VSD-MOEA: A Dominance-Based Multi-Objective Evolutionary Algorithm with Explicit Variable Space Diversity Management

Joel Chacón Castillo, Carlos Segura, *Member, IEEE,* and Carlos A. Coello Coello, *Fellow, IEEE,*

***Abstract*—Most state-of-the-art Multi-Objective Evolutionary Algorithms (MOEAs) promote the preservation of the diversity of the objective space, whereas the information about the diversity of the variable space is usually neglected. In this paper, the Variable Space Diversity based MOEA (VSD-MOEA) is presented. VSD-MOEA is a dominance-based MOEA that explicitly considers the diversity of the variable and objective space. The information gathered on both spaces is used simultaneously with the aim of properly adapting the balance between exploration and intensification during the optimization process. Particularly, at the initial stages, decisions made by the approach are more biased by the information on the diversity of the variable space, whereas in the last stages, decisions are only based on the information on the objective space. The latter is achieved through a novel density estimator. The new method is compared with state-of-art MOEAs using several benchmarks with two and three objectives. This novel proposal yields much better results than state-of-the-art schemes when considering metrics of the objective space, exhibiting a more stable and robust behavior.**

I. Introduction

MULTI-OBJECTIVE Optimization Problems (MOPs) involve the simultaneous optimization of several objective functions that are usually in conflict [1]. A continuous box-constrained minimization MOP, which is the kind of problem addressed in this paper, can be defined as follows:

$$\begin{aligned} &minimize \quad \vec{F} = [f_1(\vec{\mathbf{x}}), f_2(\vec{\mathbf{x}}), ..., f_M(\vec{\mathbf{x}})] \\ &subject \ \ to \quad x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, ..., n. \end{aligned} \tag{1}$$

where n corresponds to the dimension of the variable space, $\vec{\mathbf{x}}$ is a vector of n variables $\vec{\mathbf{x}} = (x_1, ..., x_n) \in R^n$, which are constrained by $x_i^{(L)}$ and $x_i^{(U)}$, i.e. the lower bound and upper bound, and M is the number of objective functions to optimize. The feasible space bounded by $x_i^{(L)}$ and $x_i^{(U)}$ is denoted by Ω. Each solution is mapped to the objective space with the function $F : \Omega \rightarrow R^M$, which consist of M real-valued objective functions, and R^M is called the *objective space*.

Given two solutions $\vec{\mathbf{x}}$, $\vec{\mathbf{y}} \in \Omega$, $\vec{\mathbf{x}}$ dominates $\vec{\mathbf{y}}$, mathematically denoted by $\vec{\mathbf{x}} \prec \vec{\mathbf{y}}$, iff $\forall m \in 1, 2, ..., M : f_m(\vec{\mathbf{x}}) \leq f_m(\vec{\mathbf{y}})$ and $\exists m \in 1, 2, ..., M : f_m(\vec{\mathbf{x}}) < f_m(\vec{\mathbf{y}})$. The best solutions of a MOP are those that are not dominated by any other feasible vector. These solutions are known as the Pareto optimal solutions. The Pareto set is the set of all Pareto optimal solutions, and the Pareto front are the images of the Pareto set. The goal of most multi-objective optimizers is to obtain a proper approximation of the Pareto front, i.e., a set of well distributed solutions that are close to the Pareto front.

One of the most popular meta-heuristics used to deal with MOPs is the Evolutionary Algorithm (EA). In single-objective EAs, it has been shown that taking into account the diversity of the variable space to properly balance between exploration and exploitation is highly important to attain high quality solutions [2]. Diversity can be taken into account in the design of several components, such as in the variation stage [3], [4], replacement phase [5] and/or population model [6]. The explicit consideration of diversity leads to improvements in terms of avoiding premature convergence, meaning that taking into account diversity in the design of EAs is specially important when dealing with long-term executions. Recently, some diversity management algorithms that combine the information on diversity, stopping criterion and elapsed generations have been devised. They have yielded a gradual loss of diversity that depends on the time or evaluations granted to the execution [5]. Specifically, the aim of such a methodology is to promote exploration in the initial generations and gradually alter the behavior towards intensification. These schemes have provided highly promising results. For instance, new best-known solutions for some well-known variants of the frequency assignment problem [7], and for a two-dimensional packing problem [5] have been attained using the same principles. Additionally, this principle guided the design of the winning strategy at the Second Wind Farm Layout Optimization Competition[1], which was held in the Genetic and Evolutionary Computation Conference. Thus, the benefits of this type of design patterns have been shown in several different single-objective optimization problems.

One of the goals when designing Multi-objective Evolutionary Algorithms (MOEAs) is to obtain a well-spread set of solutions in the objective space. As a result, most state-of-the-art MOEAs consider the diversity of the objective space explicitly. However, this is not the case for the diversity of the variable space. Maintaining some degree of diversity in the objective space implies that complete convergence does

J. Chacón, and C. Segura are with the Center for Research in Mathematics (CIMAT), Computer Science Department, Callejón Jalisco s/n, Mineral de Valenciana, Guanajuato, Guanajuato 36240, Mexico (e-mail: joel.chacon@cimat.mx; carlos.segura@cimat.mx)

C. A. Coello Coello is with the Evolutionary Computation Group, Computer Science Department, CINVESTAV-IPN, Mexico City 07300, Mexico (e-mail: ccoello@cs.cinvestav.mx).

Digital Object Identifier xxx

Manuscript received XX, XX 2019; revised XX XX, XX.

[1]https://www.irit.fr/wind-competition/

not appear in the variable space [8]. In some way, the variable space inherits some degree of diversity due to the way in which the objective space is taken into account. However, this is just an indirect way of preserving the diversity of the variable space, so in some cases the level of diversity might not be large enough to ensure a proper degree of exploration. For instance, it has been shown that with some of the WFG benchmarks, in most state-of-the-art MOEAs the *distance parameters* quickly converge, meaning that the approach focuses just on optimizing the *position parameters* for a long period of the optimization process [8]. Thus, while some degree of diversity is maintained, a similar situation to premature convergence occurs, meaning that genetic operators might not be able to generate better trade-offs.

In light of the differences between state-of-the-art single-objective EAs and MOEAs, this paper proposes a novel MOEA, the Variable Space Diversity based MOEA (VSD-MOEA), which relies on explicitly controlling the amount of diversity in the variable space. Similarly to the successful methodology applied in single-objective optimization, the stopping criterion and the number of evaluations performed are used to vary the amount of diversity desired in the variable space. The main difference with respect to the single-objective case is that the diversity of the objective space is simultaneously considered by using a novel objective space density estimator. Particularly, the approach grants more importance to the diversity of the variable space in the initial stages, and as the generations evolve, it gradually grants more importance to the diversity of the objective space. In fact, in the last period of the execution, the diversity of the variable space is neglected, so in the last phases the proposal is quite similar to current state-of-the-art approaches. To our knowledge, this is the first MOEA whose design follows this adaptive principle. Since there currently exist quite a large number of different MOEAs [9], three popular schemes have been selected to validate our proposal. This validation was performed with several well-known benchmarks and proper quality metrics. This paper clearly shows the important benefits of properly taking into account the diversity of the variable space. In particular, the advantages are clearer in the most complex problems. Note that this is consistent with the single-objective case, where the most important benefits have been obtained in complex multi-modal cases [7].

The rest of this paper is organized as follows. Section II provides a review of related papers. Some key components related to diversity and to the VSD-MOEA design are discussed. The VSD-MOEA proposal is detailed in section III. Section IV is devoted to the experimental validation of the novel proposal. Finally, conclusions and some lines of future work are given in Section V. Note also that some supplementary materials are provided. They include details of the experimental results with additional metrics, as well as an explanatory video.

