



Article

Variation Rate to Maintain Diversity in Decision Space within Multi-Objective Evolutionary Algorithms [†]

Oliver Cuate ^{1,*} and Oliver Schütze ^{2,*}

¹ Computer Science Department, Cinvestav-IPN, Mexico City 07360, Mexico

² Dr. Rodolfo Quintero Ramirez Chair, UAM Cuajimalpa, Mexico City 05370, Mexico

* Correspondence: ocuate@computacion.cs.cinvestav.mx (O.C.); schuetze@cs.cinvestav.mx (O.S.)

† This paper is an extended version of our paper published in the 10th International Conference on Evolutionary Multi-Criterion Optimization (EMO 2019), East Lansing, MI, USA, in 10–13 March 2019.

Received: 30 July 2019; Accepted: 11 September 2019; Published: 13 September 2019



Abstract: The performance of a multi-objective evolutionary algorithm (MOEA) is in most cases measured in terms of the populations' approximation quality in objective space. As a consequence, most MOEAs focus on such approximations while neglecting the distribution of the individuals of their populations in decision space. This, however, represents a potential shortcoming in certain applications as in many cases one can obtain the same or very similar qualities (measured in objective space) in several ways (measured in decision space). Hence, a high diversity in decision space may represent valuable information for the decision maker for the realization of a given project. In this paper, we propose the Variation Rate, a heuristic selection strategy that aims to maintain diversity both in decision and objective space. The core of this strategy is the proper combination of the averaged distance applied in variable space together with the diversity mechanism in objective space that is used within a chosen MOEA. To show the applicability of the method, we propose the resulting selection strategies for some of the most representative state-of-the-art MOEAs and show numerical results on several benchmark problems. The results demonstrate that the consideration of the Variation Rate can greatly enhance the diversity in decision space for all considered algorithms and problems without a significant loss in the approximation qualities in objective space.

Keywords: evolutionary computation; multi-objective optimization; decision space diversity

1. Introduction

In many areas such as Economy, Finance, or Industry, the problem arises naturally that several conflicting objectives have to be optimized concurrently [1,2]. Such problems are called multi-objective optimization problems (MOPs) in literature [3–5]. The solution set of an MOP (in decision space) is called the Pareto set and its image (defined in objective space) the Pareto front. One important characteristic of continuous MOPs is that their Pareto sets and fronts typically form $(k - 1)$ -dimensional objects, where k is the number of objectives considered in the problem. In many applications, the Pareto front is of primary interest as it contains information about the desired qualities of each selected solution. As a consequence, almost all existing MOEAs focus on approximations in objective space while entirely neglecting the distribution of the individuals in decision space. There exist, however, also applications where the values of the solutions in decision variable space are of great importance. As an example, consider that the amount of a certain resource (i.e., the value of a variable x_i) used to obtain the desired quality (measured in objective space) is important. For two solutions that are equal or similar in objective space, one may prefer the one that has a lower value of x_i . Another example is that the variable x_i could represent the launch date of a project, as, e.g., in [6] in the context of space mission design. For such problems, different values of x_i directly relate to different timescales in the

realization of the project. In that case, the decision maker may select one “optimal” realization of the project, and keep solutions with similar objective values but later launch dates as back-up solutions. For such problems, the sole consideration of the approximation quality in objective space represents a potential shortcoming. On the one hand, it is of course possible to formulate all such problems via additional constraints and objectives. On the other hand, such re-modellations of the problem lead in general to a higher complexity compared to the original problem. For instance, by each added objective, the dimension of the Pareto set/front increases by one. Hence, in this context, there exists an additional challenge in solving a given MOP, since we have to find a suitable approximation to the optimal set both in objective and decision space, in order to provide a satisfying overview of the possible solutions to the decision maker. Another application can be found in [7], where data analysis techniques are used to discover patterns and rules. Here, authors conclude that such process of innovation by optimization (“innovation”) has an enormous potential to revolutionize the engineering of the design process in the industry.

In this paper, we propose a framework for equipping MOEAs with a mechanism that performs an exploration of both decision and objective spaces. The underlying idea of this proposal, called Variation Rate, is to measure the spread via using the averaged distance in decision space for elements for which function values are close in objective space. After discussion of the general framework, we present implementations of variants of NSGA-II [8] (elitist algorithm), SMS-EMOA [9] (indicator based algorithm), MOEA/D [10] (decomposition based algorithm), and NSGA-III [11] (designed for the treatment of MOPs with many objectives). Further on, we demonstrate the effectiveness of our approach on several benchmark problems, where we show that, compared to the base variants, we greatly improve the approximation qualities in decision space without any significant loss in the qualities in objective space. A preliminary study of this work can be found in [12], where the discussion on the proposed method is reduced, and where it has only been applied to NSGA-III.

The remainder of the paper is organized as follows: in Section 2, we briefly present the background required for the understanding of this paper and discuss the related work. In Section 3, we first present the general framework of the Variation Rate and further on provide particular implementations for four different MOEAs that are representative of the state-of-the-art. In Section 4, we present some numerical results on selected benchmark problems using both the four base MOEAs as well as their variants that use the Variation Rate. Finally, in Section 5, we discuss the advantages of the proposed approach and discuss possible paths for future research.

2. Background and Related Work

Optimization refers to finding the best possible solution to a problem given a set of constraints [4]. Multi-objective optimization refers to the simultaneous optimization of multiple and usually conflicting objectives. More precisely, a multi-objective optimization problem (MOP) with k objectives is mathematically defined as follows:

$$\min_{x \in D} F(x), \quad (1)$$

where $D \subset \mathbb{R}^n$ is the domain and $F : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^k$ is called the objective function.

The optimality of an MOP is defined by the concept of *strict dominance*. Let $v, w \in \mathbb{R}^k$, the vector v is less than w ($v <_p w$), if $v_i < w_i$ for all $i \in \{1, \dots, k\}$; the relation \leq_p is defined analogously. A vector $y \in D$ is *dominated* by a vector $x \in D$ ($x \prec y$) with respect to (1) if $F(x) \leq_p F(y)$ and $F(x) \neq F(y)$, else y is called non-dominated by x . A point $x^* \in \mathbb{R}^n$ is Pareto optimal to (1) if there is no $y \in D$ that dominates x . The set of all the Pareto optimal points P_D is called the Pareto set and its image $F(P_D)$ is called the Pareto front. Typically, i.e., under certain mild smoothness assumption on the model, both Pareto set and front form at least locally $(k - 1)$ -dimensional objects.

Unlike evolutionary algorithms for single objective optimization problems (SOPs), maintaining diversity in decision space is not a priority for most MOEAs; most of the performance indicators are developed in order to measure the accuracy based only on the objective function (e.g., the hypervolume [13] and the Degree of Approximation [14]). As exceptions, we have some of the measures used for multimodal optimization (see [15]) and two particular examples. The first one is the Δ_p indicator [16,17], which can be viewed as an averaged Hausdorff distance and which actually measures the distance between two general sets and we can use it as an indicator both in objective space as well as in decision space. The other work that can deal with this aspect is the diversity integrating multiobjective optimizer (DIOP) [18], which is user-defined and concurrently optimizes two set-based diversity measures, one in decision space and the other in objective space. In this work, the relationship between the two set-based diversity measures is conceived as a bi-objective optimization problem and it is solved via a weighted sum of the two diversity indicators. In particular, the authors consider the Solow–Polasky measure as it satisfies three requirements: monotonicity in varieties, twinning, and monotonicity in the distance.

Although works that explicitly consider at the same time variables and objectives are scarce, one can find some related work on this topic in [19] and some specific algorithms. For instance, the NSGA [20] (the algorithm that precedes the well-known NSGA-II) uses fitness sharing in decision space. In [21], some possible techniques are proposed to spread out solutions both in objective and decision space: *pointwise expansion*, *threshold sharing*, *sequential sharing*, *simultaneous sharing multiplicative*, and *simultaneous sharing additive*. It is important to point out that the above approaches are only part of the discussion of the paper and they were not implemented; the implemented algorithm was the Niched Pareto GA, a method with phenotypic sharing. In addition, all of the described techniques depend on the normal fitness sharing method, that is, two additional parameters must be provided or approximated (the niche radius σ_{share} in both decision and objective spaces).

The omni-optimizer algorithm [22] is proposed as a procedure that aims at solving a wide variety of optimization problems (single or multi-objective and uni- or multi-modal problems). The authors argue that, to solve different kinds of problems, it is necessary to know different specialized algorithms. Thus, it is desirable to have an algorithm that adapts itself for handling any number of conflicting objectives, constraints, and variables. The omni-optimizer is important in the context of this work as it uses a two-tier fitness assignment scheme based on the crowding distance of the NSGA-II. The primary fitness is computed using the phenotypes (objectives and constraint values) and the secondary fitness is computed using both phenotypes and genotypes (decision variables). The modified crowding distance computes the average crowding distance of the population in objective and decision spaces. If the crowding value for some individual is above average (at any space), it is assigned the larger of the two distances; otherwise, the smaller of the two distances is assigned. However, we must not lose sight of the fact that omni-optimizer was developed not only to maintain diversity in decision space but with a more general purpose (it adapts itself to solve different kinds of problems).

