On the utilization of pair-potential energy functions in multi-objective optimization[★]

Jesús Guillermo Falcón-Cardona[a,*], Edgar Covantes Osuna[a], Carlos A. Coello Coello[b], Hisao Ishibuchi[c]

[a] *Tecnologico de Monterrey, School of Engineering and Sciences, Ave. Eugenio Garza Sada 2501, Monterrey, N.L., México, 64849*
[b] *CINVESTAV-IPN, Department of Computer Science, Av. IPN 2508, Col. San Pedro Zacatenco, Mexico City, México, 07360*
Basque Center for Applied Mathematics (BCAM) & Ikerbasque, Spain
[c] *Southern University of Science and Technology, Department of Computer Science and Engineering, 1088 Xueyuan Avenue, Shenzhen, People's Republic of China, 518055*

Abstract

In evolutionary multi-objective optimization (EMO), the pair-potential energy functions (PPFs) have been used to construct diversity-preserving mechanisms to improve Pareto front approximations. Despite PPFs have shown promising results when dealing with different Pareto front geometries, there are still some open research questions to improve the way we employ them. In this paper, we answer three important questions: 1) what is the effect of a crucial parameter of some PPFs?, 2) how do we set the optimal parameter value?, and 3) what is the best PPF in EMO? To solve these questions, we designed a brand-new fast algorithm to generate an approximate solution to a PPF-based subset selection problem and, then, we conducted a comprehensive parametrical study to predict the optimal parameter values using a deep neural network. To show the effectiveness of the PPF-based diversity-preserving mechanisms, we selected two application cases: the generation of reference point sets of benchmark problems (DTLZ, WFG, IDTLZ, IWFG, IMOP, and Viennet) with different Pareto front shapes, and the definition of a PPF-based archive that can be coupled to any multi-objective evolutionary algorithm to construct well-diversified Pareto front approximations. Using several diversity indicators, it is shown that the utilization of PPF-based mechanisms lead to good Pareto front approximations regardless of the Pareto front shape.

Keywords: Pair-Potential Energy Functions, Multi-Objective Optimization, Reference Sets, Diversity-oriented Optimization

1. Introduction

In physics, a pair-potential energy function (PPF) $\mathcal{K} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ measures the interaction between two objects located in the Euclidean space $\mathbb{R}^m$. A well-known PPF is the Coulomb's law that measures the potential energy between two stationary electrically charged particles located at positions $\vec{u}, \vec{v} \in \mathbb{R}^m$. In addition to its utilization in physics, PPFs have been used in mathematics to tackle the problem of distributing N points $\mathcal{A} = \{\vec{a}_1, \ldots, \vec{a}_N\}, \vec{a}_i \in \mathbb{R}^m$ on a manifold [1]. The underlying idea is to find a configuration of the N candidate points all over the manifold to minimize the total $\mathcal{K}$-energy ($U^{\mathcal{K}}$) that sums all the pair-potential interactions using $\mathcal{K}$. Reaching the optimal N-point configuration (or state of minimal energy) leads to a proper discretization of the manifold [1, 2].

Due to the good properties of PPFs to discretize manifolds, they have recently been used in evolutionary multi-objective optimization (EMO) to discretize a Pareto front, aiming to improve the spread and uniformity[1]. It is worth noting that a Pareto front is a manifold of dimension (at most) $m - 1$, where $m \geq 2$ represents the number of objective functions. In EMO, the discrete Riesz α-energy function ($\mathcal{K}^{\text{RSE}}$) [2] has been the most popularly used PPF. $\mathcal{K}^{\text{RSE}}$ has been used to design a density estimator [3] and an archiving strategy [4], to measure the diversity of an approximation to the Pareto front, i.e., as a quality indicator (QI[2]) [4], to define a combined QI [8], to manage a set of weight vectors [9–11], and to construct a reference point set [12, 13]. In addition to $\mathcal{K}^{\text{RSE}}$, other PPFs have been empirically analyzed in the EMO field [14]: Gaussian α-energy ($\mathcal{K}^{\text{GAE}}$), Pöschl-Teller potential ($\mathcal{K}^{\text{PTP}}$), Modified Pöschl-Teller potential $\mathcal{K}^{\text{MPT}}$), Kratzer potential ($\mathcal{K}^{\text{KRA}}$), and Coulomb's law ($\mathcal{K}^{\text{COU}}$). Similarly to $\mathcal{K}^{\text{RSE}}$, these PPFs have shown promising results as diversity-preserving mechanisms regardless of the Pareto front geometry.

Despite the promising results on the utilization of PPFs in EMO, there are still some open research questions. First, most diversity-preserving mechanisms involve solving an optimal subset selection problem which is computationally expensive [15].

[★] Carlos A. Coello Coello gratefully acknowledges support from CONACyT grant no. 2016-01-1920 (Investigación en Fronteras de la Ciencia 2016). He was also partially supported by the Basque Government through the BERC 2022-2025 program and by Spanish Ministry of Economy and Competitiveness MINECO: BCAM Severo Ochoa excellence accreditation SEV-2017-0718.

[*] Corresponding author

Email addresses: jfalcon@tec.mx (Jesús Guillermo Falcón-Cardona), edgar.covantes@tec.mx (Edgar Covantes Osuna), ccoello@cs.cinvestav.mx (Carlos A. Coello Coello), hisao@sustech.edu.cn (Hisao Ishibuchi)

[1] From this point onwards, we will call diversity when both properties, spread and uniformity, are present.

[2] The fundamental difference between PPFs and QIs such as the hypervolume indicator (HV) [5], $R2$ [6], or the degree of approximation QI [7] is that these QIs reward convergence towards the Pareto front. In contrast, PPFs do not consider convergence.

PPF-based diversity-preserving mechanisms are not the exception. Therefore, it is mandatory to design efficient algorithms to approximate the solution to the PPF-based subset selection problem (PPFSS) [12]. Secondly, some PPFs such as $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$ are parameter-dependent. It has been shown that different parameter values generate a different solution set to the PPFSS [12, 14]. However, we do not exactly known how to set the parameter values to obtain well-diversified Pareto front approximations for every PPF and test instance. Currently, we only have some fixed-value recommendations but they do not lead to good Pareto front approximations in all cases [14]. Last but not least, we should identify what is the best PPF, among all the available ones, that allows an MOEA to generate well-diversified Pareto front approximations. Addressing these open questions is primordial to correctly integrate PPFs into MOEAs to construct high-quality discretizations.

In consequence, in this paper, we focus on studying six PPFs (namely, $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, $\mathcal{K}^{\text{KRA}}$, and $\mathcal{K}^{\text{COU}}$) to exploit their advantages to generate well-diversified Pareto front approximations regardless of the geometry of the corresponding manifold. As a result, we aim to improve their utilization in EMO. It is worth noting that this paper presents a comprehensive extension of our previous works [12, 14]. The two applications of PPFs that we highlight are:

1. The design of a standard mechanism to generate well-diversified reference point sets for benchmark problems in EMO, varying the geometry, dimension, and cardinality. These reference point sets are critical for the calculation of some QIs [7, 16–18] and to construct selection mechanisms of multi-objective evolutionary algorithms (MOEAs) [19–21].
2. The design of a PPF-based diversity-preserving archive that can be coupled to any MOEA. The archive aims to take advantage of all the solutions generated throughout the evolutionary process although some of them are dropped by the main selection mechanism of the MOEA [13, 14]. The archive aims to maintain a well-diversified Pareto front approximation in comparison with the main population of an MOEA.

To show the properties of PPFs in EMO, we first provide a comprehensive parametrical study where we show that a single parameter (denoted as α) is critical to generating Pareto front approximations with high diversity. We approximate the optimal value of α using Newton's method and a genetic algorithm (GA). As a result of this study, we show that the optimal value of α depends on the geometry, dimensionality, and cardinality of the Pareto front approximation. In consequence, we propose a deep neural network that approximates the best value of α so that a brand-new fast algorithm (that we propose) generates a good approximate solution to a PPFSS. Using this fast algorithm and the neural network, we can utilize the PPFs on the construction of reference point sets related to well-known benchmark problems and increasing the diversity of solutions in MOEAs via an archiving strategy. In summary, the contributions of this paper are as follows:

1. A comprehensive study of a critical parameter associated with the selected PPFs.
2. A deep neural network model that predicts an appropriate value of α taking into account the geometry, dimensionality, and cardinality of a Pareto front approximation.
3. A fast greedy removal algorithm to approximately solve a PPFSS which is fed by the neural network to generate well-diversified discretizations.
4. A parameterless mechanism to generate reference point sets with high diversity properties.
5. An analysis that shows that a PPF-based archive can store well-diversified Pareto front approximations taking advantage of the solutions generated by an MOEA.

The remainder of this paper is organized as follows. Section 2 provides a brief introduction to some basic terms to make this paper self-contained. Section 3 briefly describes some state-of-the-art diversity-preserving mechanisms. Section 4 outlines our fast greedy removal algorithm to approximate the solution to PPFSS. Section 5 describes the analysis of parameters of PPFs to generate well-diversified Pareto front approximations using the fast greedy removal algorithm. In Section 6, we show the utilization of PPFs to construct reference point sets and an archiving strategy. Finally, our main conclusions and future work are highlighted in Section 7.

2. Background

In this section, we present some basic mathematical definitions for the understanding of this paper. First, we define an unconstrained multi-objective optimization problem. Then, we provide the mathematical definitions of the selected PPFs.

2.1. Multi-objective optimization

According to Coello Coello *et al.* [22], and without loss of generality, an unconstrained multi-objective optimization problem, for minimization, is defined as follows:

$$\min_{\vec{x}\in\Omega} f(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \ldots, f_m(\vec{x})) \tag{1}$$

where $\vec{x} = (x_1, \ldots, x_n)$ is a vector of n decision variables and $\Omega \subseteq \mathbb{R}^n$ is the decision space. $f(\vec{x})$ is a vector of $m \geq 2$, conflicting, objective functions where for $i = 1, 2, \ldots, m$, $f_i : \Omega \to \mathbb{R}$. Due to the conflict among the objective functions, there is no single solution that simultaneously minimizes all the objective functions. Hence, the goal is to find a set of optimal solutions that represent the best-possible trade-offs among the objective functions. In consequence, it is first necessary to introduce the Pareto dominance relation to establish a general notion of optimality, i.e., Pareto optimality.

Definition 1 (Pareto dominance relation). *Given $\vec{x}, \vec{y} \in \Omega$, we say that $\vec{x}$ Pareto-dominates $\vec{y}$ (denoted as $\vec{x} \prec \vec{y}$) if and only if $f_i(\vec{x}) \leq f_i(\vec{y})$ for $i = 1, 2, \ldots, m$ and there exists at least an index $j \in \{1, 2, \ldots, m\}$ such that $f_j(\vec{x}) < f_j(\vec{y})$.*

Based on the Pareto dominance relation, we can define the notion of Pareto-optimal solutions. Pareto optimality does not take into account any preferences, thus, it is a general notion of optimality and it has been commonly employed in multi-objective optimization [22, 23].

Definition 2 (Pareto optimality). *We say that $\vec{x}^*$ is a Pareto-optimal solution if there is no other vector of decision variables $\vec{x}$ such that $\vec{x} \prec \vec{x}^*$.*

The solution to a MOP is the set of all Pareto-optimal solutions. This set is also known as the Pareto set and its image in the objective space is called the *Pareto front*.

Definition 3 (Pareto set). *The Pareto set (P^*) is defined as follows:*

$$P^* = \{\vec{x}^* \mid \vec{x}^* \text{ is a Pareto-optimal solution}\}. \tag{2}$$

Definition 4 (Pareto front). *The Pareto front (PF^*) is the image of the Pareto set, i.e., $PF^* = \{f(\vec{x}^*) \mid \vec{x}^* \in P^*\}$.*

Since Pareto fronts can be continuous, it is not possible to have their complete representations. Hence, it is necessary to have a finite representation denoted as Pareto front approximation (or approximation set).

Definition 5 (Pareto front approximation). *A Pareto front approximation is a finite set $\mathcal{A} = \{f(\vec{x}_i) \mid \vec{x}_i \in \Omega, i = 1, \ldots, N\}$, consisting of N objective vectors, where neither $\vec{x}_i \prec \vec{x}_j$ nor $\vec{x}_j \prec \vec{x}_i$ for all i and j.*

2.2. Definition of PPFs

According to Borodachov *et al.* [1], the total potential $\mathcal{K}$-energy ($U^{\mathcal{K}}$) that measures the interaction between points in any N-point configuration (or system) is defined as follows:

$$U^{\mathcal{K}}(\mathcal{A}) = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} \mathcal{K}(\vec{a}_i, \vec{a}_j) \tag{3}$$

where $\mathcal{A} = \{\vec{a}_1, \ldots, \vec{a}_N\}$ is an N-point configuration and $\vec{a}_i \in \mathbb{R}^m, i = 1, \ldots, N$. Additionally, $\mathcal{K} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ is a PPF that models the interaction between two points. This interaction can be characterized as repulsive or attractive. A PPF is repulsive when as long as the distance between two points increases, the PPF value strictly decreases. On the other hand, if the PPF value strictly increases, the PPF is denoted as attractive.

In the specialized literature [1, 24], there are several PPFs, but we only focus on six repulsive ones that we empirically analyzed in our previous work [14]. A repulsive PPF penalizes two points close to each other, which helps to break up crowded regions. Moreover, a PPF rewards a pair of distant points, which encourages the diversification of $\mathcal{A}$. This behavior has been taken into accout to create diversity-preserving mechanisms in EMO [3, 12].

For all cases, let $\vec{u}$ and $\vec{v}$ be two m-dimensional vectors, $||\cdot||$ represents the Euclidean distance, and $\alpha, V_1, V_2 > 0$ are user-defined parameters.

Definition 6 (Riesz α-energy [1]). *Given a parameter α[3], this PPF is defined as follows:*

$$\mathcal{K}^{RSE}(\vec{u}, \vec{v}) = \frac{1}{\left\|\vec{u} - \vec{v}\right\|^{\alpha}}. \tag{4}$$

Definition 7 (Gaussian α-energy [1]).

$$\mathcal{K}^{GAE}(\vec{u}, \vec{v}) = e^{-\alpha\left\|\vec{u}-\vec{v}\right\|^2}. \tag{5}$$

Definition 8 (Pöschl-Teller potential [25, 26]).

$$\mathcal{K}^{PTP}(\vec{u}, \vec{v}) = \frac{V_1}{\sin^2(\alpha\left\|\vec{u} - \vec{v}\right\|)} + \frac{V_2}{\cos^2(\alpha\left\|\vec{u} - \vec{v}\right\|)}. \tag{6}$$

Definition 9 (Modified Pöschl-Teller potential [25, 26]).

$$\mathcal{K}^{MPT}(\vec{u}, \vec{v}) = \frac{V_1}{\cosh^2(\alpha\left\|\vec{u} - \vec{v}\right\|)}. \tag{7}$$

Definition 10 (Kratzer potential [26, 27]).

$$\mathcal{K}^{KRA}(\vec{u}, \vec{v}) = V_1 \left(\frac{\left\|\vec{u} - \vec{v}\right\| - 1/\alpha}{\left\|\vec{u} - \vec{v}\right\|} \right)^2 + V_2. \tag{8}$$

Definition 11 (Coulomb's law [28]). *The Coulomb's law is given by:*

$$\mathcal{K}^{COU}(\vec{u}, \vec{v}) = \frac{q_1 q_2}{4\pi\epsilon_0} \cdot \frac{1}{\left\|\vec{u} - \vec{v}\right\|^2}, \tag{9}$$

where we set $q_1 = \left\|\vec{u}\right\| = \sqrt{\sum_{i=1}^m u_i^2}$ and $q_2 = \left\|\vec{v}\right\| = \sqrt{\sum_{i=1}^m v_i^2}$, and $\frac{1}{4\pi\epsilon_0}$ is the Coulomb's constant, being $\epsilon_0 \approx 8.85418781762039 \times 10^{-12}[F/m]$ as the vacuum permittivity.

By replacing each of these PPFs in Equation 3, we define six $\mathcal{K}$-energy functions, denoted as $U^{\mathcal{K}^{\text{RSE}}}$, $U^{\mathcal{K}^{\text{GAE}}}$, $U^{\mathcal{K}^{\text{MPT}}}$, $U^{\mathcal{K}^{\text{PTP}}}$, $U^{\mathcal{K}^{\text{KRA}}}$, and $U^{\mathcal{K}^{\text{COU}}}$. Regardless of the PPF utilized, we are interested in searching for an optimal N-point configuration $\mathcal{A}^*$ over a finite representation $\mathcal{M}$ of an m-dimensional manifold, where $|\mathcal{M}| >> N$. To this aim, it is mandatory to solve the following subset selection problem where $\mathcal{M}$ is a candidate set:

Definition 12 (PPF-based subset selection problem (PPFSS)).

$$\mathcal{A}^* = \underset{\substack{\mathcal{A} \subset \mathcal{M} \\ |\mathcal{A}| = N}}{\arg\min}\, U^{\mathcal{K}}(\mathcal{A}), \tag{10}$$

provided that such a minimum exists.