II. Literature Review

This section is devoted to reviewing some of the most important papers that are closely related to our proposal. First, some of the most popular ways of managing diversity in EAs are presented. Then, the state of the art in MOEAs is summarized.

A. Diversity Management in Evolutionary Algorithms

The proper balance between exploration and exploitation is one of the keys to designing a successful EAs. In the single-objective domain, it is known that properly managing the diversity of the variable space is a way to achieve this balance, and as a consequence, a large number of diversity management techniques have been devised [10]. Specifically, these methods are classified depending on the component(s) of the EA that is modified to alter how much diversity is maintained. A popular taxonomy identifies the following groups [11]: *selection-based*, *population-based*, *crossover/mutation-based*, *fitness-based*, and *replacement-based*. Additionally, the methods are referred to as *uniprocess-driven* when a single component is altered, whereas the term *multiprocess-driven* is used to refer to those methods that act on more than one component.

Among the previous proposals, the replacement-based methods have yielded very high-quality results in recent years [7], so this alternative was selected with the aim of designing a novel MOEA that explicitly incorporates a way to control the diversity of the variable space. The basic principle of these methods is to bias the level of exploration in successive generations by controlling the diversity of the survivors [7]. Since premature convergence is one of the most common drawbacks in the application of EAs, modifications are usually performed with the aim of slowing down the convergence. One of the most popular proposals belonging to this group is the *crowding* method, which is based on the principle that offspring should replace similar individuals from the previous generation [12]. Several replacement strategies that do not rely on crowding have also been devised. In some methods, diversity is considered as an objective. For instance, in the hybrid genetic search with adaptive diversity control (HGSADC) [13], individuals are sorted by their contribution to diversity and by their original cost. Then, the rankings of the individuals are used in the fitness assignment phase. A more recent proposal [7] incorporates a penalty approach to gradually alter the amount of diversity maintained in the population. Specifically, the initial phases preserve a higher amount of diversity than the final phases of the optimization. This last method has inspired the design of the novel proposal put forth in this paper for multi-objective optimization.

It is important to remark that in the case of multi-objective optimization, little work related to maintaining the diversity of the variable space has been done. The following section reviews some of the most important MOEAs and introduces some of the works that consider the maintenance of diversity of the variable space.

B. Multi-objective Evolutionary Algorithms

In recent decades, several MOEAs have been proposed. While most of them seek to provide a well-spread set of solutions close to the Pareto front, several ways of achieving this purpose have been devised. Therefore, several taxonomies have been proposed with the aim of better classifying the different schemes [14]. Particularly, a MOEA can be designed based on Pareto dominance, indicators and/or decomposition [15]. Since none of the groups is significantly superior to the others, in this work all of them are taken into account to validate our proposal. This section introduces the three types of schemes and some of the most popular approaches belonging to each category. Then, one MOEA in each category is selected to validate the VSD-MOEA.

The dominance-based category includes those schemes where the Pareto dominance relationship is used to guide the design of some of its components, such as the fitness assignment, parent selection and replacement phase. The dominance relationship does not inherently promote the preservation of diversity in the objective space; therefore, additional techniques such as objective space density estimators are usually integrated in order to obtain a proper spread and convergence to the Pareto front. The most popular dominance-based MOEA is the Non-Dominated Sorting Genetic Algorithm II (NSGA-II) [16].

Several quality indicators have been devised to assess the performance of MOEAs. In indicator-based MOEAs, the use of the Pareto dominance relationship is substituted by some quality indicators to guide the decisions made by the MOEA. An advantage of this kind of algorithm is that the indicators usually take into account both the quality and diversity of the objective space, so incorporating additional mechanisms to promote diversity in the objective space is not required. The Indicator-Based Evolutionary Algorithm (IBEA) [17] was the first method belonging to this category. A more recent one is the R2-Indicator-Based Evolutionary Multi-objective Algorithm (R2-EMOA) [18], whose performance in MOPs has been quite promising. Its most important feature is the use of the R2 indicator.

Finally, decomposition-based MOEAs [19] are based on transforming the MOP into a set of single-objective optimization problems that are tackled simultaneously. This transformation can be performed in several ways, e.g. with a linear weighted sum or with a weighted Tchebycheff function. Given a set of weights to establish different single-objective functions, the MOEA searches for a single high-quality solution for each of them. The weight vectors should be selected with the aim of obtaining a well-spread set of solutions [1]. The Multi-objective Evolutionary Algorithm Based on Decomposition (MOEA/D) [20] is the most popular decomposition-based MOEA. Its main principles include problem decomposition, weighted aggregation of objectives and mating restrictions through the use of neighborhoods.

It is important to note that none of the most popular algorithms in the multi-objective field introduces special mechanisms to promote diversity of the variable space. However, some efforts have been devoted to this principle. A popular approach to promote the diversity of the variable space is the application of fitness sharing [21], in a way similar to single-objective optimization. Although fitness sharing might be used to promote the diversity of both the objective and variable spaces, most popular variants consider only distances in the objective space. Another MOEA designed to promote diversity of both the variable and the objective space is the Genetic Diversity Evolutionary Algorithm (GDEA) [22]. In this case, each individual is assigned a diversity-based objective which is calculated as the Euclidean distance in the genotype space to the remaining individuals in the population. Then, a ranking that considers both the original objectives and the diversity objective is used to sort individuals. Another somewhat popular approach is to calculate distances between candidate solutions by taking into account both the objective and variable space [23], [24] with the aim of promoting diversity of both spaces. A different proposal combines the use of two selection operators [25]. The first one promotes diversity and quality in the objective space, whereas the second one promotes diversity in the variable space. A different approach involves modifying the hypervolume to integrate the variable space diversity into a single metric [26]. In this approach, the proposed metric is used to guide the selection in the MOEA. Finally, some indirect mechanisms that might affect diversity have also been taken into account. The most popular one is probably the use of mating restrictions [27], [19].

In light of the results of the approaches described above, it is clear that considering the diversity of the variable space in the design phase might yield benefits for decision makers, since the final solutions obtained by these methods exhibit a higher variable space diversity than those obtained by traditional approaches [23], [28]. Thus, while clear improvements are obtained when metrics related to the variable space are taken into account, the benefits in terms of the objective space are not so clear. We claim that one of the reasons for this behavior might be that the diversity of the variable space is considered in the whole optimization process. However, in a similar way as in the single objective domain, reducing the importance granted to the diversity of the variable space as the generations progress [5] might be truly important to obtaining better approximations of the Pareto front. Currently, no MOEA considers this idea, so it is this principle that has guided the design of our novel MOEA.

III. Proposal

This section provides a full description of our novel proposal called, *Variable Space Diversity based MOEA* (VSD-MOEA) [2]. The novelty of VSD-MOEA appears in the replacement phase, which incorporates the use of variable space diversity and a novel objective space density estimator. The main principle behind the design of the novel replacement is to use the stopping criterion and elapsed generations with the aim of gradually moving from exploration to exploitation during the search process. Note that this principle might be incorporated in any of the three categories of MOEAs. In this paper, our decision was to incorporate it in a dominance-based approach. Note that this category has been particularly suitable for problems with two and three objectives. Thus, some of our design decisions might not be suitable for dealing with many-objective optimization problems.

