173
UF3	0.176749	0.155086	0.203300	0.167302	0.164229
UF4	0.231336	0.250745	0.284974	0.102634	0.080311
UF5	0.188029	0.182681	0.224625	0.157101	0.130896
UF6	0.185287	0.223413	0.175621	0.155760	0.126585
UF7	0.343821	0.241429	0.277026	0.067704	0.053353
WFG1	0.186270	0.189801	0.226501	0.030068	0.100694
WFG2	0.284954	0.282962	0.284759	0.073809	0.073516
WFG3	0.245462	0.243980	0.245453	0.133846	0.131259
WFG4	0.312273	0.310428	0.312500	0.033714	0.031085
WFG5	0.246825	0.246834	0.246820	0.136243	0.123278
WFG6	0.265481	0.263599	0.273305	0.099507	0.098108
WFG7	0.294587	0.283030	0.295603	0.068986	0.057794
WFG8	0.179413	0.177196	0.180587	0.065579	0.063892
WFG9	0.309856	0.314398	0.323509	0.034465	0.017772

TABLE 3. Normalized HV mean values in 60 independent runs for the three-objective problems.

	ERAL	Adapt-ERAL	AdaW-ERAL	cNSGA-III	cMOEA/D-DE
DTLZ1	0.295427	0.302241	0.295516	0.075096	0.031720
DTLZ2	0.313179	0.311728	0.310213	0.035496	0.029384
DTLZ3	0.290967	0.287716	0.289795	0.080680	0.050843
DTLZ4	0.238018	0.230525	0.234331	0.077804	0.052655
DTLZ7	0.288185	0.280372	0.303615	0.064925	0.062902
UF8	0.079415	0.063133	0.038694	0.050908	0.084516
UF9	0.161201	0.163661	0.188460	0.145893	0.157452
WFG1	0.122455	0.107503	0.129868	0.051597	0.088576
WFG2	0.077273	0.043045	0.034822	0.034413	0.027113
WFG3	0.089667	0.089745	0.089787	0.030848	0.033286
WFG4	0.037027	0.036371	0.037661	0.168835	0.053439
WFG5	0.115685	0.110300	0.107348	0.157546	0.175787
WFG6	0.204346	0.198139	0.201911	0.037557	0.024713
WFG7	0.147657	0.146507	0.147511	0.040079	0.018245
WFG8	0.170475	0.159384	0.170141	0.311600	0.188400
WFG9	0.133397	0.144890	0.144585	0.227356	0.166439

TABLE 4. Normalized HV mean values in 60 independent runs for the five-objective problems.

	ERAL	Adapt-ERAL	AdaW-ERAL	cNSGA-III	cMOEA/D-DE
DTLZ1	0.233377	0.244917	0.232367	0.194458	0.094880
DTLZ2	0.307107	0.318787	0.305108	0.044157	0.024840
DTLZ3	0.265165	0.256079	0.261278	0.131155	0.086323
DTLZ4	0.293841	0.288255	0.295033	0.083468	0.039402
DTLZ7	0.284977	0.280613	0.287338	0.095719	0.051353
WFG1	0.292130	0.289321	0.266132	0.048856	0.103561
WFG2	0.238526	0.203094	0.212721	0.134226	0.128100
WFG3	0.296543	0.300117	0.297351	0.082385	0.023604
WFG4	0.287069	0.351871	0.284760	0.054091	0.022209
WFG5	0.230014	0.233026	0.229931	0.218068	0.088961
WFG6	0.315492	0.308980	0.313490	0.043847	0.018192
WFG7	0.206172	0.235362	0.207187	0.261683	0.089595
WFG8	0.173586	0.164849	0.176948	0.193683	0.124267
WFG9	0.307159	0.311243	0.322319	0.039245	0.020034

In turn, cNSGA-III achieves the best HV mean in 3/16 problems, and cMOEA/D-DE wins in 2/16 cases. Regarding the significance of the results for the 3-objective problems

(see Table 5), both ERAL and Adapt-ERAL obtain significantly better results in 9/16 problems compared to cNSGA-III, and in 8/16 problems when compared with