An algorithm that explicitly promotes the diversity of the decision space is the *MOEA/D with Enhanced Variable-Space Diversity (MOEA/D-EVSD)*, proposed in [23]. This method is an extension of the MOEA/D [10] but with an enhanced variable-space diversity control. In the first generations, the MOEA/D-EVSD tries to induce a larger diversity via promoting the mating of dissimilar individuals. Similarly to MOEA/D, a new individual is created for each subproblem. Then, instead of randomly selecting two individuals of the neighborhood, a pool of α candidate parents is randomly filled from the neighborhood with probability δ , and with probability $1 - \delta$ from the whole population. The two selected parents are the ones in the pool that have the largest distance among them. As the δ parameter is dynamically set, a gradual change between exploration and exploitation can be induced. Additionally, a final phase to further promote intensification is included, which is essentially a traditional MOEA/D coupled with Differential Evolution (DE) operators. For the last generations of MOEA/D-EVSD, the traditional mating selection of MOEA/D is conserved together with the

Rand/1/bin scheme for the DE operators. The authors of this paper show that, by inducing a gradual loss of diversity in the decision space, the state-of-the-art of MOEAs can be improved.

Finally, in [24], the diversity integrating hypervolume-based search algorithm (DIVA) is proposed. Here, the authors proposed a modified hypervolume indicator, which is integrated into an evolutionary algorithm. They employ a so-called diversity function, which fulfills certain requirements such that the modified hypervolume indicator remains compliant with the underlying preference relation.

3. Proposed Framework

State-of-the-art MOEAs that measure the approximation quality of their outcome entirely in objective space work typically well if there is a 1:1 relationship between Pareto set and Pareto front (that is, if, for every $y \in F(P_D)$, there exists exactly one $x \in P_D$ such that $F(x) = y$). That is, if a good finite size approximation of the Pareto front is found by the MOEA (the goodness can be measured, e.g., by any existing performance indicator), the respective finite size approximation of the Pareto set is in many cases also satisfying. This, however, does not hold any more if there is an $m : 1$ relationship between Pareto set and front (i.e., if there are multiple $x_i \in P_D$ such that $F(x_i) = y$ for a $y \in F(P_D)$). If, for instance, there are several connected components of the Pareto set that map to the same part of the Pareto front, a good Pareto front approximation does not imply a good (or at least satisfying) approximation of the Pareto set. To see this, consider the hypothetical bi-objective problem that is shown in Figure 1. The Pareto set of this problem consists of two disjunct connected components that map both to the same Pareto front (that is, every $y \in F(P_D)$ has exactly two pre-images). Figure 2 shows four possible approximations in decision and objective space. As it can be seen, the approximation quality is very high for all sets in objective space, while this is not the case for the Pareto set approximations. Out of them, only the last one is “complete” according to the given discretization. MOEAs that merely measure their outcomes in objective space cannot distinguish between those solutions, and, consequently, the Pareto set approximation is left to chance. MOPs of this kind are termed Type III problems in [25].

To overcome this problem, we propose to perform a density estimator that aims to obtain a good distribution both in objective and decision space. Usually, a classical density estimator groups the population considering only the objective values. Such classification is commonly used to define selection criteria for its elements, giving them certain **reference value** based on its distribution in objective space. According to the design of each algorithm, the individual with either lower or higher reference value is chosen. The idea is to define a relationship between this reference value in objective space and a certain measurement in decision space. In this way, the first grouping phase identifies promising solutions in objective space; meanwhile, the second phase favors solutions with the most different values in the decision space. Our goal is to properly represent the trade-off between these two aspects.

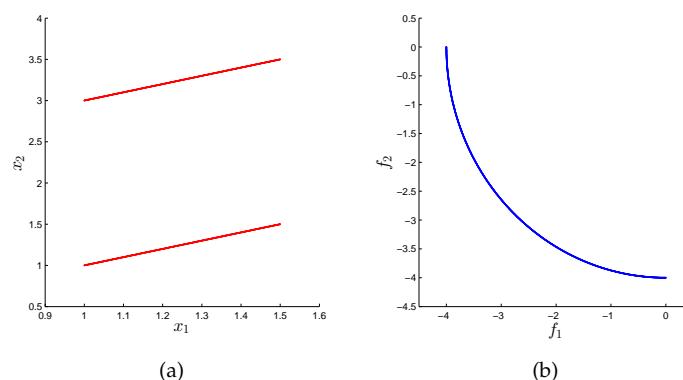


Figure 1. (a) Pareto set and (b) Pareto front of a hypothetical bi-objective problem where the Pareto set consists of two connected components that both map to the entire Pareto front.

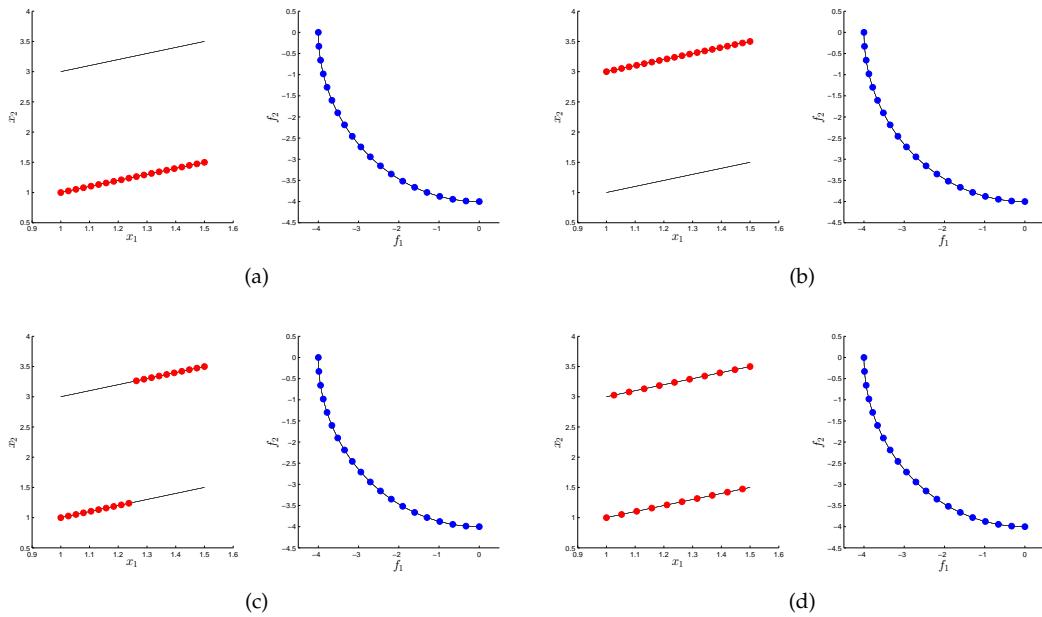


Figure 2. Four different Pareto set/front approximations, where all Pareto front approximations are good (e.g., in the Hausdorff sense), but where only in case (d) the Pareto set approximation is complete.

3.1. Using the Averaged Distance in Variable Space

In the following, we discuss why we think that the usage of the averaged distance is an adequate measure in decision space that serves our purpose.

Let $I = \{x_1, \dots, x_s\} \subset \mathbb{R}$ be a finite set, then the averaged distance \bar{d} between each element $x_i \in I$ and the rest of the elements in I is given by

$$\bar{d}(x_i, I) = \frac{1}{s-1} \sum_{\substack{j=1 \\ j \neq i}}^s d(x_i, x_j), \quad (2)$$

where $d(x_i, x_j)$ is the desired metric for the distance between the two elements x_i and x_j in the decision space that can vary according to the codification or the used norm. In this work, we consider the Euclidean distance, i.e., $d(x_i, x_j) = \|x_i - x_j\|_2$.

Though the averaged distance is defined for every finite set $I \in \mathbb{R}^n$, we will apply it on sets where the values of their images, $F(x_i)$, $i = 1, \dots, s$, are close to each other.

As an illustrative example, consider again the Type III bi-objective problem whose Pareto set and front are shown in Figure 3. The set I is given by the three points \blacktriangledown , \blacklozenge , and \star . All three images are relatively close to the given reference point Z point Z in objective space; assume, for simplicity, that the distances of all three images to Z are given by one (i.e., $\|Z - F(\blacktriangledown)\| = \|Z - F(\blacklozenge)\| = \|Z - F(\star)\| = 1$). Furthermore, we assume that, for the distances in variable space, we have $d(\blacklozenge, \star) = 0.4$, $d(\blacklozenge, \blacktriangledown) = 2$, and $d(\blacktriangledown, \star) = 2.2$. Then, we obtain

$$\bar{d}(\star, I) = \frac{2.6}{2} = 1.3, \quad \bar{d}(\blacklozenge, I) = \frac{2.4}{2} = 1.2, \quad \text{and} \quad \bar{d}(\blacktriangledown, I) = \frac{4.2}{2} = 2.1.$$

Notice that the point with the *biggest* average distance in variable space is also the most different individual in this space. In other words, elements with the maximum average distance in decision space have the desired behavior for Type III problems.