[3] Although in the literature the parameter is usually denoted by s, for the sake of homogeneity, we interchange s by α.

3. Diversity-preserving mechanisms: a brief review

Even though convergence is the most critical aspect that an MOEA should first ensure, spread and uniformity also take a relevant place to produce good Pareto front approximations [29–34]. When an MOEA has mechanisms that allow reaching both properties, the resulting finite Pareto front approximation is more likely to better discretize the continuous multi-dimensional manifold associated with the Pareto front. A diversity-preserving mechanism is essential in *a posteriori* methods (e.g., MOEAs) to avoid the convergence of the entire population to a single point and to provide a wider range of possibilities to the decision maker [23].

Since the early days of MOEAs, diversity preservation has been a challenging task [22, 29, 30]. One of the first approaches to obtain well-diversified Pareto front approximations was *niching* [35]. Niching methods rely on distance calculation to select a few solutions within a specific region of the objective or decision space [36, 37]. Similar to niching, *crowding* defines a function that measures how crowded is a region in the objective space. Crowding is implemented as the density estimator of the Nondominated Sorting Genetic Algorithm II (NSGA-II) [38]. In a similar way to crowding, clustering techniques have been applied to increase the diversity of a population [39]. Although niching and crowding/clustering boosted the development of MOEAs, the resulting Pareto front approximations presented poor diversity in some cases. To overcome this issue, decomposition-based [40, 41], reference point-based [42, 43], and indicator-based MOEAs (IB-MOEAs) [44, 45] are able to generate Pareto front approximations with better diversity[4]. Regarding decomposition-based MOEAs, the aim is to search for the intersection points between the Pareto front and a set of weight vectors[5], using a scalarizing function which in turn defines a single-objective optimization problem (SOP). A well-known representative of this class is the MOEA based on Decomposition (MOEA/D) [46]. On the other hand, some MOEAs such as NSGA-III [20] use a reference point set to push the population towards the Pareto front. In this regard, a reference point set is composed of objective vectors that represent the desires of a decision maker [23] or, in general, it approximates the geometry of the Pareto front. In EMO, reference point sets have been employed in three main directions: 1) to decompose a MOP into multiple SOPs as in MOEA/D, 2) to guide the evolution of a population toward the Pareto front, and to calculate QIs [19]. A reference point set may be fixed (using a set of weight vectors) or it can change to better adapt itself to the shape of the Pareto front. Last but not least, IB-MOEAs exploit the preferences of their baseline QIs to generate Pareto front approximations with specific diversity properties. For example, the $\mathcal{S}$-Metric Selection Evolutionary Multi-Objective Algorithm (SMS-EMOA) [47] produces solutions emphasizing the Pareto front's knee due to the preferences of the hypervolume indicator (HV) [5].

Even though the above-mentioned classes of MOEAs have significantly improved the diversity of Pareto front approximations, they still have some issues. For instance, SMS-EMOA produces uniform distributions when the Pareto front is linear but in concave ones, it leaves some regions without solutions[6]. More recently, this problem has been tackled by designing multi-indicator-based MOEAs [4]. Regarding MOEA/D, NSGA-III, and some other MOEAs that use weight vectors, it has been pointed out that not all the weight vectors intersect the Pareto front [49], e.g., when dealing with disconnected or degenerate geometries. In consequence, MOEAs face a great challenge to generate well-diversified Pareto front approximations due to a large class of possible geometries of Pareto fronts.

4. Fast greedy removal algorithm

Most of the mechanisms in EMO involve solving a subset selection problem. However, finding an optimal solution is computationally challenging due to the large number of possible subsets. Hence, approximation algorithms using heuristics are a possible way to overcome this issue. In this section, we describe a fast greedy removal algorithm to approximate the solution of PPFSS (defined in Equation 12). This algorithm is the key component to utilize PPFs in EMO for selection purposes at a low computational cost in Section 6.

Given a set $\mathcal{M}$ of size $|\mathcal{M}| >> N$, there are $\binom{|\mathcal{M}|}{N}$ subsets $\mathcal{A}$ of size N. Hence, finding $\mathcal{A}^*$ for the PPFSS is computationally expensive. To overcome this issue, we propose a fast greedy removal algorithm (see Algorithm 1) to construct an approximation to $\mathcal{A}^*$. The core of this approximation algorithm is a greedy heuristic decision based on the concept of contribution to the overall $U^{\mathcal{K}}$ value. Given $\vec{a} \in \mathcal{A}$, its individual contribution C to $U^{\mathcal{K}}$ is defined as follows:

$$C(\vec{a}, \mathcal{A}) = U^{\mathcal{K}}(\mathcal{A}) - U^{\mathcal{K}}(\mathcal{A} \setminus \{\vec{a}\}). \tag{11}$$

The greedy decision relies on deleting at each iteration the worst-contributing point $\vec{a}_{\text{worst}} = \max_{\vec{a} \in \mathcal{A}} C(\vec{a}, \mathcal{A})$. The reason to delete the solution with the maximum contribution is simply because PPFSS minimizes the value of $U^{\mathcal{K}}$. The removal of the maximum contributing solution implies the deletion of a solution in a crowded region of the manifold as shown in Figure 1.

Algorithm 1 is an improved version of the one proposed in our previous work [12]. The goal is to produce an N-point subset $\mathcal{A}$ from an initial candidate solution set $\mathcal{M}$ that represents an m-dimensional manifold with $N << |\mathcal{M}|$. First, we set $\mathcal{A} = \mathcal{M}$ and a matrix $\boldsymbol{K} \in \mathbb{R}^{|\mathcal{A}| \times |\mathcal{A}|}$ is calculated where $\boldsymbol{K}_{ij} = \mathcal{K}(\vec{a}_i, \vec{a}_j), i \neq j$, using the given value of α. The elements in the diagonal are equal to zero since $\left\|\vec{a}_i - \vec{a}_i\right\| = 0$. Line 3 initializes a memoization structure $\vec{r} = (r_1, r_2, \ldots, r_{|\mathcal{A}|})$, where $r_i = \sum_{j=1}^{|\mathcal{A}|} \boldsymbol{K}_{ij}$. Thus, we can rewrite $U^{\mathcal{K}}(\mathcal{A}) = \sum_{j=1}^{|\mathcal{A}|} r_j$.

[4] It is worth noting that the main focus of these MOEAs is to increase the convergence ability when solving MOPs with four or more objective functions (i.e., the so-called many-objective optimization problems) but they also improve the diversity as a collateral effect.

[5] Given $\vec{w} \in \mathbb{R}^m$, we say that $\vec{w}$ is a weight vector if $\sum_{i=1}^{m} w_i = 1$ and $w_i \geq 0$ for all $i \in \{1, \ldots, m\}$.

[6] This can be controlled by the problem-dependent specification of the reference point required by the hypervolume indicator [48].

The while-loop in lines 4 to 10 describes the iterative process to eliminate the solution $\vec{a}_{\text{worst}}$ from $\mathcal{A}$. In Line 5, we obtain the index j_{worst} of the maximum value in $\vec{r}$ to identify $\vec{a}_{\text{worst}}$. The justification for doing this is as follows. Since PPFs internally employ the Euclidean distance, it follows that $\boldsymbol{K}_{ij} = \boldsymbol{K}_{ji}$ and we can write $U_i = U^{\mathcal{K}}(\mathcal{A} \setminus \{\vec{a}_i\}) = \sum_{j=1}^{|\mathcal{A}|} \left(r_j - \boldsymbol{K}_{ij}\right)$. In consequence, $C(\vec{a}_i, \mathcal{A}) = U^{\mathcal{K}}(\mathcal{A}) - U_i = \sum_{j=1}^{|\mathcal{A}|} r_j - \sum_{j=1}^{|\mathcal{A}|}(r_j - \boldsymbol{K}_{ij}) = r_i$. The next step is to update all the components of the memoization structure by substracting $\boldsymbol{K}_{ij_{\text{worst}}}$. Then, we delete the $j^{\text{th}}_{\text{worst}}$ column and row from $\boldsymbol{K}$ and $r_{j_{\text{worst}}}$ from $\vec{r}$. Finally, $\vec{a}_{j_{\text{worst}}}$ is dropped from $\mathcal{A}$. The process continues until $|\mathcal{A}| = N$.

Algorithm 1 Fast greedy removal algorithm

```
Input: Candidate solution set $\mathcal{M}$; subset size $N(<< |\mathcal{M}|)$; initial $\alpha$ value
Output: $N$-point set $\mathcal{A}$
 1: $\mathcal{A} = \mathcal{M}$
 2: Calculate matrix $\boldsymbol{K}$, using the given $\alpha$ value
 3: Calculate $r_i = \sum_{j=1}^{|\mathcal{A}|} \boldsymbol{K}_{ij}, \forall i = 1, \ldots, |\mathcal{A}|$
 4: while $|\mathcal{A}| > N$ do
 5:     $j_{\text{worst}} = \arg\max_{j=1,\ldots,|\mathcal{A}|} r_j$
 6:     for $i = 1$ to $|\mathcal{A}|$ do
 7:         $r_i = r_i - \boldsymbol{K}_{ij_{\text{worst}}}$
 8:     Delete the $j^{\text{th}}_{\text{worst}}$ column and row of $\boldsymbol{K}$
 9:     Delete the $j^{\text{th}}_{\text{worst}}$ component of $\vec{r}$
10:     $\mathcal{A} = \mathcal{A} \setminus \{\vec{a}_{j_{\text{worst}}}\}$
11: return $\mathcal{A}$
```

Figure 1 exemplifies the effect of the fast greedy removal algorithm. A 11-point subset $\mathcal{A}$ is obtained from a candidate solution set $\mathcal{M}, |\mathcal{M}| = 14$, using the individual contributions to $U^{\mathcal{K}^{\text{RSE}}}$. At iteration 1, we can see two crowded regions in the Pareto front approximation, involving points $\vec{a}_3$, $\vec{a}_4$, and $\vec{a}_5$ in the top and $\vec{a}_{11}$, $\vec{a}_{12}$, $\vec{a}_{13}$, and $\vec{a}_{14}$ in the bottom. According to line 5 of Algorithm 1, the point with the maximum contribution to $U^{\mathcal{K}^{\text{RSE}}}$ is deleted, begining with $\vec{a}_{12}$ whose contribution is $r_{12} = C(\vec{a}_{12}, \mathcal{A}) \approx 69.72$ at iteration 1. In the following two iterations, points $\vec{a}_4$ and $\vec{a}_{13}$ are deleted until $|\mathcal{A}| = 11$. The final subset shows that the fast greedy removal algorithm generates a well-diversified Pareto front approximation, due to the properties of $\mathcal{K}^{\text{RSE}}$ that penalizes solutions close to each other and rewards distant solutions.

The computational cost of the fast greedy removal algorithm is given as follows. At the beginning $|\mathcal{A}| = |\mathcal{M}| = M >> N$, thus, calculating $\boldsymbol{K}$ and $\vec{r}$ takes $O(M^2)$ and $O(M)$, respectively. The overall cost of the while-loop is $\sum_{i=0}^{M-N-1} M - i = \Theta\left(M^2\right)$ since the cost of each iteration is $\Theta(M - i)$ where i represents the number of the current deleted solutions (the cost of lines 8 to 10 is $O(1)$). Hence, the overall complexity of Algorithm 1 is $\Theta\left(M^2\right)$.

In contrast to the naïve greedy removal algorithm [3] and our previous work [12], the brand-new fast greedy removal algorithm has the lowest computational cost. This is due to the use of the memoization structure $\vec{r}$ that stores all the individual contributions $C(\vec{a}_i, \mathcal{A})$ and facilitates the identification of the worst-contributing solution in line 5 in $\Theta(M)$ time. In contrast, the naïve greedy removal algorithm [3] calculates each value $C(\vec{a}, \mathcal{A})$ using Equation (11) in $\Theta\left(M^2\right)$ time. Hence, the overall cost of calculating all the individual contributions in a single iteration is $\Theta\left(M^3\right)$. On the other hand, the fast greedy removal algorithm and our previous work [12] are similar to each other. The difference between the two algorithms is how to determine the worst-contributing solution. The previous algorithm recalculates all the contribution values which takes $\Theta\left(M^2\right)$ but in our improved version this is done in $\Theta(M)$ time.

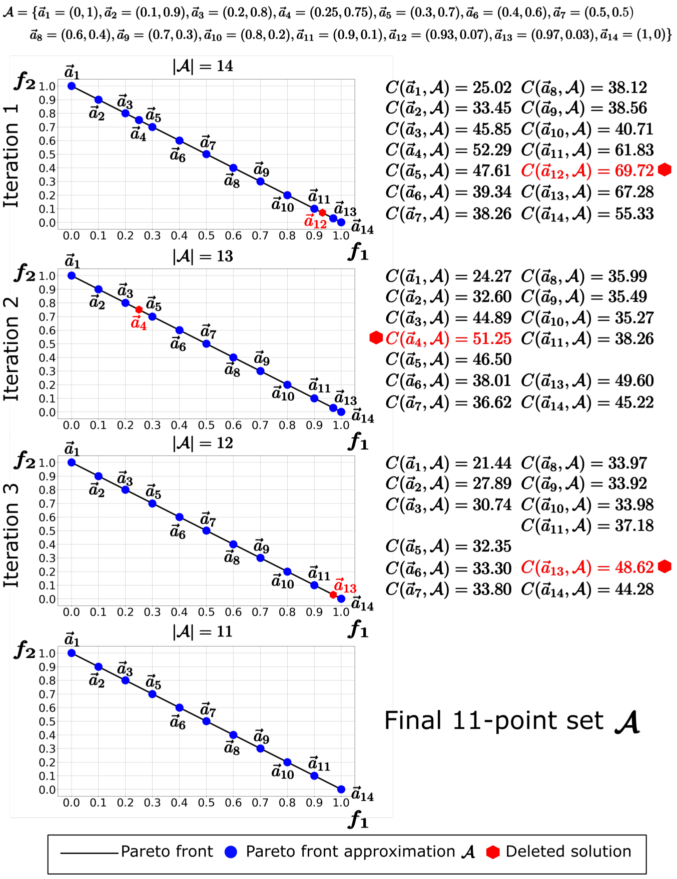

Figure 1: Effect of the fast greedy removal algorithm (see Algorithm 1) to generate a subset of size 11 from the original Pareto front approximation $\mathcal{A} = \mathcal{M}$ with 14 elements. At each iteration the solution with the maximum contribution C to $U^{\mathcal{K}^{\text{RSE}}}$ is deleted from $\mathcal{A}$ until the desired subset size $N = 11$ is reached. The deleted solution is highlighted using a red hexagon. All the values $r_i = C(\vec{a}_i, \mathcal{A})$ are shown per iteration.

To analyze the practical effect of our fast greedy removal algorithm in comparison with the implementations of the naïve greedy removal algorithm [3] and the one proposed in [12], we implemented them in the C language and we measured their computational time (in seconds). These experiments (and the remaining ones of the paper) were executed using the GCC compiler version 9.4.0 and the following environment: Intel(R) Core(TM) i7-6700 CPU 3.40 GHz (4 cores), having 16 GB RAM and Ubuntu 20.04.1 as the operating system. In Table 1, we show the results of the three greedy removal algorithms applied to Pareto front approximations, of at most 10,000 points, corresponding to the DTLZ1 problems with 2-10 objective functions, to select subsets of size 100 using $\mathcal{K}^{\text{RSE}}$ with $\alpha = m - 1$,

Table 1: Execution times (in seconds) of the three greedy removal algorithms applied to the Pareto fronts of the DTLZ1 problems with 2-10 objective functions. In each case, each algorithm searches for a subset of size 100 from a candidate set with at most 10,000 points. The best results are shown in grayscale.

MOP	Dim.	Naïve greedy removal algorithm [3]	Greedy removal algorithm [12]	Fast greedy removal algorithm
DTLZ1	2	154.175	40.562	0.269
	3	170.885	39.493	0.307
	4	182.323	39.327	0.316
	5	184.481	37.934	0.347
	6	193.756	38.282	0.339
	7	201.422	38.170	0.354
	8	205.331	39.181	0.361
	9	215.957	40.146	0.380
	10	225.494	39.518	0.404

where m is the number of objective functions. Our fast greedy removal algorithm obtains the fastest execution times in all the cases. Moreover, we can observe that the effect of m (i.e., the number of objective functions) on the computation time is small in all algorithms. It is worth noting that we obtained the selected Pareto fronts of at most 10,000 points from the PlatEMO platform [50].