The general framework of VSD-MOEA is quite standard. Algorithm 1 shows the pseudo-code of VSD-MOEA. Parents are

[2]The source code in C++ is freely available at https://github.com/joelchaconcastillo/VSD-MOEA.git

Algorithm 1 Main procedure of VSD-MOEA

Initialization: Generate an initial population P_0 with N individuals.
Evaluation: Evaluate all individuals in the population.
Assign $t = 0$
while (not stopping criterion) **do**
  Mating selection: Fill the mating pool by performing binary tournament selection on P_t, based on the non-dominated ranks (ties are broken randomly).
  Variation: Apply SBX crossover and Polynomial mutation to the mating pool to create a child population Q_t.
  Evaluation: Evaluate all individuals in Q_t.
  Survivor selection: Generate P_{t+1} by applying the replacement scheme described in Algorithm 2, using P_t and Q_t as inputs.
  $t = t + 1$

selected using a binary tournament based on dominance ranking with ties broken randomly. The variation stage is based on applying the well-known Simulated Binary Crossover (SBX) and polynomial mutation [29], [30]. Thus, the contribution appears in the replacement phase. The rest of this section is devoted to describing the replacement phase, including the novel objective space density estimator.

A. Replacement Phase of VSD-MOEA

The replacement phase of EAs is in charge of deciding, for each generation, which members of the previous population and offspring survive. The novel replacement promotes a gradual movement from exploration to exploitation, which has been a highly beneficial principle in the design of single-objective optimizers [5]. Specifically, the replacement phase operates as follows. First, the members of the previous population and offspring are joined in a multi-set with $2 \times N$ individuals. Then, an iterative process that selects an additional individual at each iteration is used to pick the N survivors. In order to take into account the diversity of the variable space, the Distance to Closest Survivor (DCS) of each individual is calculated at each iteration. Thus, the DCS of an individual I is calculated as $\min_{s \in S} Distance(I, s)$, where S is the multi-set containing the currently selected survivors. Normalized Euclidean distances are considered, so in order to calculate distances between any two individuals A and B, Eq. (2) is applied. In the first iteration, the S multi-set is empty, so the DCS of each individual is infinity.

$$Distance(A, B) = \left(\frac{1}{n} \sum_{i=1}^{n} \left(\frac{A_i - B_i}{x_i^{(U)} - x_i^{(L)}} \right)^2 \right)^{1/2} \quad (2)$$

Note that individuals with larger DCS values are those that contribute more significantly to promoting exploration. In order to avoid an excessive decrease in the degree of exploration, individuals with a DCS value below a certain threshold are penalized. Then, among the non-penalized individuals, an objective space density estimator is used to select the additional survivor of the iteration. In our case, the novel density estimator described in the next subsection is used. Note that it might happen that all individuals are penalized, in which case the individual with the largest DCS is selected to survive.

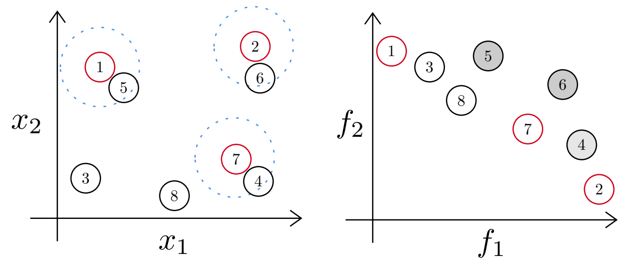

Fig. 1. Penalty Method of the Replacement Phase - The left side represents the variable space and the right side the objective space.

In order to better understand the penalty method, it can be visualized in the following way. After selecting each survivor, a hyper-sphere centered around a candidate solution — in the variable space — is created. Then, all the individuals that are inside the hyper-sphere are penalized, with the objective space estimator only taking into account the survivors and non-penalized individuals. This is illustrated in Fig. 1, which represents a state where three individuals have been selected to survive and an additional survivor must be picked. The left side shows individuals in the variable space. Current survivors are marked with a red border. Each one is surrounded by a dashed blue circle of radius D_t. In this scenario, the penalized individuals are numbers 4, 5, and 6. In the objective space — right side — the penalized individuals are shown in gray, indicating that the objective space density estimator is not considering them.

Since using a large radius for the hyper-spheres induces a large degree of exploration, it makes sense to reduce this value during the optimization process. This is precisely one of the keys of our proposal. The sizes of the hyper-spheres are modified dynamically by taking into account the stopping criterion and elapsed generations. Specifically, the radius is decreased linearly starting from an initial distance. This means that in the initial phases, exploration is promoted. However, as the size of the radius decreases, only very close individuals are penalized, meaning that more exploitation is allowed. Note that this method requires a parameter that is the initial radius of the hyper-spheres or initial threshold value. This parameter is denoted by D_I. Assigning a large value to this parameter might result in many individuals being penalized, which might thus maintain non-useful diversity. However, a value that is too small might not prevent fast convergence, meaning the approach might behave as a traditional non-diversity based MOEA. The robustness of the proposal with respect to this additional parameter is studied in our experimental validation.

Algorithm 2 formalizes the replacement phase of VSD-MOEA. First, the population of the previous generation (P_t) and the offspring (Q_t) are joined in R_t (line 3). In each iteration, the multi-set R_t contains the remaining non-penalized individuals that might be selected to survive. The population of survivors (P_{t+1}) and the set containing the penalized individuals are initialized to the empty set (lines 4 and 5). Then, the threshold value (D_t) that is used to penalize individuals that are too close is calculated (line 6). Note that D_I denotes the initial threshold value, $G_{Elapsed}$ is the number of generations

Algorithm 2 Replacement Phase of VSD-MOEA

```
1: Input: P_t (Population of current generation), Q_t (Offspring of current Generation)
2: Output: P_{t+1}
3: R_t = P_t ∪ Q_t
4: P_{t+1} = ∅
5: Penalized = ∅
6: D_t = D_I − D_I * G_Elapsed / (0.5*G_End)
7: while |P_{t+1}| ≤ N do
8:     Compute DCS of individuals in R_t, using P_{t+1} as a reference set
9:     Move the individuals in R_t with DCS < D_t to Penalized
10:    if R_t is empty then
11:        Compute DCS of individuals in Penalized, using P_{t+1} as a reference set
12:        Move the individual in Penalized with the largest DCS to R_t
13:    Identify the first front (F) in R_t ∪ P_{t+1} with an individual I ∈ R_t
14:    Use the novel density estimator (Algorithm 3) to select a new survivor from F and move it to P_{t+1}
15: return P_{t+1}
```

that have evolved, and G_{End} is the stopping criterion, i.e. the number of generations that are to be evolved during the execution of the VSD-MOEA. The linear decrease is calculated such that after 50% of the generations, the D_t value is below 0, meaning that no penalties are performed. This means that in the first 50% of the generations, more exploration is induced than in traditional MOEAs.

Then, an iterative process that selects an individual in each iteration is executed until the survivor set contains N individuals (line 7). The iterative process works as follows. First, the DCS value of each remaining non-penalized individual is calculated (line 8). Then, those individuals with a DCS value lower than D_t are moved to the set of penalized individuals (line 9). If all the remaining individuals are penalized (line 10), it means that the amount of exploration is less than desired. Thus, the individual with the largest DCS value is recovered, i.e. moved to the set of non-penalized individuals (lines 11 and 12), and thus survives. Finally, the objective space is considered. Specifically, candidate non-penalized individuals and current survivors are joined. Then, the well-known non-dominated sorting procedure [16] is executed on this set, stopping as soon as a front with a candidate individual is found, i.e. with an individual of R_t (line 13). Then, taking the identified front as an input, a novel objective space density estimator is used to select the next survivor (line 14). The specific way in which each individual's contribution to the diversity of the objective space is measured is described in the next section.