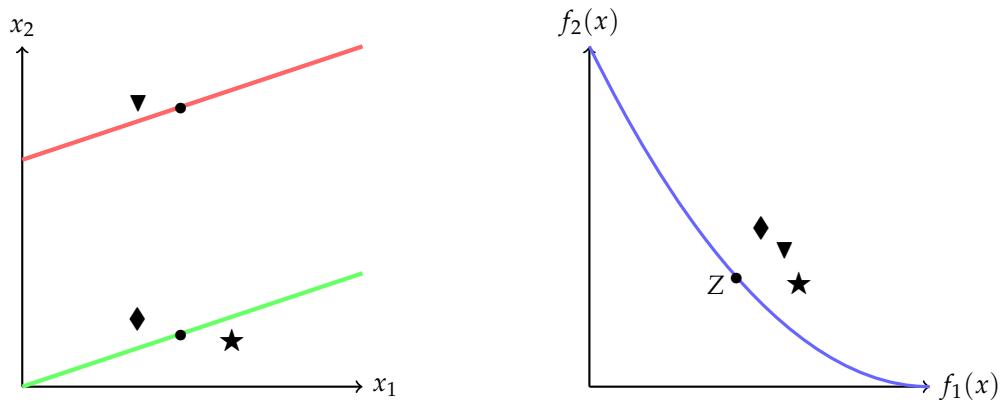


Figure 3. Illustrative example.

However, it is not sufficient to only take into account the average distance of the variables as a selection criterion. Our problem now is how to select an individual that has both a good quality in objective space as well as a good distribution in variable space. We discuss this issue in the following.

3.2. Variation Rate

As explained before, the generic selection criterion of most MOEAs prefers individuals with “the best” reference value in objective space, and this could be a maximum or a minimum value according to the selection procedure. For instance, the selection criterion of the NSGA-II prefers individuals with the biggest crowding distance, while the niching procedure of the NSGA-III favors individuals with the least distance to an induced line. On our part, we have to consider both the reference value provided by the classical selection criterion, as well as the average distance in variables to solve Type III problems.

We first consider selection mechanisms that prefer small reference values. For this, let I be a set of points in decision space whose images are close to each other, let v_i be the reference value in objective space for each $x_i \in I$ and let $\bar{d}(x_i, I)$ as defined as in (2) (in decision space). Then, the variation rate r_i for each element x_i is stated as follows:

$$r_i = \frac{v_i}{\bar{d}(x_i, I)}. \quad (3)$$

This makes sense because for Type III problems the elements of the neighborhood I will have a similar reference value in objectives while the average distances will be larger for the most different solutions in decision space. Hence, its quotient (the variation rate) will tend to be smaller than for the rest of the quotients. Thus, through the variation rate, we have a way to relate the objective and the decision spaces in order to choose the best individual in each group.

Next, we address selection mechanisms that prefer large reference values. For this, we have two options. The first one is to edit the selection criterion to prefer small values in order to use the variation rate. The second alternative is to use a product instead of a quotient. We decide to conserve the essence of each MOEA; for this reason, we implement the second option in this work, which we call *the inverse variation rate*. More precisely, for a set I as above with reference values v_i for each $x_i \in I$ and $\bar{d}(x_i, I)$ as defined in (2), the inverse variation rate \tilde{r}_i is defined as follows:

$$\tilde{r}_i = v_i \cdot \bar{d}(x_i, I). \quad (4)$$

Using this definition, elements with the largest values are hence preferred.

In order to illustrate these two approaches, we go back to the example shown in Figure 3 where we already have the values of \bar{d} and v_i for all the three elements of the set I . Suppose that we have to select two out of the three individuals \star , \blacktriangledown , and \blacklozenge .

If we work with an MOEA that has a selection criterion that prefers individuals with the least distance to Z in objective space, then the only option we have is randomly choosing them (since $v_{\blacktriangledown} = \|Z - F(\blacktriangledown)\| = 1$, $v_{\blacklozenge} = \|Z - F(\blacklozenge)\| = 1$, and $v_{\star} = \|Z - F(\star)\| = 1$). However, if we use the variation rate, then the values change to:

$$r_{\star} = \frac{1}{1.3} \approx 0.7692 \quad r_{\blacklozenge} = \frac{1}{1.2} \approx 0.8333, \quad \text{and} \quad r_{\blacktriangledown} = \frac{1}{2.1} \approx 0.4762.$$

In this way, we select \blacktriangledown as the preferred solution and the second one is \star , which preserves individuals in both of the disconnected regions in decision space.

The desired two-element population is hence given by

$$P = \{\blacktriangledown, \star\}.$$

Otherwise, if we work with an MOEA that has a selection criterion that prefers individuals with the largest distance to Z in objective space (maybe in order to preserve diversity), then the option we have again is randomly choosing two of them. However, if we use the inverse variation rate, that is:

$$\tilde{r}_{\star} = 1 \cdot 1.3 = 1.3, \quad \tilde{r}_{\blacklozenge} = 1 \cdot 1.2 = 1.2, \quad \text{and} \quad \tilde{r}_{\blacktriangledown} = 1 \cdot 2.1 = 2.1,$$

then it also leads \blacktriangledown and \star as the selected individuals.

The desired two-element population is again given by

$$P = \{\blacktriangledown, \star\}.$$

Observe that, in both cases, we conserve one individual in each disconnected component of the Pareto set. This is something that we can not guarantee with the use of a standard approach.

Notice that, for all MOPs with a 1:1 relationship of the Pareto set and the Pareto front, it is expected that solutions in the same neighborhood have similar reference values in objective space and also a similar average distance in decision space. Thus, it is also likely that making the quotient or the product of these values does not significantly affect the original selection criterion. This will be shown in Section 4 on several classical benchmark problems.

We can now state a general framework. A pseudocode of the Variation Rate is shown in Algorithm 1.

Algorithm 1 Framework to include the Average Distance in Variables within any MOEA

Require: Parameters of the selected MOEA

Ensure: Final population P_t

- 1: $t \leftarrow 0$
 - 2: $P_0 \leftarrow \text{InitializePopulation}()$
 - 3: **while** the stop criterion is not satisfied **do**
 - 4: $M_t \leftarrow \text{VariationOperator}(P_t)$
 - 5: $V_t \leftarrow \text{SelectProcedure}(P_t \cup M_t)$
 - 6: $P_t \leftarrow \text{SelectByVariationRate}(V_t)$
 - 7: $t \leftarrow t + 1$
 - 8: **end while**
-

In Algorithm 1, the procedure `SelectByVariationRate` takes the reference values V_t in objective space provided by `SelectProcedure` (we assume it is based on a classical selection criterion), and then it updates such values according to the variation rate or the inverse variation rate in order to improve the selection mechanism to deal with Type III problems.

As we can see, this framework can be used in principle within any MOEA; however, the particular use of the variation rate or the inverse variation rate will depend on the given MOEA. In the following, we explain how to adapt the Variation Rate for four of the most representative MOEAs. The reader can find the pseudocode of these four variants in Appendix A.

3.3. Integration into NSGA-II

The first algorithm that we consider is the classical NSGA-II, which has been used successfully for the treatment of a large number of applications. This is a domination-based multi-objective evolutionary algorithm; that is, this method directly applies the Pareto dominance relation and an elitism strategy to preserve the best individuals along the optimization process. The elitism operator is incorporated via a special parent selection based on two mechanisms: **fast-non-dominated-sorting** and **crowding distance**. The former conserves best individuals based on the Pareto dominance relation, whereas the latter is used to promote the preservation of the diversity.

We consider the classification in fronts performed by the fast-non-dominated-sorting as our neighborhood structure because the crowding distance is applied only in the last front that can contribute elements to the next population. This means that we have to integrate the diversity into decision space into the crowding distance. The crowding distance procedure sorts the elements in the last front according to the values of objectives; then, the crowding distance of an individual p_i is the average distance in objective space from the previous and the next individuals (according to the induced order), that is, the individuals p_{i-1} and the p_{i+1} . In order to preserve the extreme individuals, the crowding distance of the first and the last element is set as a big value. This means that the crowding distance prefers elements with big values, and, hence, we use the inverse variation rate.

The pseudocode of the modification of the NSGA-II with the variation rate (VR-NSGA-II) is shown in Algorithm A1 of Appendix A.1.

3.4. Integration into NSGA-III

We consider this algorithm here because it is able to properly deal with MOPs with many objectives. This algorithm is similar to its predecessor, the NSGA-II in the variation operators and in the classification of the fronts via the fast-non-dominated-sorting; however, the crowding distance is replaced by a more sophisticated procedure.

Here, the idea is to take advantage of the **association** method of the NSGA-III, which defines a “neighborhood” structure in a very convenient way for our purpose. The association method assigns each element of F_j (the last front classified after the fast-non-dominated-sorting) to the nearest induced line by some weight $w_i \in Z$, where Z is a set of reference points. Each weight can have more than one associated element, forming a neighborhood.

In the original NSGA-III, the **niching** is realized by sorting the obtained groups in the association stage according to its cardinality in ascending order. The element with the least distance to the induced line in each group is selected, and the algorithm continues with the next group until the population is filled. Thus, we modify the **niching** method. To include the diversity in decision space, our new niching procedure does not prefer the element with the least distance value. Instead, it prefers the one with the smallest *variation rate*.

The pseudocode of an iteration of the VR-NSGA-III algorithm with variation rate is shown in Algorithm A2 of Appendix A.2.

3.5. Integration into MOEA/D

MOEA/D is part of the Decomposition-Based Evolutionary Algorithms, which transform the original multi-objective optimization problem into a set of single-objective optimization problems that are simultaneously solved. In particular, this method takes a set of weights to define neighborhoods. The set of nearest weights defines one neighborhood and the best individuals are selected based on the value of a certain aggregative function. MOEA/D considers the weighted aggregation of

objectives as an elitism mechanism. Furthermore, the neighborhood structure promotes the mating of close solutions. Different aggregative functions can be used in the MOEA/D framework. However, individuals with the least values are selected. In this work, we employ the Tchebycheff function, which is the most popular approach.