5. Analysis of parameters

With the exception of $\mathcal{K}^{\text{COU}}$, the other five PPFs described in Section 2.2 depend on at least one parameter. Falcón-Cardona *et al.* [12] showed that as long as $\alpha \rightarrow 0$, the greedy removal algorithm based on $U^{\mathcal{K}^{\text{RSE}}}$ produces a subset of points only on the boundary of the Pareto front. Furthermore, in [14], it was pointed out that the parameters of $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$ also change the final discretization. (i.e., the finally selected subset). However, we exactly know neither the effect of each parameter of the selected PPFs on the selected subset nor how to specify the parameter values to generate well-diversified Pareto front approximations. In this section, we analyze the effect of each parameter in all the selected PPFs except for $\mathcal{K}^{\text{COU}}$ (which is parameterless). We will use $\mathcal{K}^{\text{COU}}$ later in Sections 6.1 and 6.2.

It is important to adjust the rate of decrease of $\mathcal{K}$ and the position of the minima by appropriately specifying the parameters in each PPF. These two aspects influence the amount of reward or penalization assigned to a pair of points, thus, affecting the diversity of the subset that Algorithm 1 produces. After a number of experiments, we found that only a single parameter (parameter α) has a dominant effect on these two aspects. α is a common parameter in $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$. Figure 2 supports the importance of α to alter the rate of change of $\mathcal{K}$ and the position of the minima for the five α-dependent PPFs. For each PPF, we plot the value of $\mathcal{K}$ as a function of the Euclidean distance between a pair of points for different values of α. We set $\alpha \in \{0.1, 0.5, 1, 2, 3, 5, 7, 10\}$ for $\mathcal{K}^{\text{RSE}}$; $\alpha \in \{0.1, 0.5, 1, 2, 64, 256, 1024, 2048\}$ for $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, and $\mathcal{K}^{\text{KRA}}$; and $\alpha \in \{0.001, 0.01, 0.1, 0.25, 0.5, 1, 2, 8\}$ for $\mathcal{K}^{\text{PTP}}$. The reasons to use distinct sets of values are twofold: 1) to see the effect of α in each PPF based on the analytical definition of the function, and 2) to clearly visualize the variation of the functions (e.g., setting $\alpha >> 8$ for $\mathcal{K}^{\text{PTP}}$ significantly increases its frequency, worsening the visualization). Since $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, and $\mathcal{K}^{\text{MTP}}$ are strictly monotonically decreasing functions, a different value of α only alters their rate of decrease as shown in the figure. It is worth noting that as $\alpha \rightarrow 0$, the value of $\mathcal{K}^{\text{RSE}}$ rapidly decreases; whereas for $\mathcal{K}^{\text{GAE}}$ and $\mathcal{K}^{\text{MPT}}$, the behavior is inverse. On the other hand, α changes the rate of decrease and the position of the minima of $\mathcal{K}^{\text{PTP}}$ and $\mathcal{K}^{\text{KRA}}$. For both functions, a value of α close to zero slows their rate of decrease, pushing the position of the minima to greater distance values.

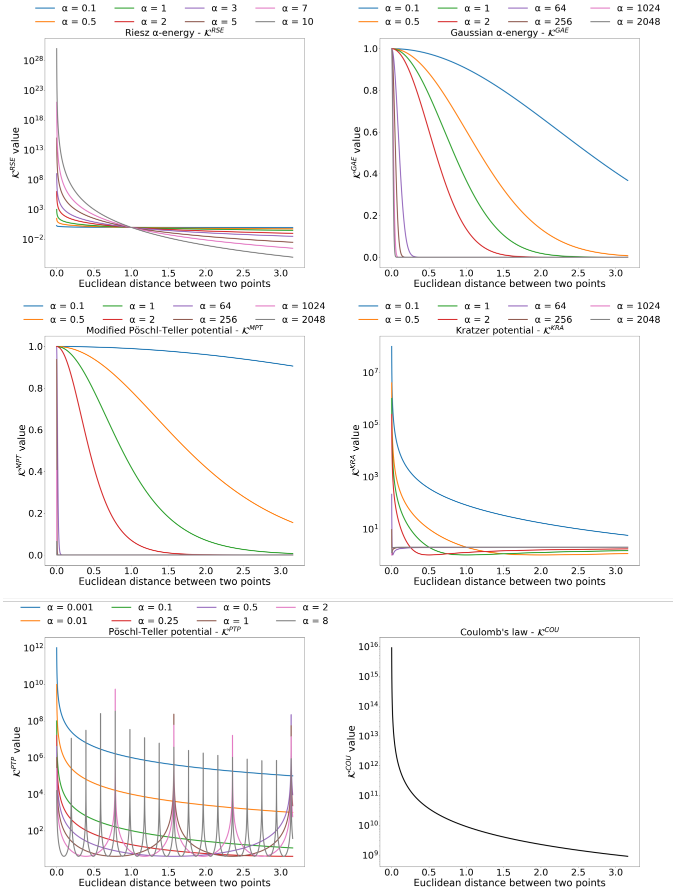

Figure 2: Each plot shows the variation of a PPF for different values of the parameter α, varying the Euclidean distance between two points. The set of values for α changes for $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$ to get a representative sample of their behavior. $\mathcal{K}^{\text{COU}}$ does not require parameters but we show how it changes for different distance values.

Despite α is the most critical parameter as shown in Figure 2, it is worth emphasizing why we did not consider the other parameters of $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$. Regarding $\mathcal{K}^{\text{MPT}}$, V_1 defines the maximum value that it takes when the distance of two points is zero. Hence, it does not alter the rate of change of the PPF value. As shown in Figure 2, $\mathcal{K}^{\text{PTP}}$ is a discontinuous and periodic function due to its trigonometric nature. However, to facilitate the discussion of its parameters V_1 and V_2, we focus on the interval $[0, T]$ that contains a single fragment of the function with vertical asymptotes at 0 and at T. If $V_1 = V_2$, the position of the minimum does not change regardless of the magnitude of the parameters but they only change the rate of decrease of the function. If $V_1 > V_2$, the location of the minimum approaches to T, and the decrease of the function is smoother. On the other hand, if $V_2 > V_1$, the position of the minimum approaches to 0, and the decrease of the function is more abrupt. Despite V_1 and V_2 alter the position of the minimum and the rate of decrease, α also controls both aspects. Hence, we set $V_1 = V_2 = 1$ for the sake of simplicity and we only let the values of α to vary. In the case of $\mathcal{K}^{\text{KRA}}$, V_1 does not change the location of the minimum but it only changes the rate of decrease of the function value. In this PPF, V_2 translates the graph vertically but does not change its shape. In consequence, since V_1 and V_2 do not completely influence the rate of decrease of the function as α does, they will be discarded from our analysis. Hence, we set $V_1 = V_2 = 1$ as in $\mathcal{K}^{\text{MPT}}$.

5.1. Approximating the optimal value of α

As mentioned in the previous section, the parameter α is the most critical for $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$. To emphasize its effect on finding $\mathcal{A}^*$ in PPFSS, we parameterize $U^{\mathcal{K}}$ and $\mathcal{K}$ by α as follows: $U_\alpha^{\mathcal{K}}$ and $\mathcal{K}_\alpha$. In consequence, we can define the following optimization problem:

$$\alpha^* = \arg\min_{\alpha>0} \min_{\substack{\mathcal{A}\subset\mathcal{M} \\ |\mathcal{A}|=N}} U_\alpha^{\mathcal{K}}(\mathcal{A}). \tag{12}$$

Calculating α^* is crucial to generate well-diversified Pareto front approximations according to our previous results [12, 14]. However, Equation 12 is a computationally expensive optimization problem that simultaneously involves solving PPFSS. In consequence, in this section, we propose two methods to approximate the optimal value of α for each one of the five α-dependent PPFs and their corresponding $U_\alpha^{\mathcal{K}}$ energy functions. It is worth noting that the approximate value of α is for $U_\alpha^{\mathcal{K}}$ that measures all the pair-potential interactions among the points in $\mathcal{A}$. Finding such an approximate optimal value of α leads to the construction a Pareto front approximation $\mathcal{A}$ with good diversity, regardless of the manifold geometry, dimension, and cardinality of $\mathcal{M}$. The first approximation algorithm is based on the Newton's method (Algorithm 2) and the second one uses a GA (Algorithm 3) to direct the search for α^*. Both methods depend on the fast greedy removal algorithm to approximate $\mathcal{A}^*$ in PPFSS. The Newton-based α approximation algorithm tackles PPFSS with $\mathcal{K}_\alpha^{\text{GAE}}$ and $\mathcal{K}_\alpha^{\text{MPT}}$ while the genetic-based α approximation deals with the other three instances of PPFSS (i.e., $\mathcal{K}_\alpha^{\text{RSE}}$, $\mathcal{K}_\alpha^{\text{PTP}}$, and $\mathcal{K}_\alpha^{\text{KRA}}$). The reason to use the two methods is described in the next section, using numerical results.

In the next two sections, we use the following common settings to analyze the performance of the two approximation methods. We obtained each candidate set $\mathcal{M}$ with $|\mathcal{M}| \approx 10,000$ uniformly sampled points from the PlatEMO platform [50]. Each $\mathcal{M}$ represents the Pareto front of some selected problems in the benchmark test suites Deb-Thiele-Laumanns-Zitzler (DTLZ) [51] and Walking-Fish-Group (WFG) [52]. Specifically, we focused on DTLZ1, DTLZ2, DTLZ5, DTLZ7, WFG1, WFG2, and WFG3 test problems with 2-10 objective functions (i.e., 7 × 9 test problems) because they all exhibit different Pareto front geometries, namely, linear, concave, degenerate, disconnected, and mixed. Moreover, we considered 8 specifications of the subset size $N \in \{25, 50, 75, 100, 150, 200, 250, 300\}$. Hence, we have a total of 504 configuration instances (7 × 9 test problems × 8 specifications of N) where we will calculate an approximate value of α^* for each configuration using Algorithms 2 and 3, considering the five α-dependent PPFs.

5.1.1. Newton-based α approximation for $U_\alpha^{\mathcal{K}}$

Algorithm 2 outlines the Newton-based α approximation. The inputs of the algorithm are a candidate set $\mathcal{M}$ and a subset size $N << |\mathcal{M}|$. The loop of the algorithm encompasses two steps. In line 4, the fast greedy removal algorithm is executed to obtain a subset $\mathcal{A}$ of size N from $\mathcal{M}$, using the current value of α (which was randomly initialized in line 1). Then, line 5 sketches the fixed-point iteration to calculate the value of α for the next iteration, using the current value, $D_\alpha U_\alpha^{\mathcal{K}}(\mathcal{A})$, and $D_\alpha^2 U_\alpha^{\mathcal{K}}(\mathcal{A})$. The while-loop continues if $|D_\alpha U_\alpha^{\mathcal{K}}(A)| \geq 1 \times 10^{-6}$; otherwise, line 6 returns the approximate value of α^*.

Algorithm 2 Newton-based α approximation

```
Input: Candidate set $\mathcal{M}$; subset size $N << |\mathcal{M}|$
Output: Best found $\alpha$ value
1: Randomly initialize $\alpha$
2: $\mathcal{A} = \mathcal{M}$
3: while $|D_\alpha U_\alpha^{\mathcal{K}}(\mathcal{A})| \geq 1 \times 10^{-6}$ do
4:     $\mathcal{A}$ = Fast greedy removal algorithm($\mathcal{M}, N, \alpha$)
5:     $\alpha = \alpha - D_\alpha U_\alpha^{\mathcal{K}}(\mathcal{A}) / D_\alpha^2 U_\alpha^{\mathcal{K}}(\mathcal{A})$
6: return $\alpha$
```

Before discussing the approximate values of α, it is worth indicating that Algorithm 2 only produces meaningful results for $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ and $U_\alpha^{\mathcal{K}^{\text{MPT}}}$. In Table SM-1 in the Supplementary Material, we present the numerical results of the Newton-based α approximation for $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ applied on the Pareto front of the 3-objective DTLZ1. We can see how $|D_\alpha U_\alpha^{\mathcal{K}^{\text{GAE}}}(\mathcal{A})|$ approaches to zero while the value of α approaches to promising values according to the experimental results in [14]. However, when Algorithm 2 uses $U_\alpha^{\mathcal{K}^{\text{RSE}}}$, the numerical results shown in Table SM-2 in the Supplementary Material (using the same set $\mathcal{M}$) show that α approaches to negative values which is not possible by definition. If we do not allow α to take negative values, Algorithm 2 pushes α to zero which encourages the generation of Pareto front approximations with almost all solutions in the boundary as shown in [12]. Regarding $U_\alpha^{\mathcal{K}^{\text{PTP}}}$ and $U_\alpha^{\mathcal{K}^{\text{KRA}}}$, Algorithm 2 exhibits an oscillatory behavior of $D_\alpha U_\alpha^{\mathcal{K}}$. In consequence, the generated values of α encourages the selection of subsets that represent a very small region of the overall Pareto front. These observations suggest minimizing $U_\alpha^{\mathcal{K}^{\text{RSE}}}$, $U_\alpha^{\mathcal{K}^{\text{PTP}}}$, and $U_\alpha^{\mathcal{K}^{\text{KRA}}}$ does not necessarily implies a high-diversity discretization. Hence, we utilize a GA to alleviate this problem (see Algorithm 3).

We executed Algorithm 2 for $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ and $U_\alpha^{\mathcal{K}^{\text{MPT}}}$, using the 504 configuration instances (defined in the previous section).

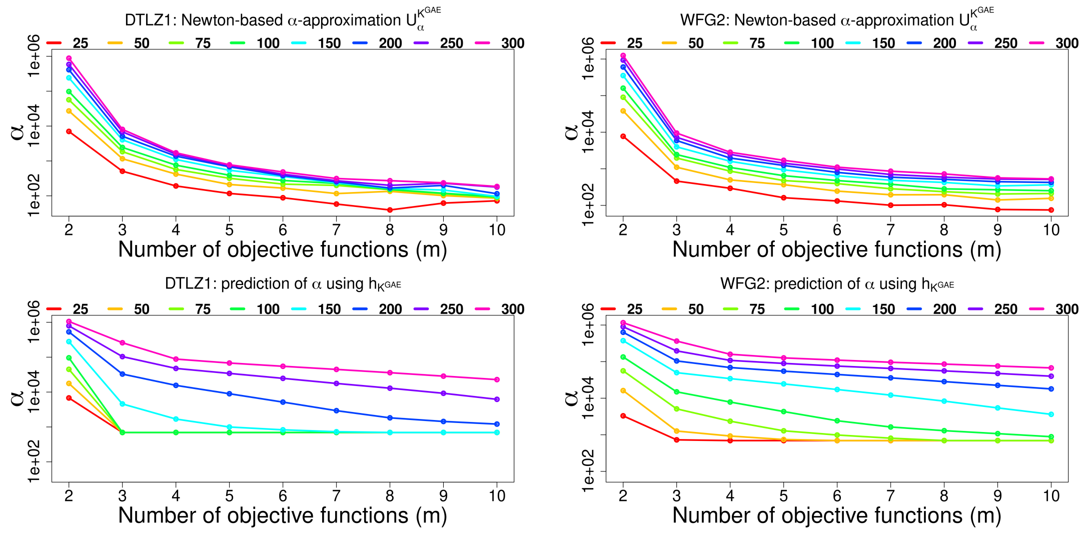

Figure 3: m-vs-α plots: value of α as a function of the dimension of the Pareto front for $U_\alpha^{\mathcal{K}^{\text{GAE}}}$. The first row shows the approximation of α using the Newton-based α approximation for DTLZ1 and WFG2 whereas the second row shows the corresponding predictions of α using the DNN model $h_{\mathcal{K}^{\text{GAE}}}$. The m-vs-α plots illustrate that the value of α depends on the geometry, dimension, and cardinality of the subset.