B. A Novel Density Estimator for the Objective Space

Since the dominance definition is not related to the preservation of diversity in the objective space, dominance-based MOEAs usually incorporate objective-space density estimators to promote the survival of diverse individuals. As previously described, our density estimator selects a new survivor from the front identified in line 14 of Algorithm 2. This front (referred to in Algorithm 3 as F) contains at least one

Algorithm 3 Density estimator

```
1: Input: P_{t+1} (Survivors), R_t (Candidates), F (Current front)
2: Output: I ∈ R_t
3: FP = P_{t+1} ∩ F
4: FR = R_t ∩ F
5: for k ∈ number of objectives do
6:     Select the best individual I ∈ F of k according to Eq. 3.
7:     if I ∈ FR then
8:         return I
9: MaxID = 0
10: for Ic ∈ FR do
11:     Improvement = min_{s∈FP} ID(Ic, s)
12:     if Improvement > MaxID then
13:         MaxID = Improvement
14:         I = Ic
15: return I
```

individual belonging to R_t, and it might also contain some elements of P_{t+1}. The aim behind the selection of the next survivor is to pick an individual of the input front that contributes significantly in terms of the quality and diversity of the objective space.

Algorithm 3 describes the selection of the next survivor. First, the sets FP and FR are identified (lines 3 and 4). FP contains the current survivors that are in F, whereas FR contains the remaining non-penalized individuals that are in F. Then, similarly to most state-of-the-art algorithms, a step to promote the selection of boundary solutions is included. Note that selecting the best solution for each objective might cause some drawbacks related to accepting a small improvement in one objective at the expense of significant degradation in other objectives [31]. This issue can be resolved by applying augmented functions, which is the alternative used in this paper. Specifically, for each objective k, the candidate solution that minimizes the Augmented Function (AF) given in Eq. 3 is iteratively identified (lines 5 to 8). If this individual belongs to FR, i.e., it has not yet been selected as a survivor, the next survivor is such an individual and the process ends (line 8). Note that augmented functions usually take into account weight vectors in order to deal with objectives that exhibit very different scales. Since benchmarks that have similar scales in each objective have been used in this paper, there was no need to apply said weight vectors.

$$AF_k(\vec{x}) = f_k(\vec{x}) + 10^{-4} \times \sum_{j=1}^{M} f_j(\vec{x}) \tag{3}$$

In cases where the individuals that optimize each AF_K function are already in P_{t+1}, a contribution to objective-space diversity is calculated for each individual in FR (lines 9 to 15). This contribution is calculated by taking into account the current survivors of the front (FP). Specifically, the "Improvement Distance" (ID) defined for the indicator IGD+ [32] is used. The ID of an individual A with respect to an individual B is calculated by taking into account only the objectives where A is better. Specifically, Eq. (4) is used.

$$ID(A,B) = \left(\sum_{i=1}^{M} (max(0, B_i - A_i))^2 \right)^{1/2} \tag{4}$$

The contribution of each member of $FR(I)$ is calculated as $\min_{s \in FP} ID(I, s)$. Then, the individual with the highest contribution is selected as the next survivor (lines 12 to 14).

IV. Experimental Validation

This section describes the experimental validation carried out to study the performance and gain a clear understanding of the specifics of VSD-MOEA. Our results clearly show that controlling the diversity of the variable space provides a way to further improve the results obtained by the state-of-art MOEAs. First, we discuss some technical specifications involving the benchmark problems and algorithms implemented. We then present a comparison between VSD-MOEA and state-of-the-art algorithms when used on the long-term. Then, three additional experiments to fully validate VSD-MOEA are included. These analyses are designed to test the scalability in the variable space, the performance with different stopping criteria, and the behavior with different initial penalty thresholds.

This work takes into account some of the most popular and widely used benchmarks in the multi-objective field. These problems are the WFG [33], DTLZ [34], and UF [35] configured in a standard way. The WFG test problems were used with two and three objectives and were configured with 24 parameters, 20 of them corresponding to distance parameters and 4 to position parameters. In the DTLZ test problems, the number of variables was set to $n = M + r - 1$, where $r = \{5, 10, 20\}$ for DTLZ1, DTLZ2 to DTLZ6 and DTLZ7, respectively. The UF benchmark comprises seven problems with two objectives (UF1-7) and three problems with three objectives (UF8-10). All of them were configured with 30 variables. Note that the experiment used to analyze the scalability in the variables considers different numbers of variables.

The experimental validation includes three well-known state-of-the-art MOEAS and VSD-MOEA. The MOEAS that are considered are NSGA-II [36], MOEA/D [37], and R2-EMOA [38], which can be classified as dominance-based, decomposition-based, and indicator-based, respectively. In the case of MOEA/D, several variants have been devised. The MOEA/D implementation considered is the one that obtained first place in the Congress on Evolutionary Computation's 2009 MOP Competition [39]. The common configuration in all the experiments was as follows: the population size was set to 100, and the genetic operators were the Simulated Binary Crossover (SBX) and polynomial mutation [29], [30]. The crossover probability was set to 0.9 and the crossover distribution index was set to 2. Similarly, the mutation probability and distribution index were fixed to $1/n$ and 50, respectively. The additional parameterization required by each algorithm is shown in Table I. Note that scalarization functions are required in MOEA/D and R2-EMOA. In both cases, the Tchebycheff approach is used. The procedure for generating the weight vectors differs in MOEA/D and R2-EMOA. R2-EMOA was applied with 501 and 496 weight vectors for two and three objectives, respectively [18]. In contrast, MOEA/D requires the same number of weight vectors as the population size. They were generated with the uniform design (UD) and the good lattice point (GLP) method [40], [41].

TABLE I
PARAMETERIZATION APPLIED TO EACH MOEA

Algorithm	Configuration
MOEA/D	Max. updates by sub-problem (η_r) = 2, tour selection = 10, neighbor size = 10, period utility updating = 30 generations, local selection probability (δ) = 0.9,
VSD-MOEA	$D_I = 0.4$
R2-EMOA	$\rho = 1$, offspring by iteration = 1

TABLE III
STATISTICAL TESTS AND DETERIORATION LEVEL OF THE HV RATIO FOR PROBLEMS WITH TWO OBJECTIVES

	$\uparrow$	$\downarrow$	$\leftrightarrow$	**Deterioration**
MOEA/D	24	36	9	1.615
NSGA-II	13	49	7	1.496
R2-EMOA	34	21	14	1.597
VSD-MOEA	50	15	4	0.059

Given that all the algorithms considered are stochastic, each execution was repeated 35 times with different seeds. The hypervolume indicator (HV) is used to compare the various schemes. Note that in the supplementary material, the results are also compared in terms of the IGD+ metric, with the conclusions being quite similar. The reference point used to calculate the HV is chosen to be a vector whose values are sightly larger (ten percent) than the nadir point, as suggested in [42]. The normalized HV is used to facilitate the interpretation of the results [43], and the value reported is computed as the ratio between the normalized HV obtained and the maximum attainable normalized HV. In this way, a value equal to one means a perfect approximation. Note that a value equal to one is not attainable because MOEAs yields a discrete approximation. Finally, in order to statistically compare the HV ratios, a guideline similar to that proposed in [44] was used. First a Shapiro-Wilk test was performed to check if the values of the results followed a Gaussian distribution. If so, the Levene test was used to check for the homogeneity of the variances. If the samples had equal variance, an ANOVA test was done; if not, a Welch test was performed. For non-Gaussian distributions, the non-parametric Kruskal-Wallis test was used to test whether samples are drawn from the same distribution. An algorithm X is said to beat algorithm Y when the differences between them are statistically significant, and the mean and median HV ratios obtained by X are higher than the mean and median achieved by Y.

A. Comparison against State-of-the-art MOEAs in long-term executions

Our first experiment aims to compare the long-term performance of VSD-MOEA against state-of-the-art proposals, which is the kind of execution where diversity-based EAs have been more successful. Specifically, the stopping criterion was set at $250,000$ generations.