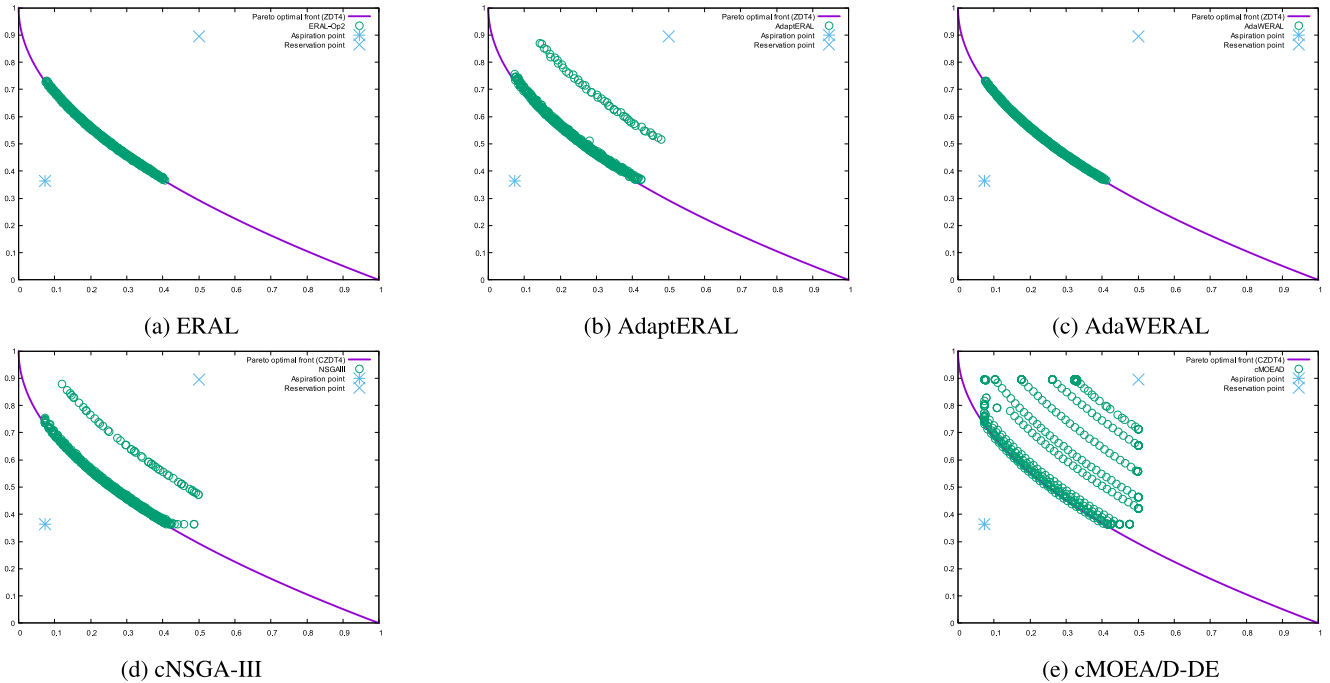


FIGURE 3. ROI approximation of ZDT4 problem.

TABLE 5. Problems in which the HV mean values of ERAL, Adapt-ERAL and AdaW-ERAL are better than (▲), equal to (◎) and worst than (▽) the ones of cNSGA-III and cMOEA/D-DE.

	cNSGA-III			cMOEA/D-DE			
2D	ERAL	▲ 24	◎ 2	▽ 0	▲ 25	◎ 1	▽ 0
	Adapt-ERAL	▲ 25	◎ 1	▽ 0	▲ 25	◎ 1	▽ 0
	AdaW-ERAL	▲ 24	◎ 2	▽ 0	▲ 25	◎ 1	▽ 0
3D	ERAL	▲ 9	◎ 4	▽ 3	▲ 8	◎ 8	▽ 0
	Adapt-ERAL	▲ 9	◎ 5	▽ 2	▲ 8	◎ 8	▽ 0
	AdaW-ERAL	▲ 9	◎ 5	▽ 2	▲ 9	◎ 7	▽ 0
5D	ERAL	▲ 11	◎ 3	▽ 0	▲ 14	◎ 0	▽ 0
	Adapt-ERAL	▲ 11	◎ 3	▽ 0	▲ 14	◎ 0	▽ 0
	AdaW-ERAL	▲ 11	◎ 3	▽ 0	▲ 14	◎ 0	▽ 0

cMOEA/D-DE, while AdaW-ERAL is significantly better than both cNSGA-III and cMOEA/D-DE in 9/16 problems.