We analyze the behavior of α as a function of the number of objective functions for each Pareto front. Moreover, we study the variation of α for different subsets of size $N \in \{25, 50, 75, 100, 150, 200, 250, 300\}$ to emphasize the effect of the cardinality. In the first row of Figure 3, we show two m-vs-α plots for the Pareto fronts of DTLZ1 and WFG2, using $U_\alpha^{\mathcal{K}^{\text{GAE}}}$[7]. These two subplots show that the values of α follow a similar pattern. As long as the number of objective functions increases, the value of α decreases regardless of the subset size. In contrast, for a given number of objective functions, the value of α increases as long as the subset size increases. This behavior is also visible in other MOPs with both $\mathcal{K}^{\text{GAE}}$ and $\mathcal{K}^{\text{MPT}}$. On the other hand, when comparing m-vs-α plots for DTLZ1 and WFG2, we can observe that the range of α changes. This is a general behavior observed in the rest of test instances for both $\mathcal{K}^{\text{GAE}}$ and $\mathcal{K}^{\text{MPT}}$. Hence, these observations lead us to conclude that the value of α depends on the geometry of the Pareto front, its dimensionality, and the subset size, i.e., there is a hidden nonlinear function ψ such that $\alpha = \psi(g, m, N)$, where $g \in \{\text{degenerate}, \text{disconnected}, \text{linear}, \text{concave}, \text{mixed}, \text{convex}\}$.

5.1.2. Genetic-based α approximation for $U_\alpha^{\mathcal{K}}$

We utilized a GA to approximate the value of α^* for $U^{\mathcal{K}^{\text{RSE}}}$, $U^{\mathcal{K}^{\text{PTP}}}$, and $U^{\mathcal{K}^{\text{KRA}}}$, (see Algorithm 3). To design this GA, we used HV as the fitness function. This decision is based on an empirical analysis where we also employed $U_\alpha^{\mathcal{K}}$ (where $\mathcal{K}$ is set accordingly), the Pure Diversity indicator (PD) [53], and the Solow-Polasky Diversity (SPD) indicator [29] as fitness functions. However, minimizing $U_\alpha^{\mathcal{K}}$ and maximizing PD and SPD produces Pareto front approximations focused on small regions of the whole Pareto front. In contrast, the maximization of HV overcomes this issue since it rewards Pareto front approximations with a high spread. Algorithm 3 shows the genetic-based α approximation that evolves a population $P = \{\alpha_1, \ldots, \alpha_\mu\}$ using a $(\mu + \lambda)$-selection scheme. Each $\alpha_i \in P$ is used to generate a subset S_{α_i} using the fast greedy removal in Lines 2-5 and 9-12. These approximation sets are later evaluated using HV in Line 13. Since a larger HV value represents a better approximation set, only the μ individuals with the largest HV values survive the selection process. When the maximum number of generations $G_{\max}$ is reached, the value α associated to the maximum HV value is returned.

Algorithm 3 Genetic-based α approximation

```
Input: Candidate set $\mathcal{M}$; subset size $N(<< |\mathcal{M}|)$
Output: Best found $\alpha$ value
 1: Randomly initialize population $P = \{\alpha_1, \ldots, \alpha_\mu\}, \alpha_i > 0$
 2: $S = \emptyset$
 3: for all $\alpha_i \in P$ do
 4:     $S_{\alpha_i}$ = Fast greedy removal algorithm($\mathcal{M}, N, \alpha_i$)
 5:     $S = S \cup \{S_{\alpha_i}\}$
 6: $t = 0$
 7: while $t < G_{\max}$ do
 8:     Generate offspring solutions $Q = \{\alpha_1, \ldots, \alpha_\lambda\}$
 9:     $R = \emptyset$
10:     for all $\alpha_j \in Q$ do
11:         $R_{\alpha_j}$ = Fast greedy removal algorithm($\mathcal{M}, N, \alpha_j$)
12:         $R = R \cup \{R_{\alpha_j}\}$
13:     Calculate HV($S_{\alpha_k}$)$\forall S_{\alpha_k} \in S \cup R$
14:     Sort HV($S_{\alpha_k}$) in descending order
15:     Let the $\mu$ individuals that produce the greatest HV values
        survive in order to generate the next population $P$ and its
        set of subsets $S$.
16:     $t = t + 1$
17: return $\alpha = \arg\max_{\alpha \in P}$ HV($S_\alpha$)
```

[7] The rest of the figures are included in the Supplementary Material and at `https://github.com/jguillermofc/Pair-Potential-Energy/`.

We performed a single execution of Algorithm 3 to produce an approximation to α^* for each one of the 504 configuration instances for each $U_\alpha^{\mathcal{K}}$, $\mathcal{K} \in \{\mathcal{K}^{\text{RSE}}, \mathcal{K}^{\text{PTP}}, \mathcal{K}^{\text{KRA}}\}$. We employed the simulated-binary crossover and polynomial-based mutation as our genetic operators [38]. The crossover and mutation distribution indexes were set to 20, and the crossover and mutation probabilities were set to 1.0 and 0.1, respectively. Finally, we set $\mu = \lambda = 20$ and $G_{\max} = 100$ due to the high computational cost of the overall execution of the GA.

To analyze the behavior of α using the genetic-based α approximation, we also generate m-vs-α plots. In the first row of Figure 4, we show two m-vs-α plots for the Pareto fronts of DTLZ2 and WFG1, for $U^{\mathcal{K}^{\text{RSE}}}$. Unlike the Newton-based α approximation in Figure 3, the plots for DTLZ2 and WFG2 in the first row of Figure 4 are chaotic. However, it is still possible to extract a pattern. The chaotic behavior is also visible for other MOPs such as DTLZ1, DTLZ7, and WFG1, which are shown in the Supplementary Material. Both subplots (in the first row of Figure 4) show that the value of α depends on the geometry of the MOP. Furthermore, as in the case of the Newton-based α approximation, there is a tendency for the value of α to increase as the subset size increases, and the value also varies depending on the number of objective functions. On the other hand, there is a critical aspect that should be pointed out. In Figure 4 and in those in the Supplementary Material, the value of α for $U_\alpha^{\mathcal{K}^{\text{RSE}}}$ is similar to the empirical value that has been utilized, i.e, $\alpha = m - 1$ [12]. Regarding both $U_\alpha^{\mathcal{K}^{\text{PTP}}}$ and $U_\alpha^{\mathcal{K}^{\text{KRA}}}$, α tends to be in the interval $(0, 1)$ which contradicts the empirical values found in [14]. Taking into account $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ and $U_\alpha^{\mathcal{K}^{\text{MPT}}}$ using the Newton-based α approximation, the values of α are larger than 1, but the values associated to $U_\alpha^{\mathcal{K}^{\text{MPT}}}$ are considerably smaller than those of $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ which correspond to the empirical values found in [14].

The analysis of the results of Algorithms 2 and 3 indicates that the value of α depends on some unknown function $\psi(g, m, N)$. This function needs a categorical parameter g that indicates the geometry of the Pareto front, the number of objectives (i.e., m) objective functions, and the cardinality of the subset (i.e., N). Hence, establishing a fixed value of α as a general recommendation for every kind of instance is not possible to generate well-diversified Pareto front approximations, using the fast greedy removal algorithm. Moreover, approximating the optimal value of α (using either Algorithm 2 or 3), for every instance is not realistic since for a real-world MOP we do not know in advance the Pareto front. Consequently, we need a method (e.g., a machine learning technique) that approximates the function ψ and, thus, predicts the value of α, especially for unknown instances, taking into account the data generated in the study of the approximation of the value of α.

5.2. *Predicting the optimal value of α*

In this section, we propose to use a deep neural network (DNN) to define five prediction models: $h_{\mathcal{K}^{\text{RSE}}}$, $h_{\mathcal{K}^{\text{GAE}}}$, $h_{\mathcal{K}^{\text{MPT}}}$, $h_{\mathcal{K}^{\text{PTP}}}$, and $h_{\mathcal{K}^{\text{KRA}}}$. The core idea of this proposal is to utilize the prediction models in practical cases instead of a fixed value of α for each $U_\alpha^{\mathcal{K}}$. The generation of the five $h_{\mathcal{K}}$ predictors is necessary because the variation of the value of α is different for each $U_\alpha^{\mathcal{K}}$. Each model $h_{\mathcal{K}}$ receives as input an instance (g, m, N) and outputs a value of α for the corresponding $U_\alpha^{\mathcal{K}}$. The common DNN architecture consists of a sequence of densely connected layers (also called fully connected neural layers). The model comprises four dense layers and three dropout layers. As a default configuration, the first dense layer has 1024 neurons, and the second and third ones have 512 neurons, with a rectifier linear unit activation each. All dropout layers are defined with a dropout rate equal to 0.5. The model uses an Adam optimizer and minimum squared error (MSE) for loss calculation. The graphical representation of this model is shown in Figure 5. The last part of the model contains a dense layer with a single output neuron (its output is the predicted value) with no activation. This architecture was selected because it has been used for similar regression applications [54, 55]. The prediction models were implemented in the deep-learning framework Keras [56], and using the open-source deep learning library TensorFlow (version 2.3.0) [57].

For each prediction model $h_{\mathcal{K}}$, we generated a 504-instance dataset, using the information from Algorithms 2 and 3 as indicated in Sections 5.1.1 and 5.1.2, respectively. It is worth emphasizing that we executed these algorithms on some known instances (i.e., MOPs for which we have sufficiently dense Pareto fronts) to obtain a training set. Without executing these algorithms on some known instances and getting the corresponding α values, it would be impossible to construct a dataset to train the DNN model. Moreover, we try to generate input-output pairs of ψ as many as possible to train the DNN for approximating ψ. It is worth noting that the domain of ψ is large. Each dataset has 504 instances because we calculated the α values for 7 test problems with 2-10 objective functions, and 8 different specifications of the cardinality as described in Section 5.1. To ease the treatment of the categorical feature $g \in$ {degenerate, disconnected, linear, concave, mixed, convex}, this one was converted into a numerical value using the following integer encoding: degenerate: 1, disconnected: 2, linear: 3, concave: 4, mixed: 5, convex: 6. In addition, we implemented Min-Max scaling and standardization. Min-Max was selected due to its well-known use in machine learning, and standardization was chosen since it is much less affected by outliers, and for comparison reasons with Min-Max.

At this point of the investigation we would like to point out several observations related to our prediction work-flow and the architecture of the DNN. First, we recognize that it is possible for Pareto fronts to have one or multiple geometries. That is why, in this first study, it is our intention to test the utility of the prediction models with a static definition of the geometry. A possible improvement could consider the addition of more categories, or the construction of a multiclass classification model that determines the associated geometry of an N-point set $\mathcal{A}$. Other ideas include the introduction of other machine/deep learning regression models. Second, we do not claim that the current DNN architecture from Figure 5 is the best for the present problem. Nevertheless, as we will see later in this paper, the current prediction models are able to produce α values that promote well-diversified N-point sets $\mathcal{A}$. Probably, a smaller architecture could achieve better or equal results.

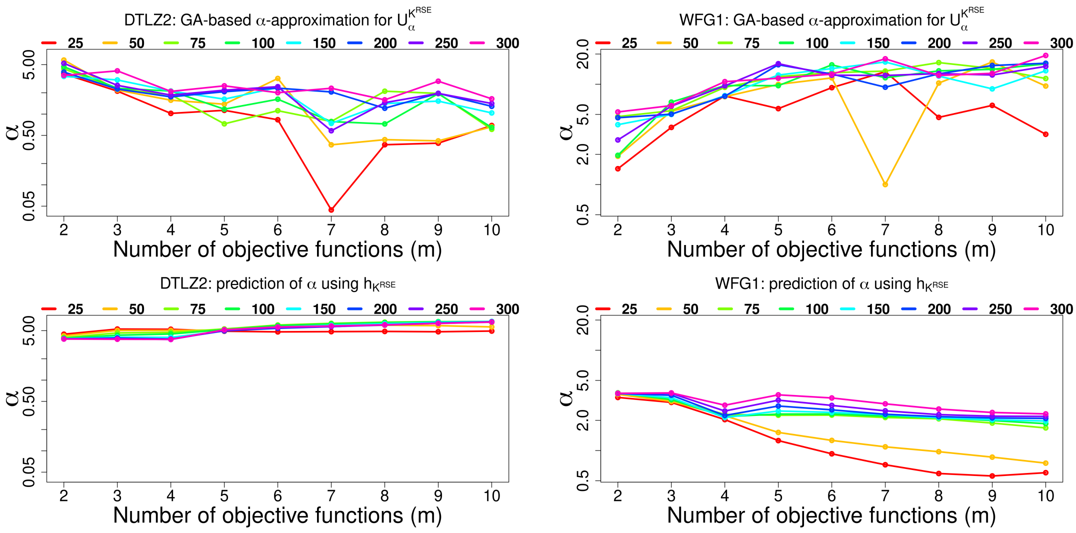

Figure 4: m-vs-α plots: value of α as a function of the dimension of the Pareto front for $U_\alpha^{\mathcal{K}^{\text{RSE}}}$. The first row shows the approximation of α using the genetic-based α approximation for DTLZ2 and WFG1 whereas the second row shows the corresponding predictions of α using the DNN model $h_{\mathcal{K}^{\text{RSE}}}$. The m-vs-α plots illustrate that the value of α depends on the geometry, dimension, and cardinality of the subset.

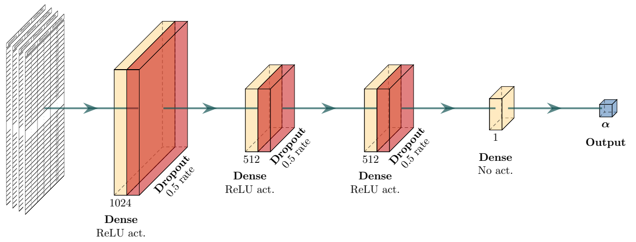

Figure 5: General DNN architecture used for the five prediction models: $h_{\mathcal{K}^{\text{RSE}}}$, $h_{\mathcal{K}^{\text{GAE}}}$, $h_{\mathcal{K}^{\text{MPT}}}$, $h_{\mathcal{K}^{\text{PTP}}}$, and $h_{\mathcal{K}^{\text{KRA}}}$. Each DNN model consists of four dense layers and three dropout layers. The first layer has 1024 neurons, the second and third layers have each one 512 neurons, and the last layer has one neuron. All dropout layers are defined with a dropout rate equal to 0.5. The output of the model is a prediction of the value of α for an instance (g, m, N).

Thus, it seems better to err on the side of choosing a large DNN rather than too small. Since all these ideas require a more exhaustive analysis, and are out of the scope of the present investigation, we leave them for future work.

Since this is the first time (to the best of our knowledge) that a prediction model is used to generate an appropriate value of α for $U_\alpha^{\mathcal{K}}$, several experiments were performed to assess the capability of the DNN models. We performed the following analysis for the five prediction models. First, the DNN model was trained using the raw data, i.e., only the geometry categorical value was changed to a numerical value, with no feature scaling and no K-fold cross validation. The reason for this was to have a baseline in which we could compare the effect of the feature scaling and K-fold cross-validation. Second, two feature scaling techniques (Min-Max and standardization) with no K-fold cross validation were introduced to observe if they helped the DNN model to improve the predictions and decide which of them is more suitable for the present analysis. Finally, for each experimental setup with raw data and feature scaling, K-fold cross-validation was introduced to validate the performance of the model.

The instances for each dataset were divided (uniformly at random) into training, validation and testing sets, as 322, 81, and 101, respectively. Then, the model was trained, using the training set, for 200 epochs, and the best-performing models for each dataset on the validation set were selected for prediction on the test set. When K-fold cross-validation is used, K is set to 10, with 362 and 41 samples for training and validation, respectively, chosen uniformly at random. For each of the K models, with $K-1$ training partitions, and the remaining evaluation partition were trained also for 200 epochs.

As in the case of the experiments to approximate the value of α using Algorithms 2 and 3, the prediction models generate numerous values of α. Hence, to summarize the results, we also provide m-vs-α plots where most of the figures are shown in the Supplementary Material and at the GitHub repository. Figure 3 presents in its second row the m-vs-α plots for DTLZ1 and WFG2 using $h_{\mathcal{K}^{\text{GAE}}}$. In comparison with the two corresponding subplots in the first row of the same figure, we can see that $h_{\mathcal{K}^{\text{GAE}}}$ have learned in a good way the behavior of the Newton-based α approximation for $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ since the ranges of the vertical axes in the four m-vs-α plots are very similar. In contrast to Figure 3, the m-vs-α plots associated with $h_{\mathcal{K}^{\text{RSE}}}$ in the second row of Figure 4 are not very similar to those in the first row. In these cases, $h_{\mathcal{K}^{\text{RSE}}}$ presents some difficulties to follow the values of α generated by the genetic-based α approximation. However, if we focus on the ranges of the vertical axes in the first and second rows, they are very similar. This implies that the values of α are in the expected range. Thus the difference between the first and second rows in Figure 4 is not a serious issue because it is expected to have some prediction errors in furtherance of gaining generalization power.