Table II shows the HV ratio obtained for the benchmark functions with two objectives. Specifically, the minimum, maximum, mean and standard deviation of the HV ratio is shown for each method and function tested. The last row shows the results considering all the functions together. For

TABLE II
SUMMARY OF THE HYPERVOLUME RATIO RESULTS ATTAINED FOR PROBLEMS WITH TWO OBJECTIVES

	MOEA/D				**NSGA-II**				**R2-EMOA**				**VSD-MOEA**			
	Min	**Max**	**Mean**	**Std**	**Min**	**Max**	**Mean**	**Std**	**Min**	**Max**	**Mean**	**Std**	**Min**	**Max**	**Mean**	**Std**
WFG1	**0.984**	**0.993**	**0.992**	**0.002**	0.987	0.993	0.992	0.002	0.946	0.994	0.988	0.012	0.980	0.994	0.992	0.003
WFG2	0.965	0.996	0.967	0.007	0.966	0.998	0.974	0.014	0.965	0.966	0.966	0.000	**0.998**	**0.998**	**0.998**	**0.000**
WFG3	0.992	0.992	0.992	0.000	0.987	0.988	0.987	0.000	0.991	0.992	0.991	0.000	**0.992**	**0.992**	**0.992**	**0.000**
WFG4	0.988	0.988	0.988	0.000	0.983	0.987	0.985	0.001	**0.991**	**0.991**	**0.991**	**0.000**	0.990	0.990	0.990	0.000
WFG5	0.876	0.893	0.882	0.005	0.884	0.899	0.890	0.002	0.886	0.895	0.891	0.003	**0.894**	**0.928**	**0.914**	**0.010**
WFG6	**0.879**	**0.940**	**0.914**	**0.016**	**0.894**	**0.942**	**0.913**	**0.012**	**0.875**	**0.942**	**0.912**	**0.015**	0.855	0.888	0.868	0.007
WFG7	0.988	0.988	0.988	0.000	0.983	0.987	0.984	0.001	**0.991**	**0.991**	**0.991**	**0.000**	0.990	0.990	0.990	0.000
WFG8	0.800	0.822	0.811	0.006	0.771	0.801	0.789	0.006	0.803	0.824	0.815	0.005	**0.828**	**0.958**	**0.928**	**0.046**
WFG9	0.795	0.972	0.883	0.082	0.793	0.966	0.832	0.070	0.797	0.976	0.884	0.079	**0.963**	**0.975**	**0.970**	**0.004**
DTLZ1	**0.993**	**0.993**	**0.993**	**0.000**	0.990	0.992	0.991	0.000	0.992	0.992	0.992	0.000	0.992	0.992	0.992	0.000
DTLZ2	0.989	0.989	0.989	0.000	0.986	0.988	0.987	0.000	**0.991**	**0.992**	**0.992**	**0.000**	0.990	0.990	0.990	0.000
DTLZ3	0.989	0.989	0.989	0.000	0.987	0.989	0.989	0.001	**0.991**	**0.992**	**0.992**	**0.000**	0.990	0.990	0.990	0.000
DTLZ4	0.259	0.989	0.781	0.330	0.259	0.988	0.863	0.274	0.259	0.992	0.657	0.365	**0.990**	**0.990**	**0.990**	**0.000**
DTLZ5	0.989	0.989	0.989	0.000	0.986	0.988	0.987	0.000	**0.991**	**0.992**	**0.992**	**0.000**	0.990	0.990	0.990	0.000
DTLZ6	0.448	0.910	0.700	0.105	0.138	0.511	0.322	0.075	0.510	0.922	0.691	0.107	**0.990**	**0.990**	**0.990**	**0.000**
DTLZ7	0.996	0.996	0.996	0.000	0.996	0.997	0.996	0.000	**0.997**	**0.997**	**0.997**	**0.000**	0.996	0.996	0.996	0.000
UF1	0.991	0.993	0.992	0.000	0.986	0.989	0.988	0.000	0.978	0.994	0.990	0.005	**0.994**	**0.995**	**0.994**	**0.000**
UF2	**0.987**	**0.993**	**0.991**	**0.002**	0.980	0.983	0.981	0.001	0.984	0.991	0.988	0.002	0.987	0.993	0.990	0.001
UF3	0.481	0.674	0.597	0.043	0.678	0.871	0.784	0.048	0.531	0.704	0.589	0.041	**0.799**	**0.916**	**0.881**	**0.025**
UF4	0.881	0.917	0.908	0.006	0.875	0.910	0.889	0.008	**0.923**	**0.935**	**0.929**	**0.003**	0.923	0.931	0.927	0.002
UF5	0.035	0.792	0.484	0.165	0.256	0.766	0.641	0.104	0.123	0.792	0.566	0.192	**0.582**	**0.763**	**0.647**	**0.040**
UF6	0.255	0.711	0.447	0.114	0.235	0.801	0.635	0.120	0.349	0.767	0.568	0.113	**0.668**	**0.900**	**0.810**	**0.061**
UF7	**0.987**	**0.991**	**0.990**	**0.001**	0.980	0.983	0.981	0.001	0.557	0.991	0.910	0.150	**0.975**	**0.992**	**0.988**	**0.004**
Mean	0.806	0.935	0.881	0.038	0.808	0.927	0.886	0.032	0.801	0.940	0.882	0.048	0.929	0.963	0.949	0.009

TABLE V
STATISTICAL TESTS AND DETERIORATION LEVEL OF THE HV RATIO FOR PROBLEMS WITH THREE OBJECTIVES

	$\uparrow$	$\downarrow$	$\leftrightarrow$	**Deterioration**
MOEA/D	16	37	4	1.601
NSGA-II	9	45	3	2.557
R2-EMOA	31	22	4	1.223
VSD-MOEA	52	4	1	0.037

each function, the data for the method that yielded the largest mean is shown in bold. Additionally, all the methods that were not statistically inferior than said method are shown in bold. From here on, the methods shown in bold for a given problem are referred to as the winning methods. Based on the number of functions where each method is in the group of the winning methods for the cases with two objectives, the best methods are VSD-MOEA and R2-EMOA with 12 and 8, respectively. Thus, VSD-MOEA is the most competitive method in terms of this metric. More impressive is the fact that the mean HV ratio attained by VSD-MOEA, when all the problems are considered simultaneously, is much higher than the one attained by R2-EMOA. In fact, the total means of R2-EMOA (0.882), NSGA-II (0.886) and MOEA/D (0.881) are quite similar. In contrast VSD-MOEA achieved a much higher value (0.949). When the data is inspected carefully, it is clear that in the cases where VSD-MOEA loses, the difference with respect to the best method is not very large. For instance, the difference between the mean HV ratio attained by the best method and by VSD-MOEA was never larger than 0.1. However, all the other methods exhibited a deterioration greater than 0.1 in several cases. Specifically, it happened in 5, 5 and 6 problems for R2-EMOA, NSGA-II and MOEA/D, respectively. This means that even if VSD-MOEA loses in some cases, its deterioration is always small, exhibiting a much more robust behavior than any other method.

In order to better clarify these findings, pair-wise statistical tests were done among each method tested in each function. For the two-objective cases, Table III shows the number of times that each method won (column $\uparrow$), lost (column $\downarrow$) and tied (column $\leftrightarrow$). Additionally, for each method M we calculated the sum of the differences between the mean HV ratio attained by the best method (the ones with the highest mean) and method M, for each problem where M was not in the group of winning methods. This value is shown in the Deterioration column. The data confirms that although VSD-MOEA loses in some cases, the overall numbers of wins and losses favors VSD-MOEA. More importantly, the total deterioration is quite lower in the case of VSD-MOEA, confirming that when VSD-MOEA loses, the deterioration is not that large.