Finally, let us analyze the results for the five-objective problems (see Table 4). Overall, ERAL and its variants achieve the best results once more (they win in 12/14 problems). We can see that ERAL is the best one in 4/14 problems, Adapt-ERAL in 5/14 problems and AdaW-ERAL in 3/14 problems. However, cNSGA-III obtains the best results in just 2/14 cases, while cMOEA/D-DE does not win in any problem. Moreover, in relation to the Wilcoxon test (see Table 5), the HV mean values of ERAL, Adapt-ERAL and AdaW-ERAL are statistically better than those of cNSGA-III in 11/14 problems, and better than the cMOEA/D-DE's in 14/14 problems.

With this computational study, our proposal's performance has been demonstrated in a wide range of benchmark problems. Overall, ERAL and its two variants have outperformed the other two algorithms in most of the problems.

Moreover, although we have not analyzed in detail which algorithm has achieved the second best HV mean value, it can be seen at a glance that, whenever they do not obtain the best results, ERAL, Adapt-ERAL and AdaW-ERAL are ranked as the second best ones in most cases of the two-, three- and five-objective problems. This indicates that our proposal seems to achieve very promising results, especially as the number of objective functions increases.

Nevertheless, we cannot clearly state that any of the variants of the proposed algorithm is more adequate than the others for the problems considered, given that the results they reached were very similar (at least for the HV metric). In the problems with two objectives, we observe that the fact of internally adapting the weight vectors while ERAL is converging may imply a significant improvement in the approximation set obtained, since ERAL stood out from Adapt-ERAL and AdaW-ERAL in these cases. However, in the comparison using the problems with three and five objectives, the performances of the three algorithms were very similar. Anyway, since the purpose of performing an adaptation of the weight vectors is to orient the search for new non-dominated solutions according to the PF's features, Adapt-ERAL and AdaW-ERAL may be more suitable when solving problems with complicated PFs.

IV. CONCLUSION

In this paper, a new preference-based EMO algorithm has been proposed that manages preference information in the form of aspiration and reservation points. The new algorithm, named ERAL, is designed to approximate the ROI of the

PF formed by the Pareto optimal solutions whose objective function values are within the desired aspiration and reservation levels. To achieve this, a number of generations are first executed using the reservation point and an initial set of weight vectors. Next, according to the relation of dominance between the solutions generated so far and the two points given by the DM, the algorithm analyzes the aspiration and reservation points to see what type they are (achievable or unachievable). Finally, the other generations are performed following different procedures depending on their achievability, which determines the ROI to be approximated.

Our proposal also incorporates the possibility of adapting the weight vectors used internally, with the aim of adjusting the convergence towards the desired ROI as much as possible, taking into account the PF's characteristics. Two procedures have been incorporated in ERAL for the adaptation of the weight vectors based on [17] and [16].

For the two-, three-, and five-objective benchmark problems considered in the computational experiments, we concluded that ERAL and its two variants, Adapt-ERAL and AdaW-ERAL (which perform the adaptation of the weights following [16], [17], respectively), have generated very promising results. They have shown a better performance in comparison to other well-known EMO algorithms, according to the HV metric.

As future research, we plan to study how the algorithm performs in higher dimension benchmark problems. In addition, we want to investigate the potential of ERAL to solve real-world problems using preferences given by real DMs, especially in areas such as engineering or economics.