So far, we have performed experiments to approximate the

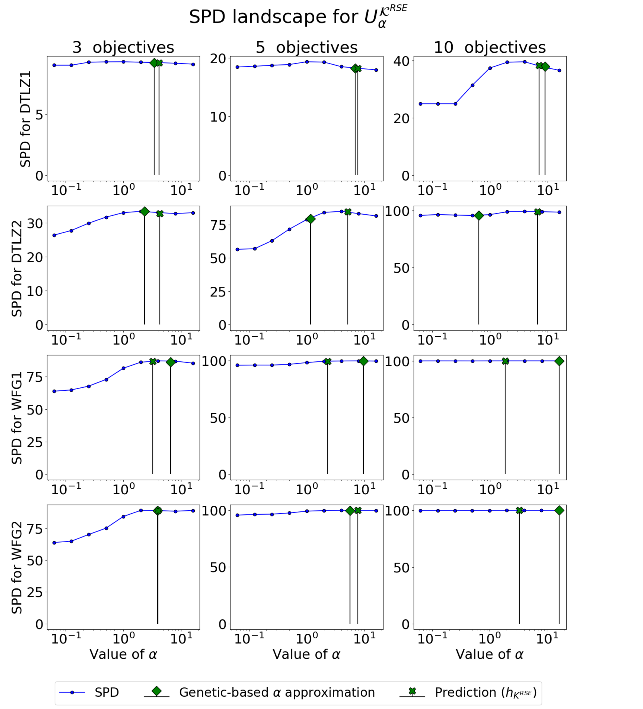

Figure 6: SPD landscape for Pareto front approximations ($\mathcal{A}$) of DTLZ1, DTLZ2, WFG1, and WFG2 with 3, 5, and 10 objective functions. Each point represents the value SPD($\mathcal{A}$). Each $\mathcal{A}$ is constructed via the fast greedy removal algorithm with $U_\alpha^{\mathcal{K}^{\text{RSE}}}$ and $\alpha \in \{-4, -3, \ldots, 3, 4\}$. We calculate the value of α using the genetic-based α approximation and $h_{\mathcal{K}^{\text{RSE}}}$. For these two values, we generate the corresponding $\mathcal{A}$ sets and we show their SPD values using markers.

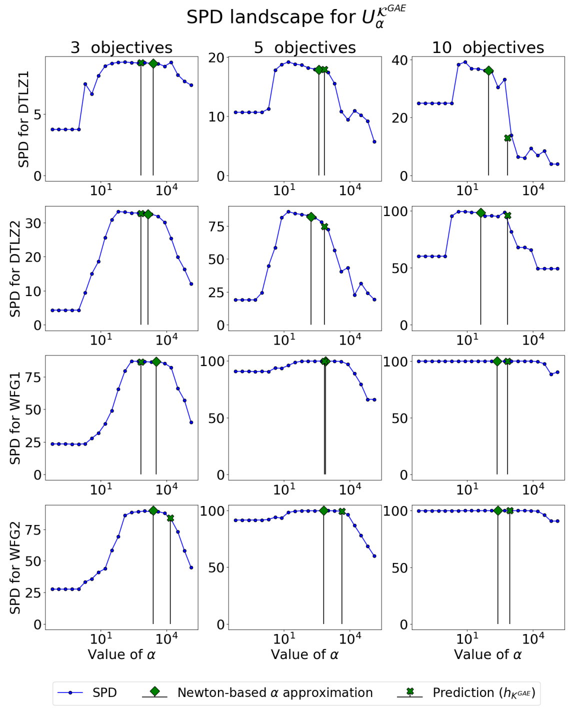

Figure 7: SPD landscape for Pareto front approximations ($\mathcal{A}$) of DTLZ1, DTLZ2, WFG1, and WFG2 with 3, 5, and 10 objective functions. Each point represents the value SPD($\mathcal{A}$). Each $\mathcal{A}$ is constructed via the fast greedy removal algorithm with $U_\alpha^{\mathcal{K}^{\text{GAE}}}$ and $\alpha \in \{-4, -3, \ldots, 18\}$. We calculate the value of α using the genetic-based α approximation and $h_{\mathcal{K}^{\text{GAE}}}$. For these two values, we generate the corresponding $\mathcal{A}$ sets and we show their SPD values using markers.

best value of α and we have created a DNN to predict this parameter. However, we have not examined information about the diversity of the resulting Pareto front approximations using the α values from the approximation and from the prediction models. Here, we analyze the diversity of the Pareto front approximations induced by different values of α, considering the values obtained by either the Newton-based or the genetic-based α approximation methods and the prediction models $h_{\mathcal{K}}$. Figure 6 shows the SPD landscape for Pareto front approximations $\mathcal{A}$ constructed via the fast greedy removal algorithm with $U_\alpha^{\mathcal{K}^{\text{RSE}}}$. We used the Pareto fronts of the DTLZ1, DTLZ2, WFG1, and WFG2 problems with 3, 5, and 10 objective functions. To get an insight of the SPD landscape, we sampled $\alpha \in \{-4, -3, \ldots, 3, 4\}$ and for each value we constructed an approximate set $\mathcal{A}$ to calculate SPD($\mathcal{A}$). The two markes in the figure show the SPD values of the subsets associated with the values of α obtained by the genetic-based approximation and the prediction model $h_{\mathcal{K}^{\text{RSE}}}$. The individual plots show how the SPD value changes over the different values of α, which suggests the promising regions of the parameter. From Figure 6, we can see that the two markers lie in the promising regions of α. That is, Figure 6 shows that the genetic-based α approximation and $h_{\mathcal{K}^{\text{RSE}}}$ appropriately specify the value of α to find well-diversified sets. Figure 7, which is similar to Figure 6, shows the SPD landscape associated with $U^{\mathcal{K}^{\text{GAE}}}$. Since $\mathcal{K}_\alpha^{\text{GAE}}$ accepts larger values of α than $\mathcal{K}_\alpha^{\text{RSE}}$ (see Figure 2), we sampled $\alpha \in \{-4, -3, \ldots, 18\}$. In the plots, we show the special SPD values associated with the Newton-based α approximation and $h_{\mathcal{K}^{\text{GAE}}}$. In these plots, the promising regions of α are clear thanks to a large number of samples and a wide range of sampled values. We can observe that the markers are always inside the promising region of α, which evidences a good performance of the approximation and prediction methods.

Figure 8 shows 100-point subsets generated by the fast greedy removal algorithm for three test problems of the Irregular MOPs (IMOPs) [58] test suite (namely, IMOP4, IMOP6, and IMOP7). It is worth noting that IMOPs were not used in Section 5.1, thus, they represent unknown instances for the prediction models. To generate each subset, we first used the corresponding $h_{\mathcal{K}}$ model to produce a value of α that Algorithm 1 requieres to construct the subset. Despite the irregularity of the Pareto front shapes of these MOPs, each prediction model produces a value of α that promotes good diversity as shown in each plot of the figure. Hence, Figure 8 supports the generalization power of the prediction models and encourages their utilization in practical cases in EMO as we show in the next section.

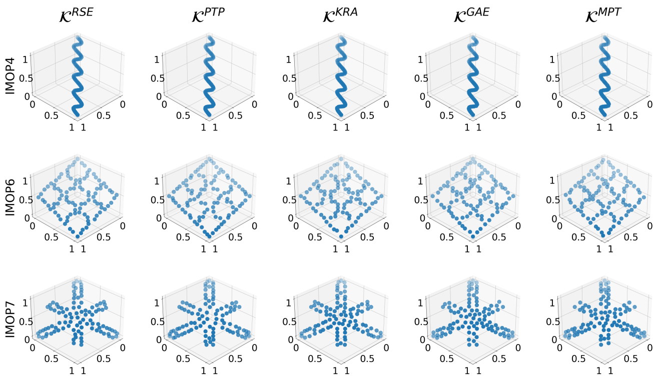

Figure 8: 100-point Pareto front approximations for the three-objective IMOP4, IMOP6, and IMOP7 problems. The fast greedy removal algorithm is applied to each problem using values of α provided by $h_{\mathcal{K}^{\mathrm{RSE}}}$, $h_{\mathcal{K}^{\mathrm{PTP}}}$, $h_{\mathcal{K}^{\mathrm{KRA}}}$, $h_{\mathcal{K}^{\mathrm{GAE}}}$, and $h_{\mathcal{K}^{\mathrm{MPT}}}$. IMOP problems represent unknown instances for the prediction models since we did not use them to construct the datasets.

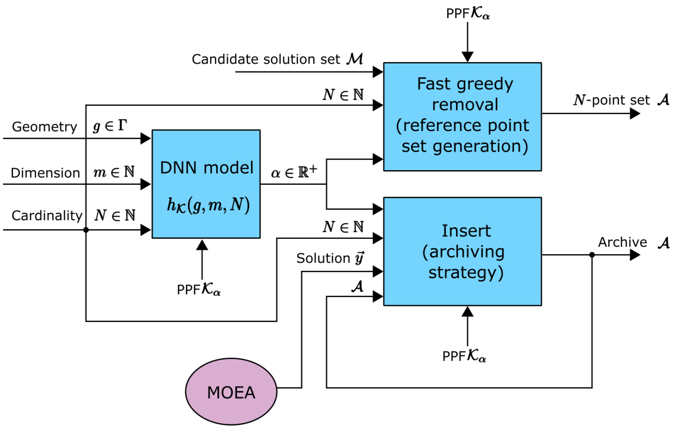

Figure 9: General framework to predict a value of α using a DNN model $h_{\mathcal{K}}$. The predicted value $\alpha = h_{\mathcal{K}}(g, m, N)$ can be employed in either the generator of reference point sets and a PPF-based archive unidirectionally connected to an MOEA.

6. On the utilization of PPFs in multi-objective optimization

Based on the analysis of the parameter α and the proposal of the DNN models to predict its value, in this section we propose the utilization of PPFs to construct well-diversified Pareto front approximations in EMO. Figure 9 shows a block diagram that represents the general framework for the utilization of PPFs. Once each DNN model $h_{\mathcal{K}}$ is trained, we can use it to calculate $\alpha = h_{\mathcal{K}}(g, m, N)$. The predicted value of α will be used in the following two application cases of PPFs in EMO:

1. **Construction of reference point sets**: the fast greedy removal algorithm uses the value of α to produce a subset $\mathcal{A} \subset \mathcal{M}$ of size N. The reference point sets are employed to calculate quality indicators to compare the Pareto front approximations of MOEAs [17]. Currently, there is no standard method to construct well-diversified reference point sets regardless of the Pareto front geometry.
2. **Archiving**. Every time an MOEA generates a new candidate solution $\vec{y} \in \Omega$, $\vec{y}$ is unidirectionally sent to a PPF-based archive for its insertion (if possible). The PPF-based archive aims to maintain a Pareto front approximation with better diversity, using the predicted value of α, than that of the MOEA's main population. It is worth noting that the PPF-based archive does not alter the performance of the MOEA (measured by the final population) due to the lack of a bidirectional connection between them.

In the following sections, we completely describe the two application cases to show the practical effect of the utilization of PPFs in EMO.

6.1. PPF-based reference point sets

A common approach to generate reference point sets (RPSs) in EMO is based on the use of weight vectors and scalarizing functions to define multiple SOPs [20, 46, 59]. However, it has been shown that this methodology cannot deal with all kinds of Pareto front shapes since some weight vectors may not intersect the associated manifold [49]. Currently, an open issue is how to generate N-point reference sets exhibiting good diversity regardless of the Pareto front geometry. In consequence, we propose to exploit the properties of PPFs to generate well-diversified RPSs, using the fast greedy removal algorithm (i.e., Algorithm 1), for benchmark problems whose Pareto front is represented by the set $\mathcal{M}$. Figure 9 sketches the proposed methodology. First, if $\mathcal{K}_\alpha \neq \mathcal{K}^{\mathrm{COU}}$, we calculate $\alpha = h_{\mathcal{K}}(g, m, N)$ to be used in the fast greedy removal algorithm to choose N points from the candidate set $\mathcal{M}$. The output will be an N-point reference set $\mathcal{A}$. In case $\mathcal{K}^{\mathrm{COU}}$ is used, we directly execute the fast greedy removal algorithm because this PPF does not depend on any parameter.

Table 2: Pareto front geometries of the MOPs used in the generation of reference point sets. Additionally, the reference point for the HV calculation is given.

MOP	Geometry	Simplex-like	Reference point
DTLZ1	Linear	✓	$(1, 1, \ldots, 1)$
DTLZ2	Concave	✓	$(2, 2, \ldots, 2)$
DTLZ5	Degenerate	✗	$(2, 2, \ldots, 2)$
DTLZ7	Disconnected	✗	$(1, 1, \ldots, 1, 21)$
WFG1	Mixed	✓	$(3, 5, \ldots, 2m + 1)$
WFG2	Disconnected	✓	$(3, 5, \ldots, 2m + 1)$
WFG3	Degenerate[8]	✗	$(3, 5, \ldots, 2m + 1)$
IDTLZ1	Linear	✗	$(1, 1, \ldots, 1)$
IDTLZ2	Convex	✗	$(2, 2, \ldots, 2)$
IWFG1	Mixed	✗	$(1, 1, \ldots, 1)$
IMOP1	Convex	✗	$(1.2, 1.2)$
IMOP2	Concave	✓	$(1.2, 1.2)$
IMOP3	Disconnected	✗	$(1.5, 1.2)$
IMOP4	Degenerate	✗	$(1.2, 1.2, 1.2)$
IMOP5	Disconnected	✗	$(1, 1, 2)$
IMOP6	Linear	✗	$(1.2, 1.2, 1.2)$
IMOP7	Concave	✗	$(1.2, 1.2, 1.2)$
IMOP8	Mixed	✗	$(1.2, 1.2, 3.2)$
VIE1	Convex	✗	$(4, 5, 4)$
VIE2	Mixed	✗	$(5, -15, -11)$
VIE3	Degenerate	✗	$(10, 18, 1)$

To analyze the advantages of PPF-based RPSs in comparison with weight vector-based RPSs, we use the following experimental setup. It worths highlighting that the core idea is to measure with QIs the diversity of RPSs.

- **Methods to generate RPSs.** We couple the six selected PPFs in Algorithm 1 to generate RPSs. We utilize three strategies to design weight vectors: simplex-lattice design (SLD) [61], uniform design using the Hammersley method (UDH) [59], and random-based design (RDD). For each weight vector, we define a SOP via the achievement scalarizing function (ASF) [23]. The underlying idea is to look for the objective vector in $\mathcal{M}$ that produces the minimum value of ASF. In this way, for a set of N weight vectors, we will produce a RPS of the same cardinality.

- **Pareto fronts.** In addition to the Pareto fronts of the selected DTLZ and WFG problems, we also use the inverted versions of these two suites, denoted as IDTLZ and IWFG, respectively [49]. We also considered Pareto fronts of the test suites Viennet (VIE) [62] and IMOP. The reason to use these benchmark problems is to have a wide variety of Pareto front shapes for low- and high-dimensional objective spaces. In Table 2, we introduce the selected MOPs, emphasizing their Pareto front shape and indicating if the geometry is correlated to the shape of an m-dimensional simplex.

- **Cardinalities.** In our experiments, we aim to generate RPSs of cardinality $N \in \{25, 50, 75, 100, 150, 200, 250, 300\}$. The fast greedy removal algorithm and both UDH- and RDD-based methods can generate RPSs of any cardinality. However, SLD is restricted to generate a combinatorial number of weight vectors given by $N = C^{H+m-1}_{m-1}$, where $H \in \mathbb{N}$ is a user-supplied parameter. Consequently, it is not always possible to generate N-point reference sets of any cardinality via SLD. To approximate the desired cardinalities $N_1 \approx 25$, $N_2 \approx 50$, $N_3 \approx 75$, $N_4 \approx 100$, $N_5 \approx 150$, $N_6 \approx 200$, $N_7 \approx 250$, and $N_8 \approx 300$, we use a two-layer SLD approach, where the number of weight vectors is given by: $C^{H_1+m-1}_{m-1} + C^{H_2+m-1}_{m-1}$, $H_1, H_2 \in \mathbb{N}$. Table 3 shows the approximated cardinalities $N_i, i = 1, \ldots, 8$, emphasizing the H_1 and H_2 values for 2 to 10 objective functions that we employ for all the methods to construct RPSs.