Tables IV and V show the same information for the problems with three objectives. In this case, the superiority of VSD-MOEA is even clearer. Taking into account the mean of all the functions, VSD-MOEA again obtained a much larger mean HV ratio than the other methods. Specifically, VSD-MOEA obtained a value of 0.918, whereas the second ranked algorithm (R2-EMOA) obtained a value of 0.855. Once again, the difference between the mean HV ratio obtained by the best method and by VSD-MOEA was never greater than 0.1. However, all the other methods exhibited a deterioration greater than 0.1 in several cases. In particular, this happened in 5, 6 and 6 problems for R2-EMOA, NSGA-II and MOEA/D, respectively. Moreover, in this case, VSD-MOEA is much superior than the other methods not only in terms of total deterioration, but also in terms of total wins and losses (see Table V and data shown in bold in Table IV). VSD-MOEA was in the group of winning methods for 16 out of 19 functions, whereas the second best-ranked algorithm (R2-EMOA) was in the group of winning methods for only 3 functions.

TABLE IV
SUMMARY OF THE HYPERVOLUME RATIO RESULTS ATTAINED FOR PROBLEMS WITH THREE OBJECTIVES

	MOEA/D				NSGA-II				R2-EMOA				VSD-MOEA			
	Min	**Max**	**Mean**	**Std**	**Min**	**Max**	**Mean**	**Std**	**Min**	**Max**	**Mean**	**Std**	**Min**	**Max**	**Mean**	**Std**
WFG1	0.958	0.969	0.966	0.002	0.925	0.945	0.935	0.005	0.968	0.979	0.975	0.002	**0.977**	**0.985**	**0.982**	**0.002**
WFG2	0.973	0.978	0.976	0.001	0.959	0.974	0.968	0.004	0.962	0.963	0.963	0.000	**0.988**	**0.991**	**0.989**	**0.001**
WFG3	**0.992**	**0.992**	**0.992**	**0.000**	0.976	0.988	0.985	0.002	0.991	0.992	0.992	0.000	0.989	0.989	0.989	0.000
WFG4	0.864	0.865	0.865	0.000	0.854	0.883	0.868	0.007	0.903	0.905	0.904	0.000	**0.919**	**0.922**	**0.919**	**0.001**
WFG5	0.795	0.804	0.797	0.002	0.806	0.836	0.821	0.008	0.843	0.853	0.848	0.002	**0.846**	**0.864**	**0.856**	**0.003**
WFG6	0.777	0.832	0.809	0.013	0.788	0.836	0.815	0.011	**0.847**	**0.875**	**0.857**	**0.007**	0.824	0.844	0.831	0.005
WFG7	0.864	0.865	0.865	0.000	0.858	0.889	0.875	0.008	0.901	0.905	0.904	0.001	**0.918**	**0.920**	**0.919**	**0.000**
WFG8	0.778	0.785	0.782	0.002	0.697	0.730	0.716	0.008	0.816	0.821	0.819	0.001	**0.832**	**0.913**	**0.899**	**0.023**
WFG9	0.726	0.851	0.819	0.039	0.720	0.833	0.746	0.027	**0.773**	**0.895**	**0.872**	**0.038**	0.771	0.889	0.864	0.037
DTLZ1	0.950	0.950	0.950	0.000	0.935	0.950	0.943	0.004	0.939	0.943	0.941	0.001	**0.961**	**0.966**	**0.964**	**0.001**
DTLZ2	0.899	0.899	0.899	0.000	0.871	0.901	0.886	0.007	0.913	0.916	0.915	0.001	**0.929**	**0.930**	**0.930**	**0.000**
DTLZ3	0.899	0.899	0.899	0.000	0.876	0.901	0.890	0.006	0.914	0.916	0.915	0.000	**0.929**	**0.930**	**0.930**	**0.000**
DTLZ4	0.151	0.899	0.813	0.238	0.871	0.904	0.888	0.007	0.151	0.916	0.675	0.298	**0.929**	**0.930**	**0.930**	**0.000**
DTLZ5	0.978	0.978	0.978	0.000	0.982	0.984	0.983	0.001	**0.985**	**0.986**	**0.986**	**0.000**	**0.986**	**0.986**	**0.986**	**0.000**
DTLZ6	0.310	0.889	0.591	0.142	0.183	0.382	0.243	0.056	0.400	0.946	0.672	0.143	**0.986**	**0.986**	**0.986**	**0.000**
DTLZ7	0.914	0.914	0.914	0.000	0.907	0.935	0.924	0.006	0.837	0.893	0.860	0.014	**0.963**	**0.965**	**0.964**	**0.000**
UF8	0.151	0.830	0.773	0.107	0.324	0.646	0.463	0.069	0.578	0.917	0.898	0.057	**0.897**	**0.929**	**0.919**	**0.008**
UF9	0.753	0.916	0.846	0.067	0.368	0.782	0.728	0.096	0.778	0.954	0.844	0.079	**0.946**	**0.975**	**0.968**	**0.009**
UF10	0.145	0.555	0.341	0.162	0.060	0.391	0.242	0.067	0.143	0.578	0.413	0.166	**0.415**	**0.740**	**0.611**	**0.098**
Mean	0.730	0.877	0.835	0.041	0.735	0.826	0.785	0.021	0.771	0.903	0.855	0.043	0.895	0.929	0.918	0.010

B. Decision Variable Scalability Analysis

In order to study the scalability of VSD-MOEA in terms of the number of variables, all of the algorithms already described were tested with the same benchmark functions, but considering 50, 100, and 250 variables. Note that in the WFG problems, the number of position parameters (k) and distance parameters (l) must be specified. Specifically, the number of distance parameters was set to 42, 84, and 210 when using 50, 100 and 250 variables, respectively. The rest of the variables were position parameters, meaning that they were 8, 16 and 40, respectively. Note that increasing the number of variables greatly increases the computing time required. As a result, this study takes into account middle-term executions. Specifically, the stopping criterion was set to $25,000$ generations. Figures 2 and 3 show the mean HV ratio for the four algorithms tested, considering the problems with two and three objectives, respectively. As expected, the HV ratio decreases as the number of variables increases. In the two-objective case, the deterioration is similar in every algorithm, so the superiority of VSD-MOEA is clear regardless of the number of variables. In contrast, in the three-objective case, the deterioration of VSD-MOEA is higher than for R2-EMOA and MOEA/D. In fact, when considering 250 variables, the performance of VSD-MOEA is just slightly superior to that of R2-EMOA.