In order to include variation rate to the MOEA/D, we modify the selection criterion. Instead of preferring individuals with the least aggregative function value, we use the least value of the variation rate. For this, we employ the neighborhood structure of the original MOEA/D.

The pseudocode of VR-MOEA/D is shown in Algorithm A3 of Appendix A.3.

3.6. Integration into SMS-EMOA

SMS-EMOA is an indicator based algorithm; this means that it uses as a selection criterion the value of a certain performance indicator. In the case of SMS-EMOA, it is the hypervolume indicator.

This algorithm is similar to the NSGA-II, but it replaces the crowding distance by the contribution to the hypervolume of each individual in the last front. That is, the individuals of the last front with the biggest contribution to the hypervolume are preferred.

In this case, to adapt the SMS-EMOA, we consider the inverse variation rate, as the original mechanism criterion of this algorithm prefers high values. Again, we use the last front as our neighborhood structure.

The pseudocode of this method is shown in Algorithm A4 of Appendix A.4.

4. Numerical Results

In this section, we show some numerical results and comparisons to the state-of-the-art to demonstrate the benefit and strength of the variation rate. To this end, we first compare the original version of each algorithm against its corresponding version that uses the variation rate (respectively, the inverse variation rate) on some widely used (non-Type III) benchmark problems. This is done in order to show that the performance of each algorithm is not significantly affected for standard problems. In the next step, we test again the original and variation rate versions of the selected MOEAs on some Type III problems, where the advantage of the variation rate becomes apparent.

The benchmark problems that we use for the first part of these experiments are the well known test problems DTLZ 1-4 [26], IDTLZ 1 and IDTLZ2 [27], as well as the test problems WFG 1-5 [28]. For the second part, we use following six Type III problems.

The first Type III problem is taken from [22], which is defined as follows:

$$f_1(x) = \sum_{i=1}^n \sin(\pi x_i), \quad f_2(x) = \sum_{i=1}^n \cos(\pi x_i), \quad (5)$$

where $0 \leq x_i \leq 6$. This problem, denoted as OMNI1 in this work, has a total of 243 different disconnected components that form the Pareto set, and all of these components map to the same Pareto front.

The second problem, also taken from [22], is defined as follows:

$$f_1(x) = \sin\left(\pi \sum_{i=1}^n x_i\right), \quad f_2(x) = \cos\left(\pi \sum_{i=1}^n x_i\right), \quad (6)$$

where $0 \leq x_i \leq 1$, and $i = 1, 2, \dots, 6$. This problem is denoted as OMNI2 in this work. Let $y = \sum_{i=1}^6 x_i$ be the sum of the variables, then the Pareto set consists of the points x where $1 \leq y \leq 1.5$ or $3 \leq y \leq 3.5$. In addition, here, both connected components map to the same Pareto front. That is, every point on the Pareto front can be obtained in different infinite ways via the combinations of the variables mentioned.

The third Type III problem is the application stated in [29,30], where subdivision techniques have been used to tackle the problem. It is stated as follows: for $f_1, f_2 : \mathbb{R}^5 \rightarrow \mathbb{R}$, it is:

$$\begin{aligned} f_1(x) &= \sum_{i=1}^n x_i, \\ f_2(x) &= 1 - \prod_{i=1}^n (1 - w_i(x_i)), \end{aligned} \quad (7)$$

where

$$w_i(z) = \begin{cases} 0.01 \cdot \exp\left(-\left(\frac{z}{20}\right)^{2.5}\right), & \text{for } i = 1, 2, \\ 0.01 \cdot \exp\left(-\frac{z}{15}\right), & \text{for } 3 \leq i \leq 5. \end{cases} \quad (8)$$

Finally, we consider the methodology from [25] to construct three more problems, denoted in this paper as RPH1, RPH2, and RPH3. These are bi-objective problems with two variables. In order to properly define them, we use the following functions.

First, we define the objective functions for the RPH1-3 problems

$$\begin{aligned} f_1(x) &= (x_1 + a)^2 + x_2^2, \\ f_2(x) &= (x_1 - a)^2 + x_2^2, \end{aligned} \quad (9)$$

where $x \in \mathbb{R}^2$ and $a \in \mathbb{R}^+$. The variants of the RPH problems are obtained with the following functions.

Let $t_1(x)$ and $t_2(x)$, with $x \in \mathbb{R}^2$, be the tile identifiers that are determined via:

$$\begin{aligned} \hat{t}_1(x) &= \operatorname{sgn}(x_1) \cdot \left\lceil \frac{|x_1| - (a + \frac{c}{2})}{2a + c} \right\rceil, \\ \hat{t}_2(x) &= \operatorname{sgn}(x_2) \cdot \left\lceil \frac{|x_2| - \frac{b}{2}}{b} \right\rceil, \end{aligned} \quad (10)$$

which restrict the problem to nine tiles using the relation $t_i = \operatorname{sgn}(\hat{t}_i(x)) \cdot \min\{|\hat{t}_i|, 1\}$, with $i = 1, 2$.

Then, RPH1 is defined as $f_i^{(1)}(x) = f(\hat{x}(x))$, where $\hat{x} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is defined by the following transformation:

$$\begin{aligned} \hat{x}_1(x_1) &= x_1 - t_1 \cdot (c + 2a), \\ \hat{x}_2(x_2) &= x_2 - t_2 \cdot b. \end{aligned} \quad (11)$$

For the RPH1-3 problems, we fix the constant values $a = 4$, $b = 10$, and $c = 4$.

The RPH2 problem is defined as the RPH1, but it rotates the variables. That is, for an angle θ , we have

$$r(x) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} x, \quad (12)$$

and then $f_i^{(2)}(x) = f_i^{(1)}(r(x))$. In this paper, we use $\theta = \frac{\pi}{4}$.

Finally, via the following transformation $d : \mathbb{R}^2 \rightarrow \mathbb{R}$

$$d(x) = x_1 \cdot \left(\frac{x_2 - L + \epsilon}{U - L} \right), \quad (13)$$

for some small $\epsilon > 0$ and where U and L denote the upper and lower bound of the search space, respectively; we can define the RPH3 as: $f_i^{(3)}(x) = f_i^{(2)}(d(x), x_2)$, which is a rotated and transformed problem. In this paper, we use $\epsilon = 0.1$ for the RPH3 problem, while $L = -20$ and $U = 20$ are the the upper and lower bounds of each variable for the RPH1-3 problems.

We use the PlatEMO platform [31] to make our test. The parameter settings of all the used algorithms are shown in Table 1. For all experiments, we have executed 30 independent runs. The numerical results with the mean and standard deviation of the hypervolume and Δ_p indicators are shown in Tables 2 and 3. In these tables, we have put in bold the best value between each pair of algorithms (the original version and the version with variation rate). We also performed the Wilcoxon test [32] as statistical significance proof to validate the results. For this, we consider the value $\alpha = 0.05$. We put in gray the cell where such difference has statistical significance according to this test.

Table 1. Parameter configuration for each algorithm. Mutation probability m_p , crossover probability c_p , neighborhood size T , and number of reference points #Z.

Parameter	NSGA-II	NSGA-III	MOEA/D	SMS-EMOA
m_p	$1/n$	$1/n$	0.1	$1/n$
c_p	0.8	0.8	1.0	0.8
T	-	-	20	-
#Z	-	200	-	-

From the tables, we obtain that, for the classical benchmark problems, the original version of the selected MOEAs has a better Δ_p value than the variation rate version in 27 out of the 44 combinations, where only 19 out of these values have statistical significance, which is an expected result. However, it is important to notice that the variation rate versions do not always lose, according to the Δ_p indicator values, the variation rate versions are better in 17 out of 44 cases, but only in two with statistical significance.

For the hypervolume indicator, something similar happens. Here, the original version of the MOEAs is better than the variation rate version in 33 out of 44 runs, with statistical significance in 21 cases, while the variation rate version wins in 11 out of 44 cases for this indicator, where three of them have statistical significance.

In total, from the 88 possible combinations (algorithms, indicators and problems), we have statistical significance in 45 cases; this means that, almost 50% of the time, it is not possible to say that the original version is different than the variation rate version. Moreover, in the cases when we have statistical significance, we can see in Table 2 that the averaged values are very similar.

We observe the advantages of the variation rate versions with the Type III problems (see Table 3). For these problems, we use the Δ_p both in objective and decision space (we denote this in the table as Obj. Δ_p and Var. Δ_p , respectively). We observe a similar behavior in objective space, here 16 out of 24 possible combinations are better for the original MOEAs (but only 6 out of these 34 have statistical significance). However, in decision space, the variation rate versions are better than the original versions; we have that 18 out of 24 combinations have better Δ_p values, almost all of them with statistical significance (only, in one case, we can not reject the null hypothesis).

Graphical results are shown in Figures 4–9; we plot the original MOEA and its corresponding variation rate version with the best value for each problem (according to the median of all runs using the Var. Δ_p).