- **QIs**: To compare diversity of RPSs, we adopted the following QIs: inverted generational distance (IGD) [63], IGD$^+$ [18], the Pareto-compliant IGD$^+$ (IGD^{++}) [64], PD and SPD. We set the parameter θ to 10 to calculate SPD. IGD^{++} involves the calculation of HV that requires a reference point, which is shown in Table 2. To calculate IGD, IGD$^+$, and IGD^{++}, we use a RPS provided for each MOP in the PlatEMO platform [50]. It is worth noting that each RPS in PlatEMO has at most 10,000 points. Although IGD, IGD$^+$, and IGD^{++} are convergence-diversity QIs, we do not aim to measure convergence since all points in $\mathcal{M}$ are Pareto optimal.

Table 3: Cardinalities of the reference point sets for each specification of the number of objective functions (i.e., "Dim." in the table). The number of points is restricted by the two-layer SLD. For each case, it is shown the subset size and the values of H_1 and H_2. NA is placed when an approximated cardinality cannot be generated.

Dim.	$N_1 \approx 25$	$N_2 \approx 50$	$N_3 \approx 75$	$N_4 \approx 100$	$N_5 \approx 150$	$N_6 \approx 200$	$N_7 \approx 250$	$N_8 \approx 300$
2	$25^{H_1=24}_{H_2=0}$	$50^{H_1=49}_{H_2=0}$	$75^{H_1=74}_{H_2=0}$	$100^{H_1=99}_{H_2=0}$	$150^{H_1=149}_{H_2=0}$	$200^{H_1=199}_{H_2=0}$	$250^{H_1=249}_{H_2=0}$	$300^{H_1=299}_{H_2=0}$
3	$21^{H_1=5}_{H_2=0}$	$45^{H_1=8}_{H_2=0}$	$66^{H_1^{1}0}_{H_2=0}$	$91^{H_1=12}_{H_2=0}$	$136^{H_1=15}_{H_2=0}$	$190^{H_1=18}_{H_2=0}$	$231^{H_1=20}_{H_2=0}$	$300^{H_1=23}_{H_2=0}$
4	$24^{H_1=3}_{H_2=1}$	$35^{H_1=4}_{H_2=0}$	$56^{H_1=5}_{H_2=0}$	$84^{H_1=6}_{H_2=0}$	$120^{H_1=7}_{H_2=0}$	$165^{H_1=8}_{H_2=0}$	$220^{H_1=9}_{H_2=0}$	$286^{H_1=10}_{H_2=0}$
5	$20^{H_1=2}_{H_2=1}$	$50^{H_1=3}_{H_2=2}$	$75^{H_1=4}_{H_2=1}$	$85^{H_1=4}_{H_2=2}$	$126^{H_1=5}_{H_2=0}$	$210^{H_1=6}_{H_2=0}$	NA	NA
6	$21^{H_1=2}_{H_2=0}$	$42^{H_1=2}_{H_2=2}$	$62^{H_1=3}_{H_2=1}$	$77^{H_1=3}_{H_2=2}$	$147^{H_1=4}_{H_2=2}$	$182^{H_1=4}_{H_2=3}$	NA	$273^{H_1=5}_{H_2=2}$
7	$14^{H_1=1}_{H_2=1}$	$35^{H_1=2}_{H_2=1}$	$56^{H_1=2}_{H_2=2}$	$91^{H_1=3}_{H_2=1}$	$112^{H_1=3}_{H_2=2}$	$168^{H_1=3}_{H_2=3}$	$238^{H_1=4}_{H_2=2}$	$294^{H_1=4}_{H_2=3}$
8	$16^{H_1=1}_{H_2=1}$	$44^{H_1=2}_{H_2=1}$	$72^{H_1=2}_{H_2=2}$	NA	$128^{H_1=3}_{H_2=1}$	$156^{H_1=3}_{H_2=2}$	$240^{H_1=3}_{H_2=3}$	NA
9	$18^{H_1=1}_{H_2=1}$	$45^{H_1=2}_{H_2=0}$	$54^{H_1=2}_{H_2=1}$	$90^{H_1=2}_{H_2=2}$	NA	$174^{H_1=3}_{H_2=1}$	$210^{H_1=3}_{H_2=2}$	NA
10	$20^{H_1=1}_{H_2=1}$	NA	$65^{H_1=2}_{H_2=1}$	NA	$110^{H_1=2}_{H_2=2}$	NA	$230^{H_1=3}_{H_2=1}$	$275^{H_1=3}_{H_2=2}$

We performed a single execution of each method to generate a RPS for all the selected MOPs. It worths emphasizing that only the Pareto fronts of DTLZ, IDTLZ, WFG, and IWFG test suites are scalable in objective space. In every case, we generated RPSs of cardinality $N \in \{N_1, N_2, N_3, N_4, N_5, N_6, N_7, N_8\}$. All the RPSs were evaluated using the selected QIs. To summarize the numerical results, Tables SM-3, SM-4, SM-5, SM-6, and SM-5 in the Supplementary Material show the average ranking values for IGD^{++}, IGD$^+$, IGD, SPD, and PD, respectively. Each average ranking denotes the average place in the comparison that each approach obtained, considering a MOP for all the objective functions and specifications of cardinality. It is worth noting that for the construction of RPSs based on $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{PTP}}$, $\mathcal{K}^{\text{KRA}}$, $\mathcal{K}^{\text{GAE}}$, and $\mathcal{K}^{\text{MPT}}$, we used the corresponding prediction models to generate the values of α.

According to the average ranking results of IGD^{++}, IGD$^+$, IGD, SPD, and PD, the PPF-based RPSs exhibit better diversity than those generated by the weight vector-based RPSs. However, it is worth mentioning that the performance of the UDH-based methodology is better than that of the SLD- and RDD-based methods. Moreover, if we consider the method with the minimum average ranking for each MOP in Tables SM-3, SM-4, SM-5, SM-6, and SM-7 in the Supplementary Material, UDH encourages the generation of RPSs with better diversity for DTLZ1 and it also has a competitive performance for DTLZ2, WFG1, and WFG2. The reason for this behavior is that the Pareto fronts of these MOPs are highly correlated with the shape of an m-dimensional simplex on which weight vectors are generated. In contrast, SLD, UDH, and RDD have difficulties to promote well-diversified RPSs for MOPs with irregular Pareto front shapes (e.g., DTLZ7, IMOP1-IMOP8, and VIE1-VIE3). This is because some of these weight vectors cannot intersect the whole shape the Pareto front [49]. Figure 10 supports these numerical results. This figure shows RPSs, induced by $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{COU}}$, SLD, UDH, and RDD for the three-objective IDTLZ2, DTLZ7, IMOP7, and IWFG1 (which have

[8] Strictly speaking, WFG3 does not have a degenerate Pareto front. The Pareto front of WFG3 is partially degenerate [60].

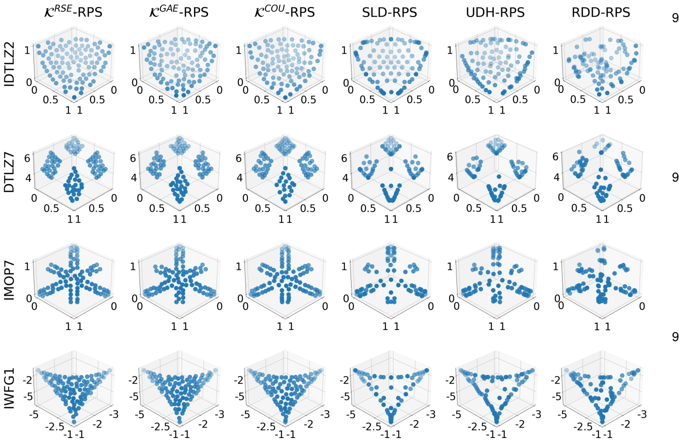

Figure 10: 91-point reference sets for the three-objective IDTLZ2, DTLZ7, IMOP7, and VIE3 problems. We compared the RPSs generated using weight vector-based approaches (SLD, UDH, and RDD) with $\mathcal{K}^{\text{RSE}}$-, $\mathcal{K}^{\text{GAE}}$-, and $\mathcal{K}^{\text{COU}}$-RPSs.

irregular Pareto front geometries). The weight vector-based methods have difficulties to generate well-diversified RPSs in contrast to PPF-based methods. However, weight vector-based approaches can be a good alternative for Pareto front shapes correlated with the shape of an m-dimensional simplex.

Regarding PPF-based RPSs, $\mathcal{K}^{\text{RSE}}$ induces well-diversified RPSs for DTLZ2, DTLZ5, WFG3, IDTLZ2, IMOP3, IMOP6, VIE1, VIE2, and VIE3. That is, $\mathcal{K}^{\text{RSE}}$ obtains the best average ranking for most of the selected QIs. Another important PPF is $\mathcal{K}^{\text{COU}}$ that produces RPSs with good diversity for DTLZ7, WFG1, IMOP1, IMOP7, IMOP8, and VIE3. We should point out that for some MOPs such as VIE3 a different PPF can be evaluated as the best by a different QI. IGD^{++} and IGD^{+} prefer $\mathcal{K}^{\text{COU}}$ whereas IGD and SPD prefer $\mathcal{K}^{\text{RSE}}$. This is an important observation because IGD^{++} and IGD^{+} have bias to prefer distributions where points are in the Pareto front's knee regions and along the Pareto front's boundary due to the Pareto compliance property [64]. On the other hand, IGD and SPD prefer points more uniformly distributed due to the lack of Pareto compliance. Hence, this result let us know that PPFs have different preferences which could be exploited to compensate the weaknesses of a PPF with the strengths of others. Finally, Figure 10 shows that PPF-based mechanisms can generate well-diversified Pareto front approximations. Furthermore, this figure also serves to argue that the regression models generate adequate α values for the fast greedy removal algorithm.

6.2. *PPF-based archiving*

As clearly pointed out in [65], the final population of any MOEA is not the best subset of the examined solutions during its execution. This means that the quality of the final Pareto front approximation generated by an MOEA can be improved by using an external archive. The usefulness of an external archive depends on the difference between the final population and the best subset of the examined solutions. In consequence, in this section, we propose a bounded PPF-based archive (denoted as $\mathcal{K}$-Archive) that can be coupled to any MOEA. The core idea is to examine all the solutions that an MOEA generates to be inserted into the $\mathcal{K}$-Archive, aiming to store a Pareto front approximation with better diversity than that of the MOEA's main population. Figure 9 shows the proposed configuration where an MOEA unidirectionally sends every newly created solution $\vec{y} \in \Omega$ to the $\mathcal{K}$-Archive ($\mathcal{A}$) of maximum size N. The archiving strategy also receives the value $\alpha = h_{\mathcal{K}}(g, m, N)$ (it is not necessary if $\mathcal{K}^{\text{COU}}$ is employed). It is worth emphasizing that due to the unidirectional connection from an MOEA to $\mathcal{A}$, the $\mathcal{K}$-Archive does not influence at all the performance of the selected MOEA. In our previous study [14], we showed that an MOEA is likely to delete some diversity-promising solutions due to its design principles. Hence, a $\mathcal{K}$-Archive aims to keep these diversity-promising solutions.

Algorithm 4 shows a generic MOEA to which a $\mathcal{K}$-Archive can be coupled. We should emphasize that any MOEA can be combined with a $\mathcal{K}$-Archive in Algorithm 4. An MOEA is unidirectionally connected to a $\mathcal{K}$-Archive, which is based on a user-supplied $\mathcal{K}_\alpha$ and its corresponding (if necessary) value of α predicted by $h_{\mathcal{K}}$. Every time an MOEA produces a new solution $\vec{y}_j$ in Line 7, $\vec{y}_j$ is sent to $\mathcal{A}$ to be inserted. In other words, the $\mathcal{K}$-Archive is used in a steady-state (or $(\mu+1)$) scheme. It is worth noting that some previous experiments indicated us that a $(\mu+1)$-$\mathcal{K}$-Archive promotes Pareto front approximations with better convergence and diversity properties than a $(\mu+\lambda)$-$\mathcal{K}$-Archive. However, using a $(\mu+1)$-$\mathcal{K}$-Archive implies a greater computational overhead. This is permissible due to the better convergence and diversity properties. Regardless of whether the MOEA keeps $\vec{y}_j$ in the main population or not, $\vec{y}_j$ has the possibility to survive in the $\mathcal{K}$-Archive as long as it helps to increase its diversity. Furthermore, it is worth emphasizing that the $\mathcal{K}$-Archive does not alter the optimization performance of the MOEA (i.e., the $\mathcal{K}$-Archive has no effect on the search behavior of the MOEA) because it never sends any stored solutions to the main population of the MOEA. In Line 8, $\vec{y}_j$ is tested to be stored in the $\mathcal{K}$-Archive $\mathcal{A}$, using Algorithm 5. This algorithm updates $\mathcal{A}$, and the standard evolutionary process of the MOEA continues until a stopping criterion is satisfied. Finally, both the main population P and $\mathcal{A}$ are returned so that they can be compared.

Algorithm 5 describes the steady-state update of a $\mathcal{K}$-Archive ($\mathcal{A}$) by examining the insertion of a solution $\vec{y}$. It is worth emphasizing that $\mathcal{A}$ is a bounded archive of maximum size given by A_{max}. The update strategy involves two phases. In the first stage (Lines 1 to 5), $\vec{y}$ is compared with all the solutions in $\mathcal{A}$ using the Pareto dominance relation. This comparison allows for maintaining selection pressure toward the Pareto front. If $\vec{y}$ is mutually non-dominated with every solution in $\mathcal{A}$, $\vec{y}$ is temporarily added to the archive. Secondly, in line 8, our fast greedy removal algorithm is executed, using $\mathcal{K}_\alpha$ and the given value of α. The execution of the fast greedy removal algorithm promotes diversity in $\mathcal{A}$ and ensures that the cardinality of the archive is at most of size A_{max}.

In our experiments, we couple a $\mathcal{K}$-Archive to four state-of-the-art MOEAs, namely, MOEA/D [46], AdaW [66], A-NSGA-III [67], and RVEA* [68]. It is worth mentioning that AdaW, A-NSGA-III, and RVEA* have adaptive mechanisms to gener-

Algorithm 4 Generic MOEA

```
Input: $\mu$: population size; $\lambda$: offspring population size; $\mathcal{K}_\alpha$:
    PPF; $A_{\max}$: maximum archive size; value of $\alpha$
Output: Main population and $\mathcal{K}$-Archive as Pareto front ap-
    proximations
Initialize the main Population $P = \{\vec{x}_1, \dots, \vec{x}_\mu\}$
Initialize $\mathcal{K}$-Archive $\mathcal{A}$ equals to $P$
while Stopping criterion is not satisfied do
    $Q = \varnothing$
    for $j = 1$ to $\lambda$ do
        Select mating parents from $P$
        Generate a new solution $\vec{y}_j$ by using variation operators
        $\mathcal{A}$ = Update $\mathcal{K}$ − Archive($\mathcal{A}, \vec{y}_j, \mathcal{K}, A_{\max}, \alpha$)
        $Q = P \cup \{\vec{y}_j\}$
    Select $\mu$ survival solutions from $Q$ to shape the next population $P$
return $\{P, \mathcal{A}\}$
```

Algorithm 5 Update $\mathcal{K}$-Archive

```
Input: $\mathcal{A}$: $\mathcal{K}$-Archive; $\vec{y}$: solution to be inserted; $\mathcal{K}_\alpha$: PPF;
    $A_{\max}$: maximum archive size; $\alpha$ value
Output: Updated $\mathcal{K}$-Archive $\mathcal{A}$
for all $\vec{a} \in \mathcal{A}$ do
    if $\vec{y} \prec \vec{a}$ then
        $\mathcal{A} = \mathcal{A} \setminus \{\vec{a}\}$
    else if $\vec{a} \preceq \vec{y}$ then
        return $\mathcal{A}$
$\mathcal{A} = \mathcal{A} \cup \{\vec{y}\}$
Normalize $\mathcal{A}$
$\mathcal{A}$ = Fast greedy removal algorithm($\mathcal{A}, A_{\max}, \alpha$)
return $\mathcal{A}$
```

Table 4: Parameters settings for MOEA/D, AdaW, A-NSGA-III, and RVEA*. $N50$, $N100$, and $N200$ denote three values of population size. $G^{N50}_{\max}$, $G^{N100}_{\max}$, and $G^{N200}_{\max}$ indicate the maximum number of generations for each population size specification. n_{WFG} and k_{WFG} denote the number of variables and the position-related parameter for the WFG and IWFG instances.