In order to better understand this behavior, we selected problems WFG1 to WFG7. WFG problems divide the variables into two kinds of parameters (this framework uses the term parameter instead of variable): the distance parameters and the position parameters. Note that a parameter i is a distance parameter when for all $\vec{x}$, modifying x_i results in a new solution that dominates $\vec{x}$, is equivalent to $\vec{x}$, or is dominated by $\vec{x}$. However, if i is a position parameter, modifying x_i in $\vec{x}$ always results in a vector that is incomparable or equivalent to $\vec{x}$ [45]. Additionally, note that we selected problems WFG1-WFG7 because their distance parameter values associated to

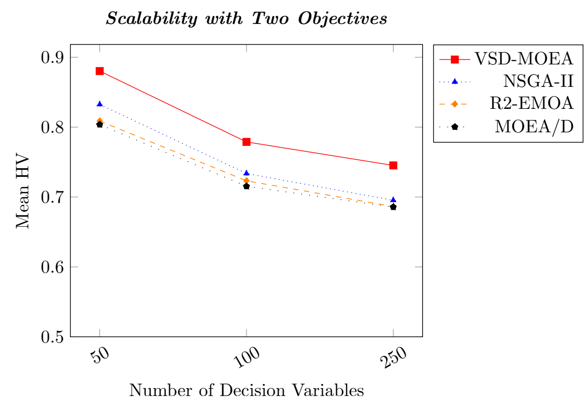

Fig. 2. Mean of the HV ratio for 35 runs for the two-objective problems considering different numbers of variables

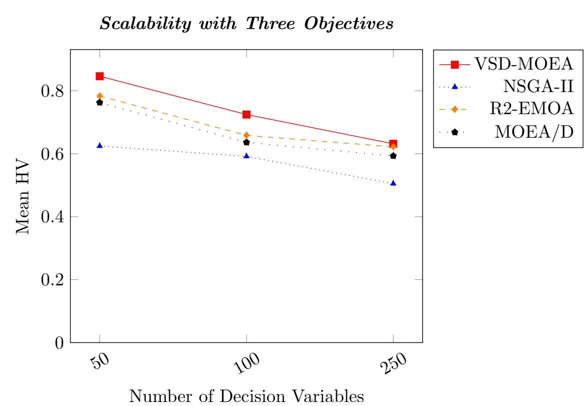

Fig. 3. Mean of the HV ratio for 35 runs for the three-objective problems considering different numbers of variables

all Pareto optimal solutions have exactly the same values:

$$x_{i=k+1:n} = 2i \times 0.35 \quad (5)$$

This is very important because it has been shown that for these kinds of cases, state-of-the-art MOEAs might provoke a quick convergence in *distance parameters*, resulting in an

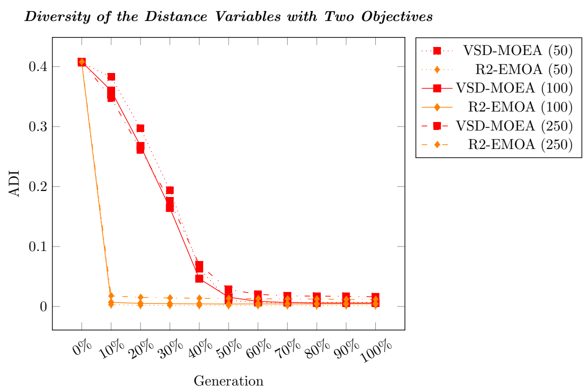

Fig. 4. Evolution of ADI for problems WFG1-WFG7 with two objectives

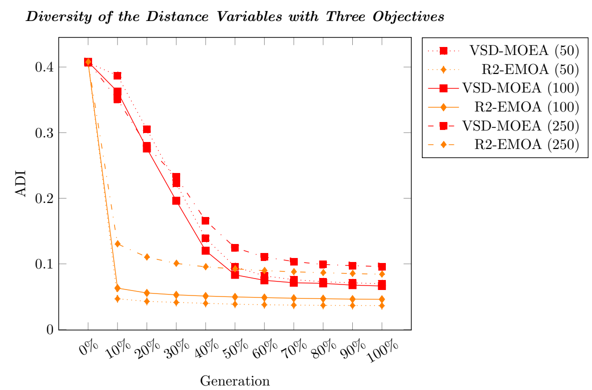

Fig. 5. Evolution of ADI for problems WFG1-WFG7 with three objectives

effect that is similar to premature convergence in the single-objective case [8].

For each algorithm, we calculated the average Euclidean distance among individuals (ADI) in the population by considering only the distance parameters. Figures 4 and 5 show how the ADI evolves for the two-objective and three-objective problems. So as not to saturate these Figures, only the information for VSD-MOEA and R2-EMOA with 50, 100 and 250 variables is shown. The performance of NSGA-II and MOEA/D — which are not included — is similar to that of R2-EMOA in terms of how the ADI evolves. The first obvious fact is that VSD-MOEA converges much slower than R2-EMOA. Accordingly, the difference between the diversity maintained in the first generation and that maintained after 10% of the execution, is much larger in R2-EMOA than in VSD-MOEA. In the case of VSD-MOEA, the decrease in ADI is quite linear until the halfway point of the execution. This is due to the way in which the threshold distance value (D_t) is calculated. Additionally, a closer inspection of the data reveals other important aspects that must be discussed. In the two-objective case, increasing the number of variables causes the diversity in the R2-EMOA to increase slightly. However, the amount of diversity is low even when using 250 variables, meaning that incorporating mechanisms to increase diversity — as is done in VSD-MOEA — is very helpful. In contrast, in the three-objective case, the amount of diversity in R2-EMOA is not as low. Moreover, increasing the number of variables yields a significant increase in the resulting ADI, meaning that in this case, fast convergence is not as important. These results show that, as the number of objectives and variables increases, MOEAs tend to maintain a higher variable space diversity in an implicitly way, meaning that explicitly controlling the variable space diversity is probably not as important.

Finally, we would like to note that we selected some problems to conduct long-term executions with 250 variables. VSD-MOEA was able to further improve the results when using long-term executions, while the other state-of-the-art algorithms did not yield significant improvements. This probably means that as technology evolves, allowing longer executions to be carried out in reasonable timeframes, the incorporation of explicit control of diversity will be even more important. Note that this also happens in the single-objective case, where the benefits of explicitly controlling diversity appears only when using executions lasting several weeks when dealing with large instances of the Traveling Salesman Problem [46].

C. Analysis of the Stopping criterion

As previously discussed, EAs with explicit control of diversity are usually more useful in long-term executions. The fact that we selected a rather large stopping criterion in our first experiment might lead readers to think that VSD-MOEA is only useful in extremely long-term executions; however, this is not the case. In this section we analyze the performance of VSD-MOEA and state-of-the-art algorithms with several stopping criteria, i.e. maximum number of generations. Three different ranges were explored for the stopping criterion. Each range was split into ten equally distributed intervals, and experiments were run with each different number of generations. The ranges considered were $[250, 2500]$, $[2500, 25000]$ and $[25000, 250000]$. These ranges are referred to as short-term, middle-term and long-term executions, respectively. Note that state-of-the-art algorithms can be executed just once (with $250,000$ generations) by saving the intermediate results. However, VSD-MOEA makes decisions that depend on the stopping criteria, so independent executions were required for each stopping criterion.

Figures 6 and 7 show the mean HV ratio obtained with each MOEA with two and three objectives, respectively. All the problems were considered to calculate this mean ratio. Each figure is divided into three graphs corresponding to short-term, middle-term and long-term. In the two-objective case, for the shortest executions VSD-MOEA is not very competitive. In the range $[250, 750]$, it exhibits the worst performance, meaning that for very short-term executions, explicitly promoting additional diversity is not helpful. When using $1,000$ generations, the resulting HV ratio is similar than that obtained by other methods. Finally, when using more than $2,500$ generations, the HV ratio obtained by VSD-MOEA is much higher than the one obtained by other methods. It is worth noting that VSD-MOEA is the only method that truly takes advantage of using long-term executions, with the remaining methods just showing a slight improvement. In the three-objective case, VSD-MOEA yields a lower HV ratio than R2-EMOA and MOEA/D in short-term executions, but as more

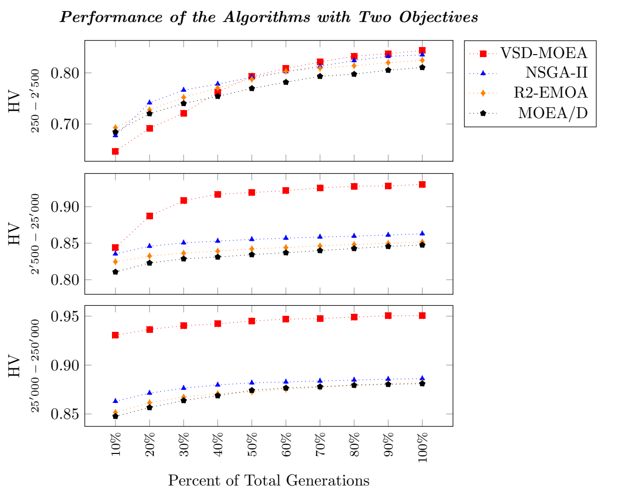

Fig. 6. Performance of MOEAs for the problems with two objectives considering three ranges for the stopping criterion: short-term (first row), middle-term (second row) and long-term (third row).