Table 2. Numerical results for the original and variation rate version of some MOEAs in standard benchmark test problems. We show the mean and standard deviation (up and down in the cell, respectively). We put in bold the best value and in gray the cells with statistical significance according to the Wilcoxon test.

Problem	Ind.	NSGA-II		NSGA-III		MOEAD		SMSEMOA	
		Original	VR	Original	VR	Original	VR	Original	VR
DTLZ1	Δ_p	0.0155 0.0004	0.0181 0.0006	0.0105 0.0000	0.0105 0.0000	0.0105 0.0000	0.0105 0.0000	0.0358 0.0639	0.0719 0.1213
	HV	0.8473 0.0013	0.8454 0.0012	0.8576 0.0003	0.8575 0.0006	0.8571 0.0005	0.8572 0.0004	0.8359 0.0081	0.8351 0.0079
DTLZ2	Δ_p	0.0410 0.0008	0.0410 0.0014	0.0277 0.0000	0.0277 0.0000	0.0277 0.0000	0.0280 0.0000	0.0539 0.0041	0.0536 0.0041
	HV	0.5630 0.0018	0.5641 0.0015	0.5813 0.0000	0.5813 0.0001	0.5814 0.0000	0.5793 0.0001	0.5593 0.0013	0.5588 0.0020
DTLZ3	Δ_p	0.0666 0.0780	0.0572 0.0855	0.0301 0.0021	0.0295 0.0015	0.0316 0.0027	0.0308 0.0019	0.0687 0.0522	0.1386 0.2410
	HV	0.5577 0.0052	0.5598 0.0074	0.5714 0.0060	0.5730 0.0044	0.5660 0.0068	0.5677 0.0053	0.5526 0.0066	0.5456 0.0213
DTLZ4	Δ_p	0.0400 0.0007	0.0450 0.0015	0.0277 0.0000	0.0277 0.0000	0.1359 0.2069	0.0293 0.0000	0.0719 0.1096	0.0725 0.1095
	HV	0.5650 0.0013	0.5676 0.0013	0.5812 0.0001	0.5813 0.0001	0.5287 0.0971	0.5721 0.0002	0.5521 0.0483	0.5522 0.0481
IDTLZ1	Δ_p	0.0156 0.0005	0.0187 0.0011	0.1542 0.2459	0.0774 0.1318	0.0177 0.0000	0.0181 0.0001	0.0371 0.0708	0.0422 0.0984
	HV	0.2301 0.0017	0.2268 0.0018	0.2331 0.0007	0.2326 0.0006	0.2287 0.0001	0.2275 0.0002	0.2218 0.0035	0.2217 0.0038
IDTLZ2	Δ_p	0.0394 0.0008	0.0406 0.0012	0.0412 0.0008	0.0412 0.0010	0.0475 0.0008	0.0456 0.0007	0.0513 0.0035	0.0510 0.0048
	HV	0.5495 0.0011	0.5425 0.0019	0.5550 0.0020	0.5533 0.0017	0.5565 0.0001	0.5548 0.0002	0.5410 0.0023	0.5408 0.0018
WFG1	Δ_p	0.1626 0.0224	0.2488 0.0405	0.2143 0.0266	0.2392 0.0342	0.2495 0.0176	0.7113 0.0929	0.3413 0.0353	0.3408 0.0283
	HV	0.9257 0.0095	0.8582 0.0249	0.8740 0.0157	0.8629 0.0198	0.8892 0.0237	0.5933 0.0428	0.8697 0.0170	0.8725 0.0152
WFG2	Δ_p	0.1270 0.0072	0.1329 0.0083	0.0830 0.0012	0.0830 0.0013	0.2166 0.0335	0.2222 0.0320	0.1405 0.0069	0.1377 0.0067
	HV	0.9356 0.0009	0.9334 0.0010	0.9393 0.0006	0.9392 0.0006	0.9203 0.0061	0.9200 0.0049	0.9309 0.0015	0.9307 0.0011
WFG3	Δ_p	0.5261 0.0241	0.7354 0.0234	0.7229 0.0180	0.7200 0.0224	0.8110 0.0261	0.8788 0.0462	0.4541 0.0488	0.4633 0.0653
	HV	0.4131 0.0013	0.4080 0.0023	0.4013 0.0016	0.4006 0.0018	0.3826 0.0163	0.3613 0.0142	0.3672 0.0052	0.3649 0.0033
WFG4	Δ_p	0.1641 0.0044	0.1968 0.0053	0.1142 0.0007	0.1147 0.0007	0.1361 0.0028	0.2240 0.0053	0.2349 0.0099	0.2301 0.0136
	HV	0.5502 0.0026	0.5328 0.0029	0.5751 0.0008	0.5741 0.0008	0.5550 0.0028	0.5180 0.0029	0.5336 0.0026	0.5339 0.0031
WFG5	Δ_p	0.1857 0.0038	0.1927 0.0041	0.1385 0.0004	0.1388 0.0003	0.1515 0.0013	0.1549 0.0014	0.2191 0.0149	0.2181 0.0131
	HV	0.5178 0.0032	0.5148 0.0029	0.5396 0.0003	0.5393 0.0002	0.5225 0.0023	0.5200 0.0029	0.5168 0.0021	0.5166 0.0027

Table 3. Numerical results for the original and variation rate version of some MOEAs in Type III test problems. We show the mean and standard deviation (up and down in the cell, respectively). We put in bold the best value and in gray the cells with statistical significance according to the Wilcoxon test.

Problem Ind.	NSGA-II		NSGA-III		MOEAD		SMSEMOA	
	Original	VR	Original	VR	Original	VR	Original	VR
OMNI1 Obj Δ_p	0.0218 0.0006	0.0220 0.0005	0.0293 0.0030	0.0306 0.0046	0.1605 0.0145	0.2363 0.0547	0.0210 0.0017	0.0206 0.0012
	2.2071 0.2928	1.8804 0.1603	1.8876 0.1852	1.9313 0.2480	4.3767 0.6295	3.2976 0.7931	2.2525 0.2978	2.0409 0.2544
OMNI2 Obj Δ_p	0.0038 0.0001	0.0039 0.0001	0.0048 0.0000	0.0048 0.0000	0.0237 0.0000	0.0278 0.0076	0.0039 0.0001	0.0039 0.0000
	1.1649 0.0360	0.7206 0.0075	0.8714 0.0956	0.7576 0.0312	1.2561 0.1039	0.9800 0.1302	0.9073 0.0853	0.7366 0.0071
SCM1 Obj Δ_p	0.3149 0.0135	0.3138 0.0123	0.3599 0.0198	0.3594 0.0303	81.3852 0.0043	81.3853 0.0042	0.2852 0.0297	0.2681 0.0252
	3.7266 0.1461	3.7751 0.1440	3.5737 0.1586	3.6209 0.1468	38.5910 0.0020	38.5912 0.0022	2.5058 0.0309	2.4572 0.0261
RPH1 Obj Δ_p	0.1636 0.0042	0.1653 0.0040	0.4323 0.0028	0.4334 0.0042	2.6365 0.0027	3.3847 0.7119	0.1551 0.0023	0.1535 0.0026
	6.8061 2.5349	4.2771 0.7180	1.6001 1.3496	1.0450 0.8738	8.0157 2.8617	5.2802 0.7714	2.7075 1.9102	5.3446 0.4255
RPH2 Obj Δ_p	0.1725 0.0027	0.1745 0.0051	0.4302 0.0109	0.4365 0.0141	2.6459 0.0175	3.1017 0.6411	0.1580 0.0038	0.1574 0.0035
	2.6822 0.9276	3.0836 0.7577	1.4661 0.5815	1.2051 0.6150	11.3906 3.0919	6.6796 1.5272	1.2333 0.5085	1.1097 0.5499
RPH3 Obj Δ_p	0.1701 0.0040	0.1732 0.0047	0.4314 0.0182	0.4340 0.0140	2.6572 0.0360	3.1012 0.6169	0.1555 0.0038	0.1564 0.0032
	6.3361 1.7299	2.8003 0.8529	3.0233 1.9181	2.1521 0.9408	8.6873 2.0764	4.7500 1.5926	2.8545 1.1084	1.6417 0.5711

In Figure 4, we can see that the obtained distribution is better for the variation rate version; it looks similar to that obtained by the original algorithm. However, we have to recall that this problem has 243 different disconnected components in the Pareto set, and some of them are overlapping in the plot. In Figure 5, we notice that the variation rate version can obtain points at the two regions in this representation (we plot y on both axes, where $y = \sum_{i=1}^6 x_i$), while the original version is only able to compute points in one of them. On the other hand, in Figure 6, we can see that both the original and the variation rate versions can approximate the disconnected components of this problem well (except by the MOEA/D algorithm). However, the variation rate version is better in this problem, according to the values of Table 3.

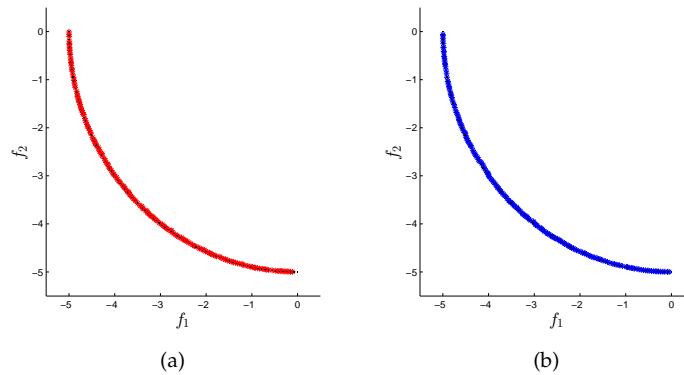


Figure 4. Cont.