Dim.	$N50$	$G^{N50}_{\max}$	$N100$	$G^{N100}_{\max}$	$N200$	$G^{N200}_{\max}$	n_{WFG}	k_{WFG}
2	50	1000	100	500	200	250	24	2
3	45	1111	91	549	190	263	26	4
4	56	892	120	416	220	227	28	6
5	50	1000	126	396	210	238	30	8
6	42	1428	147	408	273	219	32	10
7	56	1250	112	625	238	294	34	12
8	44	1818	128	625	240	333	36	14
9	54	1666	90	1000	210	428	38	16
10	65	1538	110	909	230	434	40	18

ate well-diversified Pareto front approximations regardless of the geometry of the manifold. We employ problems from the DTLZ, IDTLZ, WFG, IWFG, IMOP, and VIE test suites as defined in Table 2 since they have different Pareto front shapes. This is to demonstrate the usefulness of the $\mathcal{K}$-Archive for maintaining a well-distributed solution set regardless of the Pareto front geometry. Since our computational experiments are not to compare the performance of different MOEAs, most test problems do not have challenging search difficulties. If we compare the performance of different MOEAs, it is needed to use more challenging test problems such as mDTLZ1-mDTLZ4 [69] and WOSGZ1-WOSGZ16 [70] in addition to the selected test problems in this paper. Problems from the DTLZ, WFG, IDTLZ, and IWFG suites were used for 2 to 10 objectives. We set the number of decision variables as $n = m + K - 1$ for the DTLZ and the IDTLZ problems, where m is the number of objectives and $K = 5$ for DTLZ1 and IDTLZ1, $K = 10$ for DTLZ2 and IDTLZ2 and DTLZ5, and $K = 20$ for DTLZ7. The number of decision variables and position-related parameters for WFG and IWFG instances is shown in the last two columns of Table 4. Regarding the IMOP and VIE problems, we followed their default specification about the number of decision variables. For each test problem, we examine three specifications of the population size, which are shown in the three columns labeled as $N50$, $N100$, and $N200$ of Table 4. The archive size $A_{\max}$ is the same as the population size. For each specification, the termination condition is shown as the maximum number of generations in the columns labeled as $G^{N50}_{\max}$, $G^{N100}_{\max}$, and $G^{N200}_{\max}$. We set the neighborhood size T to 20 and ASF was used in MOEA/D. We set the parameters of AdaW, A-NSGA-III, and RVEA* as specified by their authors. All the MOEAs use the simulated binary crossover and polynomial-based mutation as genetic operators.We set the crossover and mutation probabilities to 1.0 and $1/n$, respectively. The crossover distribution index was set to 20 for two- and three-objective problems and to 30 for MOPs with four or more objectives. The mutation distribution index was set to 20 for all cases. For each instance, we performed 30 independent executions of each MOEA unidirectionally connected to six $\mathcal{K}$-Archives, considering the six selected PPFs in this paper. Hence, we compared, using HV and SPD, the main population of each MOEA with the Pareto front approximation stored in each of $\mathcal{K}^{\text{RSE}}$-Archive, $\mathcal{K}^{\text{PTP}}$-Archive, $\mathcal{K}^{\text{KRa}}$-Archive, $\mathcal{K}^{\text{GAE}}$-Archive, $\mathcal{K}^{\text{MPT}}$-Archive, and $\mathcal{K}^{\text{COU}}$-Archive. The value of α was calculated using the corresponding prediction model $h_{\mathcal{K}}$, when necessary. We employed the one-tailed Wilcoxon rank-sum test to obtain statistical confidence, using a significance level of 0.05.

Due to space limitations, the complete numerical results for HV and SPD comparisons are shown in Tables SM-8 - SM-15 in the Supplementary Material. Those Tables show the HV and SPD values when comparing the MOEA's main population with the $\mathcal{K}$-Archives using $N100$ cardinality as defined in Table 4. Table 5 shows a summary of the numerical results by emphasizing the percentage of problems in which each $\mathcal{K}$-Archive is better than the MOEA's main population according to HV and SPD. Regarding HV, the archives based on $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, $\mathcal{K}^{\text{MPT}}$, $\mathcal{K}^{\text{KRA}}$, and $\mathcal{K}^{\text{COU}}$ contain better Pareto front approximations in

Table 5: Percentage of MOPs on which each $\mathcal{K}$-Archive obtains better HV or SPD values than the main population of an MOEA.

QI	MOEA	$\mathcal{K}^{\text{RSE}}$	$\mathcal{K}^{\text{GAE}}$	$\mathcal{K}^{\text{MPT}}$	$\mathcal{K}^{\text{PTP}}$	$\mathcal{K}^{\text{KRA}}$	$\mathcal{K}^{\text{COU}}$
HV	MOEA/D	91.43%	85.72%	85.72%	94.29%	94.29%	88.58%
	AdaW	51.43%	42.86%	40.00%	40.00%	42.86%	57.15%
	A-NSGA-III	68.58%	60.00%	62.86%	31.43%	45.72%	80.00%
	RVEA*	88.58%	77.15%	80.00%	37.15%	62.86%	85.72%
SPD	MOEA/D	100.00%	100.00%	97.15%	100.00%	100.00%	100.00%
	AdaW	88.58%	82.86%	88.58%	42.86%	62.86%	82.86%
	A-NSGA-III	94.28%	94.29%	97.15%	57.15%	74.29%	91.43%
	RVEA*	88.58%	88.58%	91.43%	45.72%	74.29%	91.43%

60% to 94.29% of the selected MOPs for MOEA/D, A-NSGA-III, and RVEA*. In contrast, the $\mathcal{K}^{\text{PTP}}$-Archive exhibits poor results. For AdaW, A-NSGA-III, and RVEA*, it can only produce better Pareto front approximations in 31.43% to 40% of the test problems. This behavior could be explained by a bad value of α predicted by $h^{\text{PTP}}_{\mathcal{K}}$. For AdaW, it is not easy to find better solution sets using the $\mathcal{K}$-Archives than the main population with respect to HV values. In this regard, $\mathcal{K}^{\text{COU}}$-Archive is the best one among the archives with 57.15% of better Pareto front approximations. However, we should emphasize that the preferences of HV depend on the specification of its reference point. The reference points for the calculation of HV, that we defined in Table 2, are close to the Pareto fronts. In consequence, HV rewards Pareto front approximations with more points in the interior part of the Pareto front and with fewer solutions in the boundary.

However, if we analyze the percentages for SPD, we can observe that the $\mathcal{K}$-Archives produce Pareto front approximations with better diversity in more than 80% of the MOPs for all the selected MOEAs. These results support our claim that the PPF-based archives would take advantage of some diversity-promising solutions deleted that are deleted by the baseline MOEA due to its design principles. Hence, the $\mathcal{K}$-Archives promote the preservation of well-diversified and convergent Pareto front approximations in MOEAs. Figure 11 shows a comparison of the SPD values throughout the evolutionary process of MOEA/D's main population and the $\mathcal{K}$-Archives. These plots for the three-, five-, and ten-objective DTLZ1 and IWFG1 indicate that the $\mathcal{K}$-Archives maintain a Pareto front approximation with better diversity than the one represented by MOEA/D's main population. This behavior is present even since the early stages of the evolutionary process which implies that $\mathcal{K}$-Archives take advantage of some solutions deleted by MOEA/D. Finally, it is worth discussing why the SPD values increase and, then, decrease as shown in the left-hand column of Figure 11. SPD is a diversity QI and unlike convergence QIs, it is not always expected a monotonically increasing behavior as in the case of HV. Depending on the configuration of the points and the separation between them, the value of SPD can increase. In the case of the SPD plots related to DTLZ1 (which is a multi-frontal MOP), a possible reason for the increase of the values of SPD is that MOEA/D found some local fronts far away from the true Pareto front where the points are very well separated. Some generations later, MOEA/D escaped from the local Pareto front which produced a reduction in the separation between the points, i.e., a loss of diversity as detected by SPD. Despite this, the SPD values related to the PPF-based archives remain better than that of the MOEA/D's main population.

HV-based comparison results in Tables SM-8 - SM-11 in the Supplementary Material indicate that $\mathcal{K}^{\text{COU}}$-Archive maintains on average the best Pareto front approximations. The reason for the high performance of $\mathcal{K}^{\text{COU}}$-Archive can be explained as follows. Based on the distributions of obtained solutions, we can categorize the examined PPFs into two groups. The first group encompasses $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, and $\mathcal{K}^{\text{MPT}}$. These PPFs have the tendency to generate Pareto front approximations with more solutions in the boundary and not so dense interior parts. On the other hand, the three PPFs in the other group (i.e., $\mathcal{K}^{\text{COU}}$, $\mathcal{K}^{\text{PTP}}$, and $\mathcal{K}^{\text{KRA}}$) reward Pareto front approximations with more solutions in the interior part of the manifold and fewer solutions in the boundary. Hence, HV rewards this second group due to the configuration of points when the reference point for HV calculation is not too far from the nadir point [71]. In this aspect, we should emphasize the properties of $\mathcal{K}^{\text{COU}}$. This PPF does not depend on α but it requires the definition of q_1 and q_2. In Section 2, we defined $q_1 = \sqrt{\sum_{i=1}^{m} u_i^2}$ and $q_2 = \sqrt{\sum_{i=1}^{m} v_i^2}$. We think that the good properties of $\mathcal{K}^{\text{COU}}$ are related to these two values, allowing it to penalize dominance-resistant solutions which are present during the evolutionary process. The two other best PPFs according to HV are $\mathcal{K}^{\text{PTP}}$ and $\mathcal{K}^{\text{KRA}}$ that belong to the same group. In contrast to HV, SPD rewards the group of archives using $\mathcal{K}^{\text{RSE}}$, $\mathcal{K}^{\text{GAE}}$, and $\mathcal{K}^{\text{MPT}}$ as shown in Tables SM-12 - SM-15 in the Supplementary Material. Furthermore, Figure 11 shows that the $\mathcal{K}$-Archives store Pareto front approximations with a better SPD value than that of MOEA/D's main population for almost all the evolutionary process. HV and SPD values indicate that there is not a single $\mathcal{K}$-Archive that outperforms the other ones in all kind of problems.

In conclusion, the use of a PPF-based archive is a promising option to help MOEAs to construct Pareto front approximations with good diversity. However, we should discuss the following aspects. First, the results show that there is no single PPF that promotes well-diversified sets for all problems due to the No-Free Lunch theorem. Among the examined PPFs, we can say that $\mathcal{K}^{\text{COU}}$ and $\mathcal{K}^{\text{RSE}}$ are good PPFs due to their simplicity. Especially, $\mathcal{K}^{\text{COU}}$ is parameterless. Secondly, the final Pareto front approximation stored in a $\mathcal{K}$-Archive strictly depends on the ability of the MOEA to explore the search space. If an MOEA cannot produce solutions on a specific region of the Pareto front, the $\mathcal{K}$-Archive is unable to compensate the exploration/exploitation ability of the MOEA.

6.3. Limitations on the use of PPFs

Regarding our experimental results, the utilization of PPFs in multi-objective optimization is promising to generate well-diversified Pareto front approximations. However, there are some limitations in our study that should be outlined. First of all, it is worth emphasizing that the amount of manifolds on which we can test our methods is infinite. For the study of

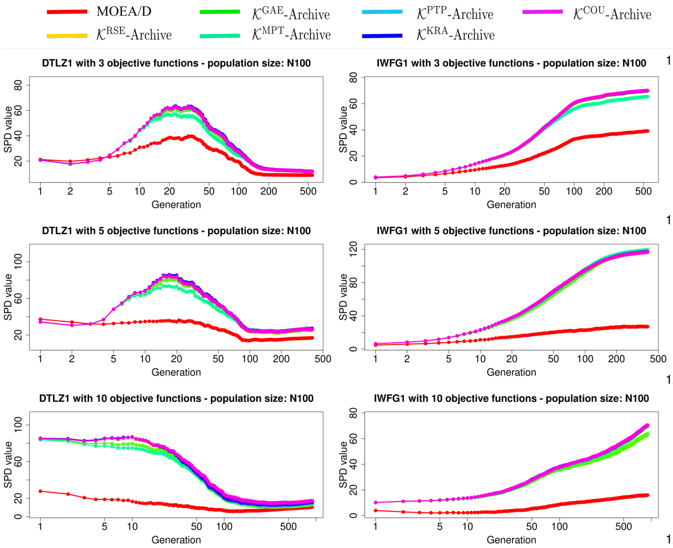

Figure 11: For each plot, the SPD values of the MOEA/D's main population and the six PPF-based archives at different generations. The goal is to observe the difference in diversity between the current population and the six archives over the execution of MOEA/D. From the plots, we observe that the PPF-based archives consistently maintain a better Pareto front approximation than the current population for DTLZ1 and IWFG1 with 3, 5, and 10 objectives.

α, we selected benchmark problems whose Pareto fronts represent different geometries (e.g., linear, concave, convex, disconnected, degenerate, or mixed), aiming to (hopefully) get a good sample of the universe of manifolds. At this point, a good question that would condition our results is how to selecte a representative sample of the infinite universe of manifolds.

Using the selected sample of Pareto fronts, we aim to approximate the optimal value of α for each manifold using two heuristic methods: the Newton- and genetic-based α approximations in Algorithms 2 and 3, respectively. Due to the heuristic nature of the methods, we expect to have an error in the approximation of α^* that collaterally impacts the quality of the datasets to train the DNN models. Furthermore, another aspect that conditions the quality of the datasets is the number of sampled manifolds as we previously emphasized. Despite these conditions, our experimental results showed that the DNN models have a good generalization power when tackling unknown manifolds as depicted in Figures 6, 7, and 8. A possible path path to increase the generalization power of the DNN models is to create a database of candidate sets $\mathcal{M}$ associated to Pareto fronts, varying the geometries, dimensions, and cardinalities. Moreover, it is still open the design of new heuristic methods to approximate the solution of PPFSS that could increase the performance of the Newton- and genetic-based α approximation methods.

Considering the two use-cases of the utilization of PPFs, we have the following observations. For the generation of reference point sets, it is necessary to have a discretization $\mathcal{M}$ with a large number of points for every Pareto front. However, there is not a set $\mathcal{M}$ for some benchmark problems (e.g., IDTLZ and IWFG suites), especially when the number of objective functions is greater than 3. On the other hand, if we focus on the PPF-based archive embedded into MOEA/D, AdaW, A-NSGA-III, and RVEA*, there is a clear limitation. In our experiments, we assumed that the geometry and dimension of the Pareto front of each MOP are known. Due to this assumption, we can consult the corresponding DNN model to obtain the α value that feeds the PPF-based archive during the execution of each MOEA. It is worth noting than when solving a real-world MOP, the characteristics of the Pareto front are unknown. However, the goal of the related experiments is to demonstrate that the PPF-based archive can take advantage of some solutions that an MOEA could delete due to the design of its selection mechanism. Our results support this goal in Tables SM-8 - SM-15 in the Supplementary Material. To overcome the need of prior knowledge of the Pareto front, a DNN could identify the geometry and dimension of the current Pareto front approximation at each iteration of an MOEA to send these values to our prediction model of α.

7. Conclusions and future work

In this paper, we analyzed six pair-potential energy functions to be employed as a mechanism to generate reference point sets and to improve the final Pareto front approximation of an MOEA. To this aim, we first studied the effect of a parameter that controls how these energy functions select a subset of solutions to lead to an N-point equilibrium configuration. We also proposed a DNN-based regression model to provide good parameter values that promote the generation of well-diversified Pareto front approximations. Based on our experimental results, we showed that the utilization of pair-potential energy functions leads to the generation of Pareto front approximations with high diversity values regardless of the Pareto front shape. This encourages the *a posteriori* nature of an MOEA that should provide Pareto front discretizations with good convergence, distribution, and spread properties. Among the six pair-potential energy functions, we found that $\mathcal{K}^{\text{RSE}}$ and $\mathcal{K}^{\text{COU}}$ are promising PPFs that ensure convergence and diversity. As part of our future work, we aim to exploit the properties of the two groups of pair-potential energy functions to generate approximation sets with better diversity properties. Moreover, we aim to design new heuristic methods to approximate the pair-potential energy-based subset selection problem.

References

[1] S. V. Borodachov, D. P. Hardin, E. B. Saff, Discrete Energy on Rectifiable Sets, 1st Edition, Springer Monographs in Mathematics, Springer-Verlag, 2019. doi:10.1007/978-0-387-84808-2.

[2] D. P. Hardin, E. B. Saff, Minimal Riesz energy point configurations for rectifiable *d*-dimensional manifolds, Advances in Mathematics 193 (1) (2005) 174–204. doi:10.1016/j.aim.2004.05.006.

[3] R. H. Gómez, C. A. C. Coello, A Hyper-Heuristic of Scalarizing Functions, in: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '17, Association for Computing Machinery, 2017, pp. 577–584. doi:10.1145/3071178.3071220.