REFERENCES

- [1] J. Branke, K. Deb, K. Miettinen, and R. Slowinski, Eds., *Multiobjective Optimization. Interactive and Evolutionary Approaches*. Berlin, Germany: Springer, 2008.
- [2] C. A. C. Coello, G. B. Lamont, and D. A. V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed. New York, NY, USA: Springer, 2007.
- [3] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.
- [4] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Syst.*, vol. 9, no. 2, pp. 115–148, 1995.
- [5] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach—Part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Apr. 2013.
- [6] K. Deb, J. Sundar, B. Ubay, and S. Chaudhuri, "Reference point based multi-objective optimization using evolutionary algorithm," *Int. J. Comput. Intell. Res.*, vol. 2, no. 6, pp. 273–286, 2006.
- [7] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proc. Congr. Evol. Comput. (CEC)*, vol. 1, May 2002, pp. 825–830.
- [8] M. Gong, F. Liu, W. Zhang, L. Jiao, and Q. Zhang, "Interactive MOEA/D for multi-objective decision making," in *Proc. 13th Annu. Conf. Genetic Evol. Comput. (GECCO)*, Jul. 2011, pp. 721–728.
- [9] S. Gonzalez-Gallardo, R. Saborido, A. B. Ruiz, and M. Luque, "An improvement study of the decomposition-based algorithm Global WASF-GA for evolutionary multiobjective optimization," in *Advances in Artificial Intelligence (Lecture Notes in Computer Science)*, vol. 11160, F. Herrera, S. Damas, R. Montes, S. Alonso, O. Cordon, A. Gonzalez, and A. Troncoso, Eds. Cham, Switzerland: Springer, 2018, pp. 219–229.
- [10] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.
- [11] H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes," *IEEE Trans. Evol. Comput.*, vol. 21, no. 2, pp. 169–190, Apr. 2017.
- [12] M. A. Jan and Q. Zhang, "MOEA/D for constrained multiobjective optimization: Some preliminary experimental results," in *Proc. UK Workshop Comput. Intell. (UKCI)*, Sep. 2010, pp. 721–726.
- [13] H. Li and D. Landa-Silva, "An adaptive evolutionary multi-objective approach based on simulated annealing," *Evol. Comput.*, vol. 19, no. 4, pp. 561–595, 2011.
- [14] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.
- [15] M. Li, S. Yang, and X. Liu, "Pareto or non-Pareto: Bi-criterion evolution in multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 5, pp. 645–665, Oct. 2016.
- [16] M. Li and X. Yao, "What weights work for you? Adapting weights for any Pareto front shape in decomposition-based evolutionary multiobjective optimisation," *Evol. Comput.*, vol. 28, no. 2, pp. 227–253, Jun. 2020.
- [17] M. Luque, S. Gonzalez-Gallardo, R. Saborido, and A. B. Ruiz, "Adaptive global WASF-GA to handle many-objective optimization problems," *Swarm Evol. Comput.*, vol. 54, May 2020, Art. no. 100644.
- [18] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston, MA, USA: Kluwer, 1999.
- [19] A. J. Nebro, J. J. Durillo, and M. Vergne, "Redesigning the jMetal multi-objective optimization framework," in *Proc. Companion Publication Annu. Conf. Genetic Evol. Comput.*, Jul. 2015, pp. 1093–1100.
- [20] Y. Qi, X. L. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu, "MOEA/D with adaptive weight adjustment," *Evol. Comput.*, vol. 22, no. 2, pp. 231–264, 2014.
- [21] A. B. Ruiz, R. Saborido, and M. Luque, "A preference-based evolutionary algorithm for multiobjective optimization: The weighting achievement scalarizing function genetic algorithm," *J. Global Optim.*, vol. 62, no. 1, pp. 101–129, 2015.
- [22] R. Saborido, A. B. Ruiz, and M. Luque, "Global WASF-GA: An evolutionary algorithm in multiobjective optimization to approximate the whole Pareto optimal front," *Evol. Comput.*, vol. 25, no. 2, pp. 309–349, Jun. 2017.
- [23] R. Saborido, A. B. Ruiz, M. Luque, and K. Miettinen, "IRA-EMO: Interactive method using reservation and aspiration levels for evolutionary multiobjective optimization," in *Evolutionary Multi-Criterion Optimization (Lecture Notes in Computer Science)*, vol. 11411, K. Deb, E. Goodman, C. A. C. Coello, K. Klamroth, K. Miettinen, S. Mostaghim, P. Reed, Eds. Cham, Switzerland: Springer, 2019, pp. 618–630.
- [24] S. Jiang, Z. Cai, J. Zhang, and O. Yew-Soon, "Multiobjective optimization by decomposition with Pareto-adaptive weight vectors," in *Proc. 7th Int. Conf. Natural Comput.*, vol. 3, Jul. 2011, pp. 1260–1264.
- [25] R. Wang, R. C. Purshouse, and P. J. Fleming, "Whatever works best for you—A new method for a priori and progressive multi-objective optimisation," in *Evolutionary Multi-Criterion Optimization (Lecture Notes in Computer Science)*, vol. 7811, R. C. Purshouse, P. J. Fleming, C. M. Fonseca, S. Greco, and J. Shaw, Eds. Berlin, Germany: Springer, 2013, pp. 337–351.
- [26] L. While, L. Bradstreet, and L. Barone, "A fast way of calculating exact hypervolumes," *IEEE Trans. Evol. Comput.*, vol. 16, no. 1, pp. 86–95, Feb. 2012.
- [27] A. P. Wierzbicki, "The use of reference objectives in multiobjective optimization," in *Multiple Criteria Decision Making, Theory and Applications*, G. Fandel and T. Gal, Eds. Berlin, Germany: Springer, 1980, pp. 468–486.
- [28] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, 1945.
- [29] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
- [30] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," Univ. Essex and Nanyang Technol. Univ., Singapore, Tech. Rep. CES-487, 2008.
- [31] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.