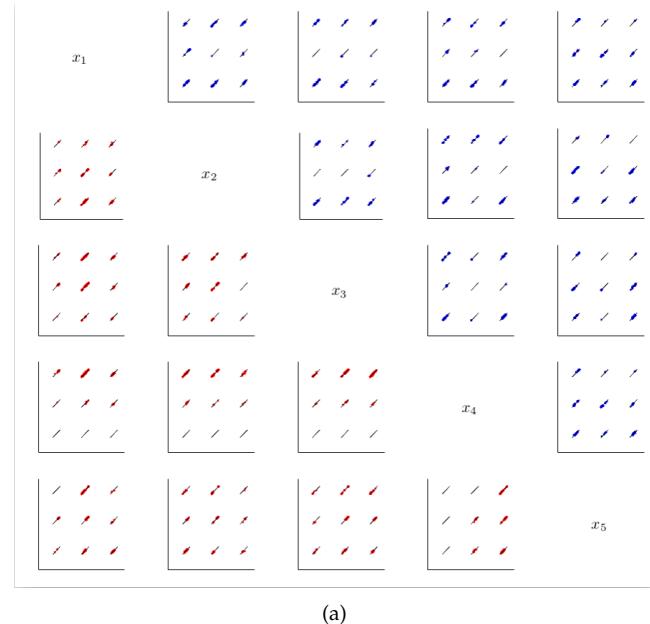


Figure 4. Graphical results of the run with the median values for the OMNI1 function for NSGA-II. (a) Objective Space Original; (b) Objective Space VR; (c) Decision Space, pairwise plot of each variable. The left-down and red marks correspond to the original algorithm, while the right-up and blue ones are the VR version.

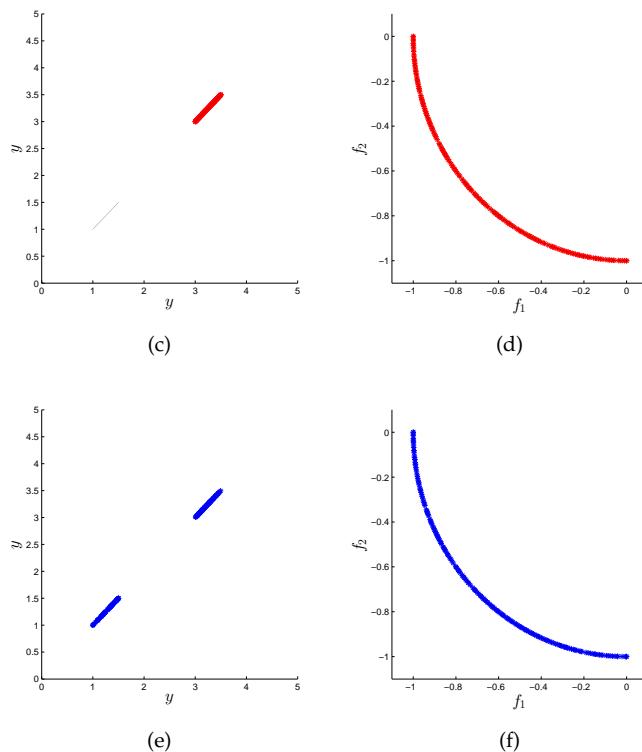


Figure 5. Graphical results of the run with the median values for the OMNI2 function for the NSGA-II algorithm. (a) Decision Space Original; (b) Objective Space Original; (c) Decision Space VR; (d) Objective Space VR.

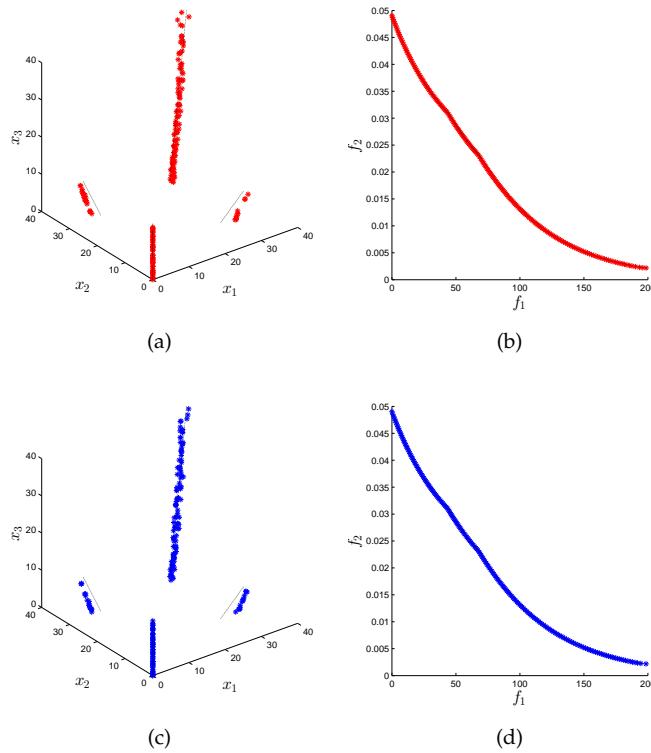


Figure 6. Graphical results of the run with the median values for the SCM1 function for the SMS-EMOA algorithm. (a) Decision Space Original; (b) Objective Space Original; (c) Decision Space VR; (d) Objective Space VR.

For the RPH problems, we can see in Figure 7 that the variation rate version of the NSGA-II can obtain four out of the nine disconnected components of the Pareto set, while the original version only gets three out of them; moreover, the distribution in decision space is also improved. In Figure 8, we can see again a similar behavior between the variation rate and the original version of the NSGA-III algorithm, which is also confirmed by the values of Table 3, where the variation rate version wins in three out of the four baseline algorithms for the RPH2 problem, but only in one does it have statistical significance. Finally, for the RPH3 problem, we can see in Figure 9 how the variation rate can significantly improve the performance of the MOEA/D algorithm in its distribution in decision space, as the original version only obtains points in one out of the nine disconnected components of the Pareto set, while the variation rate version obtains five out of them.

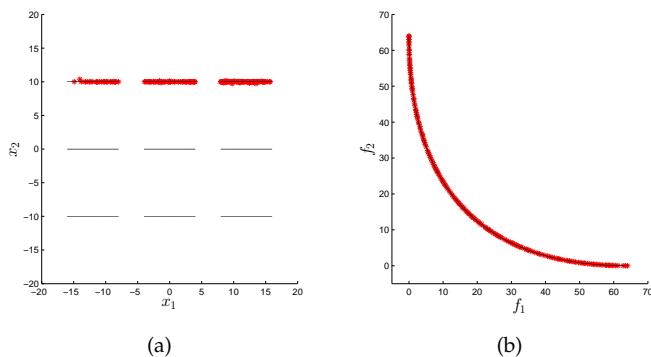


Figure 7. Cont.

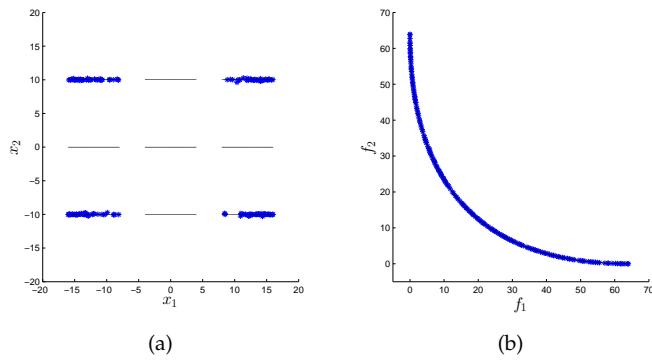


Figure 7. Graphical results of the run with the median values for the RPH1 function for the NSGA-II algorithm. (a) Decision Space Original; (b) Objective Space Original; (c) Decision Space VR; (d) Objective Space VR.

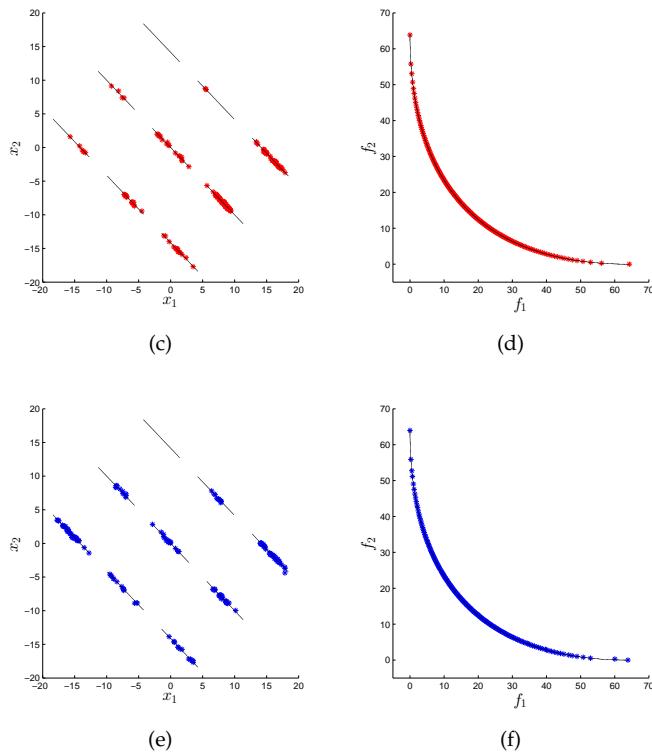


Figure 8. Graphical results of the run with the median values for the RPH2 function for the NSGA-III algorithm. (a) Decision Space Original; (b) Objective Space Original; (c) Decision Space VR; (d) Objective Space VR.