[4] J. G. Falcón-Cardona, H. Ishibuchi, C. A. Coello Coello, M. Emmerich, On the Effect of the Cooperation of Indicator-Based Multiobjective Evolutionary Algorithms, IEEE Transactions on Evolutionary Computation 25 (4) (2021) 681–695. doi:10.1109/TEVC.2021.3061545.

[5] E. Zitzler, L. Thiele, Multiobjective Optimization Using Evolutionary Algorithms—A Comparative Study, in: A. E. Eiben, T. Bäck, M. Schoenauer, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature — PPSN V, Springer Berlin Heidelberg, 1998, pp. 292–301. doi:10.1007/BFb0056872.

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

[7] E. Dilettoso, S. A. Rizzo, N. Salerno, A Weakly Pareto Compliant Quality Indicator, Mathematical and Computational Applications 22 (1) (2017). doi:10.3390/mca22010025.

[8] J. G. Falcón-Cardona, H. Ishibuchi, C. A. C. Coello, Exploiting the Trade-off between Convergence and Diversity Indicators, in: 2020 IEEE Symposium Series on Computational Intelligence (SSCI), 2020, pp. 141–148. doi:10.1109/SSCI47803.2020.9308469.

[9] I. Giagkiozis, R. C. Purshouse, P. J. Fleming, Generalized Decomposition, in: R. C. Purshouse, P. J. Fleming, C. M. Fonseca, S. Greco, J. Shaw (Eds.), Evolutionary Multi-Criterion Optimization, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 428–442. doi:10.1007/978-3-642-37140-0_33.

[10] K. S. Bhattacharjee, H. K. Singh, T. Ray, Q. Zhang, Decomposition Based Evolutionary Algorithm with a Dual Set of Reference Vectors, in: 2017 IEEE Congress on Evolutionary Computation (CEC), 2017, pp. 105–112. doi:10.1109/CEC.2017.7969302.

[11] J. Blank, K. Deb, Y. Dhebar, S. Bandaru, H. Seada, Generating Well-Spaced Points on a Unit Simplex for Evolutionary Many-Objective Optimization, IEEE Transactions on Evolutionary Computation 25 (1) (2021) 48–60. doi:10.1109/TEVC.2020.2992387.

[12] J. G. Falcón-Cardona, H. Ishibuchi, C. A. Coello Coello, Riesz s-energy-based Reference Sets for Multi-Objective optimization, in: 2020 IEEE Congress on Evolutionary Computation (CEC), 2020, pp. 1–8. doi:10.1109/CEC48606.2020.9185833.

[13] L. A. Márquez-Vega, J. G. Falcón-Cardona, E. Covantes Osuna, Towards a Pareto Front Shape Invariant Multi-Objective Evolutionary Algorithm Using Pair-Potential Functions, in: Advances in Computational Intelligence, MICAI 2021, Vol. 13067 of Lecture Notes in Computer Science, Springer International Publishing, 2021, pp. 369–382. doi:10.1007/978-3-030-89817-5_28.

[14] J. G. Falcón-Cardona, E. Covantes Osuna, C. A. Coello Coello, An Overview of Pair-Potential Functions for Multi-objective Optimization, in: Evolutionary Multi-Criterion Optimization (EMO 2021), Vol. 12654 of Lecture Notes in Computer Science, Springer International Publishing, 2021, pp. 401–412. doi:10.1007/978-3-030-72062-9_32.

[15] K. Shang, T. Shu, H. Ishibuchi, Y. Nan, L. M. Pang, Benchmarking large-scale subset selection in evolutionary multi-objective optimization, Information Sciences 622 (2023) 755–770. doi:10.1016/j.ins.2022.11.155.

[16] X. Cai, H. Sun, Z. Fan, A diversity indicator based on reference vectors for many-objective optimization, Information Sciences 430–431 (2018) 467–486. doi:10.1016/j.ins.2017.11.051.

[17] M. Li, X. Yao, Quality evaluation of solution sets in multiobjective optimisation: A survey, ACM Computing Surveys 52 (2) (2019) 26:1–26:38.

[18] H. Ishibuchi, H. Masuda, Y. Tanigaki, Y. Nojima, Modified Distance Calculation in Generational Distance and Inverted Generational Distance, in: A. Gaspar-Cunha, C. H. Antunes, C. Coello Coello (Eds.), Evolutionary Multi-Criterion Optimization, 8th International Conference, EMO 2015, Vol. 9019 of Lecture Notes in Computer Science, Springer, Cham, 2015, pp. 110–125. doi:10.1007/978-3-319-15892-1_8.

[19] B. Li, J. Li, K. Tang, X. Yao, Many-Objective Evolutionary Algorithms: A Survey, ACM Comput. Surv. 48 (1) (2015). doi:10.1145/2792984.

[20] K. Deb, H. Jain, 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) (2014) 577–601. doi:10.1109/TEVC.2013.2281535.

[21] Y. Tian, R. Cheng, X. Zhang, F. Cheng, Y. Jin, An Indicator-Based Multiobjective Evolutionary Algorithm With Reference Point Adaptation for Better Versatility, IEEE Transactions on Evolutionary Computation 22 (4) (2018) 609–622. doi:10.1109/TEVC.2017.2749619.

[22] C. A. Coello Coello, G. B. Lamont, D. A. Van Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd Edition, Genetic and Evolutionary Computation, Springer New York, 2007. doi:10.1007/978-0-387-36797-2.

[23] K. Miettinen, Nonlinear Multobjective Optimization, 1st Edition, International Series in Operations Research & Management Science, Kluwer Academic Publishers, Boston, 1999. doi:10.1007/978-1-4615-5563-6.

[24] K. Rasmussen, Potential Energy Functions in Conformational Analysis, Vol. 37 of Lecture Notes in Chemistry, Springer Science & Business Media, 2012. doi:10.1007/978-3-642-45591-9.

[25] S.-H. Dong, Pöschl-Teller Potential, in: Factorization Method in Quantum Mechanics, Springer Netherlands, 2007, pp. 95–110. doi:10.1007/978-1-4020-5796-0_7.

[26] M. Hamzavi, S. Ikhdair, Approximate l-state solution of the trigonometric Pöschl-Teller potential, Molecular Physics 110 (24) (2012) 3031–3039. doi:10.1080/00268976.2012.695029.

[27] G. Simons, J. M. Parr, Robert G.and Finlan, New alternative to the Dunham potential for diatomic molecules, The Journal of Chemical Physics 59 (6) (1973) 3229–3234. doi:10.1063/1.1680464.

[28] J. D. Jackson, Classical electrodynamics, 3rd Edition, Wiley, 1999.

[29] V. Basto-Fernandes, I. Yevseyeva, A. Deutz, M. Emmerich, A Survey of Diversity Oriented Optimization: Problems, Indicators, and Algorithms, in: M. Emmerich, A. Deutz, O. Schütze, P. Legrand, E. Tantar, A.-A. Tantar (Eds.), EVOLVE – A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation VII, Springer International Publishing, 2017, pp. 3–23. doi:10.1007/978-3-319-49325-1_1.

[30] L. Martí, E. Segredo, N. Sánchez-Pi, E. Hart, Impact of selection methods on the diversity of many-objective pareto set approximations, Procedia Computer Science 112 (2017) 844–853, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8. doi:10.1016/j.procs.2017.08.077.

[31] G. Chen, J. Li, A diversity ranking based evolutionary algorithm for multi-objective and many-objective optimization, Swarm and Evolutionary Computation 48 (2019) 274–287. doi:10.1016/j.swevo.2019.03.009.

[32] Y. Wang, W. Gao, M. Gong, H. Li, J. Xie, A new two-stage based evolutionary algorithm for solving multi-objective optimization problems, Information Sciences 611 (2022) 649–659.

[33] W. Qiu, J. Zhu, G. Wu, M. Fan, P. N. Suganthan, Evolutionary many-Objective algorithm based on fractional dominance relation and improved objective space decomposition strategy, Swarm and Evolutionary Computation 60 (2021) 100776. doi:h10.1016/j.swevo.2020.100776.

[34] X. Li, X. Li, K. Wang, S. Yang, A strength pareto evolutionary algorithm based on adaptive reference points for solving irregular fronts, Information Sciences 626 (2023) 658–693. doi:10.1016/j.ins.2023.01.073.

[35] O. M. Shir, Niching in Evolutionary Algorithms, in: G. Rozenberg, T. Bäck, J. N. Kok (Eds.), Handbook of Natural Computing, Springer Berlin Heidelberg, 2012, pp. 1035–1069. doi:10.1007/978-3-540-92910-9_32.

[36] C. M. Fonseca, P. J. Fleming, Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, in: S. Forrest (Ed.), Proceedings of the ICGA-93: Fifth International Conference on Genetic Algorithms, Morgan Kauffman Publishers, 1993, pp. 416–423.

[37] N. Srinivas, K. Deb, Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms, Evolutionary Computation 2 (3) (1994) 221–248. doi:10.1162/evco.1994.2.3.221.

[38] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182–197. doi:10.1109/4235.996017.

[39] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm, in: K. Giannakoglou, D. Tsahalis, J. Periaux, P. Papailou, T. Fogarty (Eds.), EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, 2001, pp. 95–100.

[40] J. Zhou, X. Yao, F. T. Chan, L. Gao, X. Jing, X. Li, Y. Lin, Y. Li, A decomposition based evolutionary algorithm with direction vector adaption and selection enhancement, Information Sciences 501 (2019) 248–271. doi:doi.org/10.1016/j.ins.2019.05.083.

[41] F. Wang, Y. Li, H. Zhang, T. Hu, X.-L. Shen, An adaptive weight vector

guided evolutionary algorithm for preference-based multi-objective optimization, Swarm and Evolutionary Computation 49 (2019) 220–233. doi:h10.1016/j.swevo.2019.06.009.

[42] S. S. Das, M. M. Islam, N. A. Arafat, Evolutionary algorithm using adaptive fuzzy dominance and reference point for many-objective optimization, Swarm and Evolutionary Computation 44 (2019) 1092–1107. doi:10.1016/j.swevo.2018.11.003.

[43] G. Dai, C. Zhou, M. Wang, X. Li, Indicator and reference points co-guided evolutionary algorithm for many-objective optimization problems, Knowledge-Based Systems 140 (2018) 50–63. doi:10.1016/j.knosys.2017.10.025.

[44] S. Zapotecas-Martínez, A. López-Jaimes, A. García-Nájera, LIBEA: A Lebesgue Indicator-Based Evolutionary Algorithm for multi-objective optimization, Swarm and Evolutionary Computation 44 (2019) 404–419. doi:10.1016/j.swevo.2018.05.004.

[45] J. Luo, X. Huang, Y. Yang, X. Li, Z. Wang, J. Feng, A many-objective particle swarm optimizer based on indicator and direction vectors for many-objective optimization, Information Sciences 514 (2020) 166–202. doi:10.1016/j.ins.2019.11.047.

[46] Q. Zhang, H. Li, MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712–731. doi:10.1109/TEVC.2007.892759.

[47] N. Beume, B. Naujoks, M. Emmerich, SMS-EMOA: Multiobjective selection based on dominated hypervolume, European Journal of Operational Research 181 (3) (2007) 1653–1669. doi:10.1016/j.ejor.2006.08.008.

[48] H. Ishibuchi, R. Imada, N. Masuyama, Y. Nojima, Dynamic Specification of a Reference Point for Hypervolume Calculation in SMS-EMOA, in: 2018 IEEE Congress on Evolutionary Computation (CEC'2018), IEEE Press, 2018, pp. 701–708, iSBN: 978-1-5090-6017-7. doi:10.1109/CEC.2018.8477903.

[49] H. Ishibuchi, Y. Setoguchi, H. Masuda, Y. Nojima, Performance of Decomposition-Based Many-Objective Algorithms Strongly Depends on Pareto Front Shapes, IEEE Transactions on Evolutionary Computation 21 (2) (2017) 169–190. doi:10.1109/TEVC.2016.2587749.

[50] Y. Tian, R. Cheng, X. Zhang, Y. Jin, PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum], IEEE Computational Intelligence Magazine 12 (4) (2017) 73–87. doi:10.1109/MCI.2017.2742868.

[51] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable Test Problems for Evolutionary Multiobjective Optimization, in: A. Abraham, L. Jain, R. Goldberg (Eds.), Evolutionary Multiobjective Optimization: Theoretical Advances and Applications, Springer London, 2005, pp. 105–145. doi:10.1007/1-84628-137-7_6.

[52] S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, IEEE Transactions on Evolutionary Computation 10 (5) (2006) 477–506. doi:10.1109/TEVC.2005.861417.

[53] H. Wang, Y. Jin, X. Yao, Diversity assessment in many-objective optimization, IEEE Transactions on Cybernetics 47 (6) (2017) 1510–1522. doi:10.1109/TCYB.2016.2550502.

[54] F. Chollet, Deep Learning with Python, 1st Edition, Manning Publications Co., 2017.

[55] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd Edition, O'Reilly Media, Inc., 2019.

[56] F. Chollet, et al., Keras, https://keras.io (2015).

[57] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, Software available from https://www.tensorflow.org/ (2015).

[58] Y. Tian, R. Cheng, X. Zhang, M. Li, Y. Jin, Diversity Assessment of Multi-Objective Evolutionary Algorithms: Performance Metric and Benchmark Problems [Research Frontier], IEEE Computational Intelligence Magazine 14 (3) (2019) 61–74. doi:10.1109/MCI.2019.2919398.

[59] J. A. M. Berenguer, C. A. Coello Coello, Evolutionary Many-Objective Optimization Based on Kuhn-Munkres' Algorithm, in: A. Gaspar-Cunha, C. H. Antunes, C. Coello Coello (Eds.), Evolutionary Multi-Criterion Optimization, 8th International Conference, EMO 2015, Vol. 9019 of Lecture Notes in Computer Science, Springer, Cham, 2015, pp. 3–17. doi:10.1007/978-3-319-15892-1_1.

[60] H. Ishibuchi, H. Masuda, Y. Nojima, Pareto Fronts of Many-Objective Degenerate Test Problems, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 807–813. doi:10.1109/TEVC.2015.2505784.

[61] I. Das, J. E. Dennis, Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems, SIAM Journal on Optimization 8 (3) (1998) 631–657. doi:10.1137/S1052623496307510.

[62] D. A. V. Veldhuizen, Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations, Ph.D. thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, USA (May 1999).

[63] C. A. Coello Coello, N. Cruz Cortés, Solving Multiobjective Optimization Problems Using an Artificial Immune System, Genetic Programming and Evolvable Machines 6 (2) (2005) 163–190. doi:10.1007/s10710-005-6164-x.

[64] J. G. Falcón-Cardona, M. T. M. Emmerich, C. A. Coello Coello, On the Construction of Pareto-compliant Combined Indicators, Evolutionary Computation (2022) 1–27doi:10.1162/evco_a_00307.

[65] M. Li, X. Yao, An Empirical Investigation of the Optimality and Monotonicity Properties of Multiobjective Archiving Methods, in: K. Deb, E. Goodman, C. A. Coello Coello, K. Klamroth, K. Miettinen, S. Mostaghim, P. Reed (Eds.), Evolutionary Multi-Criterion Optimization, Springer International Publishing, Cham, 2019, pp. 15–26.

[66] M. Li, X. Yao, What Weights Work for You? Adapting Weights for Any Pareto Front Shape in Decomposition-Based Evolutionary Multiobjective Optimisation, Evolutionary Computation 28 (2) (2020) 227–253. doi:10.1162/evco_a_00269.

[67] H. Jain, K. Deb, An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach, IEEE Transactions on Evolutionary Computation 18 (4) (2014) 602–622. doi:10.1109/TEVC.2013.2281534.

[68] R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 773–791. doi:10.1109/TEVC.2016.2519378.

[69] Z. Wang, Y.-S. Ong, H. Ishibuchi, On Scalable Multiobjective Test Problems With Hardly Dominated Boundaries, IEEE Transactions on Evolutionary Computation 23 (2) (2019) 217–231. doi:10.1109/TEVC.2018.2844286.

[70] Z. Wang, Y.-S. Ong, J. Sun, A. Gupta, Q. Zhang, A Generator for Multiobjective Test Problems With Difficult-to-Approximate Pareto Front Boundaries, IEEE Transactions on Evolutionary Computation 23 (4) (2019) 556–571. doi:10.1109/TEVC.2018.2872453.

[71] A. Auger, J. Bader, D. Brockhoff, E. Zitzler, Theory of the Hypervolume Indicator: Optimal μ-Distributions and the Choice of the Reference Point, in: Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms, FOGA '09, Association for Computing Machinery, 2009, pp. 87–102. doi:10.1145/1527125.1527138.