SANDRA GONZÁLEZ-GALLARDO received the degree in mathematics, the master's degree in economic and business analysis, and the Ph.D. degree from the University of Málaga, in 2016, 2017, and 2020, respectively.

She has been in a predoctoral research stay at ISCAC, Portugal. She is currently a Researcher with the Department of Applied Economics (Mathematics), University of Málaga, Spain. Her research interests include multiobjective programming techniques, particularly evolutionary algorithms and interval programming. The multiobjective techniques are applied to economics of education, specifically in the students' well-being and satisfaction. Her first publications have been in swarm and evolutionary computation and lecture notes in artificial intelligence. Indeed, her works have been showed in international conferences, such as MCDM, EURO, MESS, and among others.



RUBÉN SABORIDO received the B.S. degree in computer engineering and the M.S. degree in software engineering and artificial intelligence from the University of Malaga, Spain, and the Ph.D. degree in computer engineering from Polytechnique Montréal, Canada, in 2017.

He worked as a Research Assistant with the University of Malaga, for a period of three years. He is currently a Researcher with the Networking and Emerging Optimization Group, Department of Computer Science, University of Malaga. His thesis was nominated for best thesis award. He held a Postdoctoral Fellowship at Concordia University, Canada, in 2018, where he worked on search-based software engineering for the Internet of Things (IoT). He has recently been awarded a Juan de la Cierva grant, funded by the Spanish State Research Agency, and ranked in the top five of the candidates in the information and communications technologies area. His research interest includes search-based software engineering. He is also interested in the use of metaheuristics to solve multidisciplinary real-world problems of interest for our society and computer science. He has published several articles in ISI indexed journals, such as *EMSE*, *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, *Evolutionary Computation*, and *Swarm and Evolutionary Computation*, and conference papers in *IEEE ICPC*, *MCDM*, *IEEE SANER*, and *ACM ESEC/FSE*.

Dr. Saborido has co-organized the International Conference on Multiple Criteria Decision Making, in 2013. He is on the Organizing Committee of the 1st, 2nd, and 3rd International Workshop on Software Engineering Research & Practices for the Internet of Things (SERP4IoT), co-located with ICSE 2019, ICSE 2020, and ICSE 2021, respectively. Since 2016, he has been on the Program Committee of the Real World Applications (RWA) Track of the Genetic and Evolutionary Computation Conference (GECCO).



ANA B. RUIZ received the degree and the Ph.D. degree in mathematics from the University of Málaga, Spain, in 2006 and 2012, respectively. She is currently a Senior Lecturer with the Department of Applied Economics (Mathematics), University of Málaga. Her research interests include multi-objective optimization and multiple criteria decision-making approaches, such as evolutionary algorithms, interactive methods, and reference point-based techniques, as well as applications to real problems of different fields, such as education, economics, sustainability or engineering.



MARIANO LUQUE received the degree in mathematics with an academic average score of 9 out of 10, in 1995, the bachelor's degree in mathematics, in 1997, and the Ph.D. degree in quantitative methods in economics, in 2000.

Since 2015, he has been a Full Professor of quantitative methods in economics with the Department of Applied Economics (Mathematics), University of Málaga, Spain. He is also working on applications in the fields of education economics, workers' satisfaction, electricity generation, and sustainability issues. He has published about 50 articles in JCR journals, such as *Evolutionary Computation*, *Swarm and Evolutionary Computation*, *European Journal of Operational Research*, *Applied Soft Computing*, *Computers and Operations Research*, *Omega*, and *Management Science*. He has given several talks at different national and international universities such as The University of Manchester, University of Jyväskylä, and University of Valencia. He has participated and led numerous national and regional research projects, including the current research group Optimization and Econometrics (OPTECO), University of Malaga. His research interests include multiobjective optimization techniques, especially interactive methods, reference point-based methods, interval programming, and evolutionary algorithms.

...