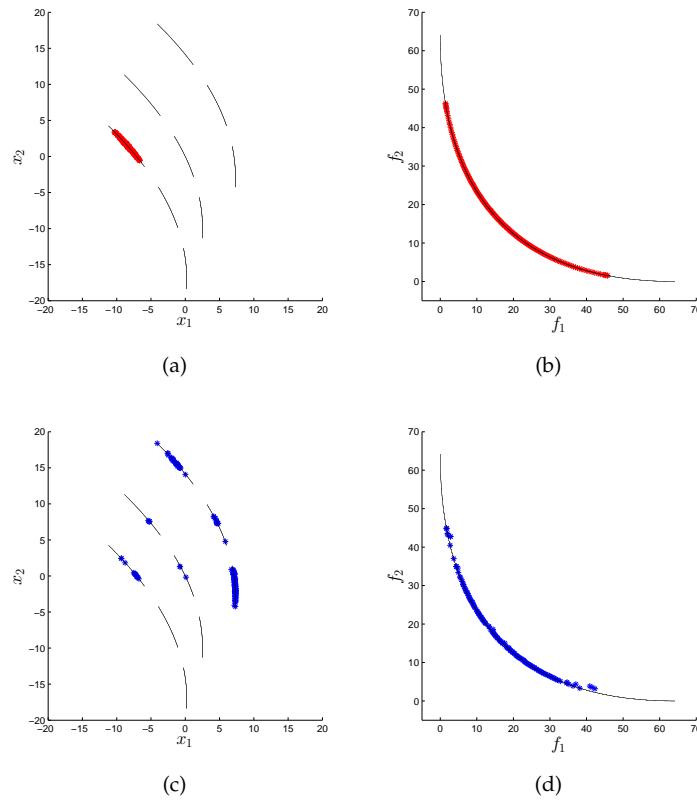


Figure 9. Graphical results of the run with the median values for the RPH3 function for the MOEA/D algorithm. (a) Decision Space Original; (b) Objective Space Original; (c) Decision Space VR; (d) Objective Space VR.

5. Conclusions and Future Work

In this work, we have addressed the problem of computing diverse solutions both in decision and objective space for a given multi-objective optimization problem via specialized evolutionary strategies. While so far quite a few good diversity mechanisms exist to obtain a spread in objective space, the consideration of the Pareto set approximations has been mainly neglected so far. This represents a possible shortcoming in particular for Type III problems where points in the Pareto front may have multiple pre-images. To achieve this goal, we have first presented the general framework of the variation rate that combines the usage of the averaged distance in variable space with the selection operator that is given by the multi-objective evolutionary algorithm (MOEAs). We have demonstrated further on possible integration of the variation rate into four MOEAs that represent the state-of-the-art.

Numerical results have shown that the use of the variation rate improves the performance of the standalone algorithms for Type III problems, while the variation rate algorithms are not significantly worse for the standard benchmark problems (i.e., problems with a 1:1 relationship between Pareto set and Pareto front), even in some cases variation rate improves the performance of the original algorithm. Of course, the behavior of variation rate by itself is not enough for the treatment of every kind of MOP as this depends on the operators of every algorithm. That is, the variation rate can enhance the overall performance of a certain algorithm, but if, for instance, such algorithm is not conceived to deal with MOPs with many objectives, the addition of the variation rate would be not enough to solve them. We can, for instance, consider some of the algorithms and test functions of [33], in order to show if the variation rate provides some advantages when dealing with MOPs with many objectives.

As future work, it will be mandatory to adapt the genetic operators of the evolutionary algorithms to exploit the diversity in decision space. As the above results have shown, the diversity in decision space has already increased significantly; however, the variation rate is only a selection mechanism. In order to obtain

optimal solutions in particular in decision space, the exploration will have to be increased. Furthermore, it will be necessary to develop a particular indicator for problems of Type III. In general, performance indicators evaluate an approximation based on the value of the objectives, but, for problems such as OMNI1, this does not provide enough information. Once we have such an indicator, we can validate better performance of our methods in decision space. However, it is ad hoc not clear which property has to be satisfied with this approximation. We also need to test this approach in problems with different properties in decision space, in particular problems with disconnected Pareto sets.

Author Contributions: O.C. and O.S. conceived and designed the experiments; O.C. performed the experiments; O.C. and O.S. analyzed the data; O.C. and O.S. wrote the paper.

Funding: The authors acknowledge funding from the Conacyt Basic Science project No. 285599 and SEP Cinvestav project No. 231.

Acknowledgments: The first author acknowledges support from the Conacyt to pursue his Ph.D. studies.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Pseudo-Codes

The appendix contains the pseudocode of all the variation rate algorithms that were used in this work.

Appendix A.1. VR-NSGA-II

The differences between the original NSGA-II and the VR-NSGA-II are lines 12 and 20. In line 12, we compute the inverse variation rate values of the last front, while, on line 20, we perform the selection according to these values.

Algorithm A1 Pseudocode of VR-NSGA-II

Require: Population size (P_s), crossover probability (P_c), mutation probability (P_m)

Ensure: Final Population

```

1: Population ← InitializePopulation( $P_s$ )
2: FastNondominatedSorting(Population)
3: Selected ← SelectParentsByRank(Population,  $P_s$ )
4: Children ← CrossoverAndMutation(Selected,  $P_c$ ,  $P_m$ )
5: while StopCondition() do
6:   Union ← Merge(Population, Children)
7:   Fronts ← FastNondominatedSorting(Union)
8:   Parents ←  $\emptyset$ 
9:    $Front_L \leftarrow \emptyset$ 
10:  for  $Front_i \in \text{Fronts}$  do
11:     $V_t \leftarrow \text{CrowdingDistanceAssignment}(Front_i)$ 
12:     $R_t \leftarrow \text{InverseVariationRateAssignment}(V_t)$ 
13:    if Size(Parents)+Size( $Front_i$ ) >  $P_s$  then
14:       $Front_L \leftarrow i$ 
15:      break
16:    else
17:      Parents ← Merge(Parents,  $Front_i$ )
18:    end if
19:    if Size(Parents) <  $P_s$  then
20:       $Front_L \leftarrow \text{SortByRankAndInverseVariationRate}(Front_L)$ 
21:      for  $P_1$  to  $P_{P_s-\text{Size}(Front_L)}$  do
22:        Parents ←  $P_i$ 
23:      end for
24:    end if
25:    Selected ← SelectParentsByRankAndDistance(Parents,  $P_s$ )
26:    Population ← Children
27:    Children ← CrossoverAndMutation(Selected,  $P_c$ ,  $P_m$ )
28:  end for
29: end while
30: return Children

```

Appendix A.2. VR-NSGA-III

We only show the iteration as the complete code is basically identical to NSGA-II with a different selection mechanism. Here, the main difference of the variation rate version, compared to the original NSGA-III, is on line 14; while the original NSGA-III uses the niching as selection criterion, the VR-NSGA-III employs the variation rate, as it is stated on line 14.

Algorithm A2 Iteration of the VR-NSGA-III

Require: Reference points Z , current population P_t

Ensure: Next, population P_{t+1}

- 1: $S_t = \emptyset$, $i = 1$
 - 2: Q_t = apply variation operators to P_t
 - 3: $M_t = P_t \cup Q_t$
 - 4: (F_1, F_2, \dots) = fast-non-dominated-sorting(M_t)
 - 5: **while** $|S_t| \leq N$ **do**
 - 6: $t = S_t \cup F_i$
 - 7: $i = i + 1$
 - 8: **end while**
 - 9: Add first fronts to P_{t+1}
 - 10: F_i := last added front
 - 11: Normalize F_i
 - 12: Associate elements of F_i with each Z
 - 13: V_t := Niching of F_i
 - 14: R_t := SelectByVariationRate(V_t)
 - 15: $P_{t+1} : S_t \cup R_t$
-

Appendix A.3. VR-MOEA/D

The change in this algorithm, concerning the original MOEAD, is very subtle. On lines 9 and 10, we compute the variation rate of the elements of the neighborhood and the offspring ($B(i) \cup y$) instead of only computing the values of the aggregative function.

Algorithm A3 Pseudocode of VR-MOEA/D

Require: N number of solutions and weight vectors; T neighborhood size.

Ensure: Final Population

- 1: Initialize N weight vectors $\lambda^1, \lambda^2, \dots, \lambda^N$
 - 2: Set N subproblems defined by the N weight vectors
 - 3: Set N neighborhoods $B(i) = \{w_{i,1}, \dots, w_{i,T}\}$, where $w_{i,j} = \lambda^j$ are the closest weight vectors to λ^i
 - 4: $\{x^1, \dots, x^N\} \leftarrow \text{InitializePopulation}(N)$
 - 5: **while** StopCondition() **do**
 - 6: **for** $i \in N$ **do**
 - 7: Randomly select two solutions from $B(i)$ to generate an offspring y
 - 8: Apply using variation operators to y
 - 9: Compute the values V_r of $B(i) \cup y$ via the aggregative function g .
 - 10: Compute Variation Rate V_t of the elements of $B(i) \cup y$
 - 11: **for** x in $B(i)$ **do**
 - 12: **if** $r_y < r_x$ **then**
 - 13: Replace x with y
 - 14: **end if**
 - 15: **end for**
 - 16: **end for**
 - 17: **end while**
 - 18: **return** Children
-

Appendix A.4. VR-MOEA/D

Again, in this case, we only show an iteration of the method as the rest of the algorithm is basically the NSGA-II. We observe that, on line 12, we substitute the values of the hypervolume contributions with the variation rate and we use them for the selection mechanism.

Algorithm A4 Iteration of the VR-SMS-EMOA

Require: Current population P_t
Ensure: Next, population P_{t+1}

- 1: $S_t = \emptyset, i = 1$
- 2: $Q_t = \text{apply variation operators to } P_t$
- 3: $M_t = P_t \cup Q_t$
- 4: $(F_1, F_2, \dots) = \text{fast-non-dominated-sorting}(M_t)$
- 5: **while** $|S_t| \leq N$ **do**
- 6: $t = S_t \cup F_i$
- 7: $i = i + 1$
- 8: **end while**
- 9: Add first fronts to P_{t+1}
- 10: $F_i := \text{last added front}$
- 11: $V_t \leftarrow \text{ComputeHypervolumeContributions}(F_i)$
- 12: $R_t := \text{SelectByVariationRate}(V_t)$
- 13: $P_{t+1} : S_t \cup R_t$

References

1. Coello, C.A.C.; Lamont, G.B. *Applications of Multi-Objective Evolutionary Algorithms*; World Scientific: Singapore, 2004; Volume 1.
2. Lewis, A.; Mostaghim, S.; Randall, M. *Biologically-Inspired Optimisation Methods: Parallel Algorithms, Systems and Applications*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 210.
3. Deb, K. *Multi-Objective Optimization Using Evolutionary Algorithms*; John Wiley & Sons: Chichester, UK, 2001.
4. Coello, C.A.C.; Lamont, G.B.; Van Veldhuizen, D.A. *Evolutionary Algorithms for Solving Multi-Objective Problems*; Springer: New York, NY, USA, 2007; Volume 5.
5. Peitz, S.; Dellnitz, M. A Survey of Recent Trends in Multiobjective Optimal Control—Surrogate Models, Feedback Control and Objective Reduction. *Math. Comput. Appl.* **2018**, *23*, 30. [[CrossRef](#)]
6. Schütze, O.; Vasile, M.; Coello Coello, C.A. Computing the Set of Epsilon-Efficient Solutions in Multiobjective Space Mission Design. *J. Aerosp. Comput. Inf. Commun.* **2011**, *8*, 53–70. [[CrossRef](#)]
7. Basto-Fernandes, V.; Yevseyeva, I.; Deutz, A.; Emmerich, M. A Survey of Diversity Oriented Optimization: Problems, Indicators, and Algorithms. In *EVOLVE—A Bridge Between Probability, Set Oriented Numerics and Evolutionary Computation VII*; Emmerich, M., Deutz, A., Schütze, O., Legrand, P., Tantar, E., Tantar, A.A., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 3–23. [[CrossRef](#)]
8. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
9. Beume, N.; Naujoks, B.; Emmerich, M. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *Eur. J. Oper. Res.* **2007**, *181*, 1653–1669. [[CrossRef](#)]
10. Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [[CrossRef](#)]
11. Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Trans. Evol. Comput.* **2014**, *18*, 577–601. [[CrossRef](#)]
12. Cuate, O.; Schütze, O. Variation Rate: An Alternative to Maintain Diversity in Decision Space for Multi-Objective Evolutionary Algorithms. In *Evolutionary Multi-Criterion Optimization*; Deb, K., Goodman, E., Coello Coello, C.A., Klamroth, K., Miettinen, K., Mostaghim, S., Reed, P., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 203–215.
13. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 257–271. [[CrossRef](#)]

14. Dilettoso, E.; Rizzo, S.A.; Salerno, N. A Weakly Pareto Compliant Quality Indicator. *Math. Comput. Appl.* **2017**, *22*, 25. [[CrossRef](#)]
15. Preuss, M.; Wessing, S. Measuring Multimodal Optimization Solution Sets with a View to Multiobjective Techniques. In *EVOLVE—A Bridge Between Probability, Set Oriented Numerics, and Evolutionary Computation IV*; Emmerich, M., Deutz, A., Schuetze, O., Bäck, T., Tantar, E., Tantar, A.A., Moral, P.D., Legrand, P., Bouvry, P., Coello, C.A., Eds.; Springer International Publishing: Heidelberg, Germany, 2013; pp. 123–137.
16. Schütze, O.; Esquivel, X.; Lara, A.; Coello, C.A.C. Using the averaged Hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Trans. Evol. Comput.* **2012**, *16*, 504–522. [[CrossRef](#)]
17. Bogoya, J.M.; Vargas, A.; Cuaté, O.; Schütze, O. A (p,q)-Averaged Hausdorff Distance for Arbitrary Measurable Sets. *Math. Comput. Appl.* **2018**, *23*, 51. [[CrossRef](#)]
18. Ulrich, T.; Bader, J.; Thiele, L. Defining and Optimizing Indicator-Based Diversity Measures in Multiobjective Search. In *Parallel Problem Solving from Nature, PPSN XI*; Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G., Eds.; Springer: Berlin, Germany, 2010; pp. 707–717.
19. Shir, O.M.; Preuss, M.; Naujoks, B.; Emmerich, M. Enhancing Decision Space Diversity in Evolutionary Multiobjective Algorithms. In *Evolutionary Multi-Criterion Optimization*; Ehrgott, M., Fonseca, C.M., Gandibleux, X., Hao, J.K., Sevaux, M., Eds.; Springer: Berlin, Germany, 2009; pp. 95–109.
20. Srinivas, N.; Deb, K. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evol. Comput.* **1994**, *2*, 221–248. [[CrossRef](#)]
21. Jeffrey, H.; Nafpliotis, N.; Goldberg, D.E. Multiobjective optimization using the Niched Pareto genetic algorithm. *IlliGAL Rep.* **1993**, 93005. Available online: https://www.researchgate.net/publication/2763393_Multiobjective_Optimization_Using_The_Niche_Pareto_Genetic_Algorithm (accessed on 13 September 2019).
22. Deb, K.; Tiwari, S. Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization. *Eur. J. Oper. Res.* **2008**, *185*, 1062–1087. [[CrossRef](#)]
23. Castillo, J.C.; Segura, C.; Aguirre, A.H.; Miranda, G.; León, C. A Multi-Objective Decomposition-Based Evolutionary Algorithm with Enhanced Variable Space Diversity Control. In Proceedings of the GECCO 2017, Berlin, Germany, 15–19 July 2017; ACM: New York, NY, USA, 2017; pp. 1565–1571.
24. Ulrich, T.; Bader, J.; Zitzler, E. Integrating Decision Space Diversity into Hypervolume-Based Multiobjective Search. In Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, Portland, OR, USA, 7–10 July 2010; ACM: New York, NY, USA, 2010; pp. 455–462. [[CrossRef](#)]
25. Rudolph, G.; Naujoks, B.; Preuss, M. Capabilities of EMOA to Detect and Preserve Equivalent Pareto Subsets. In Proceedings of the EMO 2017, Münster, Germany, 19–22 March 2017; Springer: Berlin, Germany, 2007; pp. 36–50.
26. Deb, K.; Thiele, L.; Laumanns, M.; Zitzler, E. Scalable Test Problems for Evolutionary Multiobjective Optimization. In *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*; Abraham, A., Jain, L., Goldberg, R., Eds.; Springer: London, UK, 2005; pp. 105–145.
27. Jain, H.; Deb, K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach. *IEEE Trans. Evol. Comput.* **2014**, *18*, 602–622. [[CrossRef](#)]
28. Huband, S.; Hingston, P.; Barone, L.; While, L. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans. Evol. Comput.* **2006**, *10*, 477–506. [[CrossRef](#)]
29. Schütze, O.; Witting, K.; Ober-Blöbaum, S.; Dellnitz, M. Set Oriented Methods for the Numerical Treatment of Multiobjective Optimization Problems. In *EVOLVE—A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation*; Tantar, E., Tantar, A.A., Bouvry, P., Del Moral, P., Legrand, P., Coello Coello, C.A., Schütze, O., Eds.; Springer: Berlin, Germany, 2013; pp. 187–219.
30. Sun, J.Q.; Xiong, F.R.; Schütze, O.; Hernández, C. *Cell Mapping Methods*; Springer: Singapore, 2018.
31. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum]. *IEEE Comput. Intell. Mag.* **2017**, *12*, 73–87. [[CrossRef](#)]
32. Gibbons, J.D.; Chakraborti, S. *Nonparametric Statistical Inference*; Springer: Berlin, Germany, 2011.
33. Li, K.; Wang, R.; Zhang, T.; Ishibuchi, H. Evolutionary Many-Objective Optimization: A Comparative Study of the State-of-the-Art. *IEEE Access* **2018**, *6*, 26194–26214. [[CrossRef](#)]