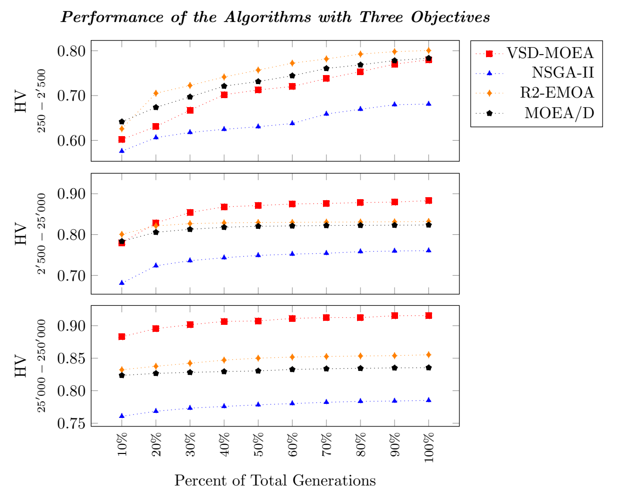

Fig. 7. Performance of MOEAs for the problems with three objectives considering three ranges for the stopping criterion: short-term (first row), middle-term (second row) and long-term (third row).

generations are executed, the differences decrease. In this case, after $5,000$ generations, the performance of VSD-MOEA is similar to that of R2-EMOA. Finally, as in the two-objective case, with more generations, the differences between VSD-MOEA and the remaining algorithms increase in favor of VSD-MOEA. Thus, while the most important benefits arise in long-term executions, users can benefit from VSD-MOEA even in shorter executions.

D. Analysis of the Initial Threshold Value

One of the disadvantages of including a strategy for controlling diversity is that this is usually done at the cost of incorporating additional parameters in the EA designed. In the case of VSD-MOEA, the initial threshold value (D_I) must be set. Note that in all the previous experiments, $D_I = 0.4$ was used. This value was selected based on some preliminary experiments. This section is devoted to analyzing the performance of VSD-MOEA when using different D_I values. Note that, since normalized distances are used, the maximum difference that can appear is 1. Additionally, note that when D_I is set to 0, no individual is penalized on the basis of its diversity contribution, so VSD-MOEA would behave like a more traditional MOEA. As a result, the values $D_I = \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ were tested. As in previous experiments, the whole set of benchmark problems was used and the stopping criterion was set to $250,000$ generations.

Figure 8 shows the mean HV ratio obtained for both the two-objective and three-objective case. Note that even when D_I is set to 0, VSD-MOEA yielded better HV ratios than other state-of-the-art algorithms (see Tables II and IV). Specifically, the values were 0.912 and 0.893 for two and three objectives, respectively. This means that the novel density estimator put forth in this paper is indeed helpful. However, the increase in performance when using other D_I values is clear. The HV ratio obtained quickly increases as higher D_I values up to 0.4 are used. Then, with values in the range $[0.5, 0.9]$, the performance decreases slightly. There is a large range of values where the performance is very good, meaning that the behavior of VSD-MOEA is quite robust. Thus, properly setting this parameter is not a complex task.

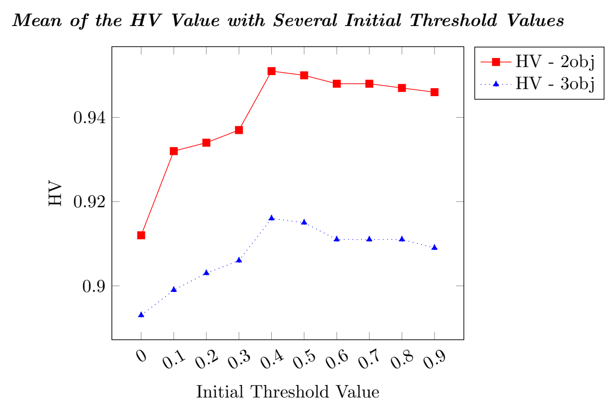

Fig. 8. Mean of HV values taking into account all the problems with several initial threshold values

V. Conclusions and Future Work

EAs have been one of the most popular approaches for dealing with complex optimization problems. Their design is a highly complex task that requires defining several components. Looking at the differences between single-objective and multi-objective optimizers, it is worth noting that several state-of-the-art single-objective optimizers explicitly consider the diversity of the variable space, especially when dealing with long-term executions, whereas this is not the case for MOEAs.

This paper proposes a novel MOEA, called VSD-MOEA, that takes into account the diversity of both the variable space and the objective space. The main novelty is that the importance given to the different diversities is adapted during the optimization process. In particular, in VSD-MOEA more importance is given to the diversity of the variable space in the initial stages, but as the generations evolve, it gradually grants more importance to the diversity of the objective space. This is performed using a penalty method that is integrated into the replacement phase. Also included is a novel density estimator based on IGD+ that is used to select from the non-penalized individuals.

The experimental validation carried out shows a remarkable improvement in VSD-MOEA when compared to state-of-the-art MOEAs both in two-objective and three-objective problems. Moreover, our proposal not only improves the state-of-the-art algorithms in long-term and medium-term executions, it also offers a competitive performance in short-term executions. The scalability analyses show that as the number of objectives and variables increases, the implicit variable space maintained by state-of-the-art MOEAs also increases. Thus, for large enough objectives and variables, explicitly considering the diversity of the variable space is not very helpful. Finally, the analysis of the initial threshold distance, which is an additional parameter required by VSD-MOEA, shows that finding a proper value for this parameter is not a difficult task.

In the future, we plan to apply the principles studied in this paper to other categories of MOEAs. For instance, including the diversity management put forth in this paper in decomposition-based and indicator-based MOEAs seems plausible. Additionally, we would like to develop an adaptive scheme to avoid setting the initial threshold value. Finally, in order to obtain even better results, these strategies are going to be incorporated into a multi-objective memetic algorithm.

ACKNOWLEDGMENTS

Authors acknowledge the financial support from CONACyT through the "Ciencia Básica" project no. 285599.

REFERENCES

[1] K. Deb and D. Kalyanmoy, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: John Wiley & Sons, Inc., 2001.

[2] F. Herrera and M. Lozano, "Adaptation of genetic algorithm parameters based on fuzzy logic controllers," in *Genetic Algorithms and Soft Computing*. Physica-Verlag, pp. 95–125.

[3] ——, "Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions," *Soft Computing*, vol. 7, no. 8, pp. 545–562, 2003. [Online]. Available: http://dx.doi.org/10.1007/s00500-002-0238-y

[4] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1998.

[5] C. Segura, C. A. C. Coello, E. Segredo, and A. H. Aguirre, "A Novel Diversity-Based Replacement Strategy for Evolutionary Algorithms," *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 3233–3246, Dec 2016.

[6] V. Koumousis and C. Katsaras, "A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 19–28, Feb 2006.

[7] C. Segura, A. Hernndez-Aguirre, F. Luna, and E. Alba, "Improving diversity in evolutionary algorithms: New best solutions for frequency assignment," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 4, pp. 539–553, Aug 2017.

[8] S. Kukkonen and J. Lampinen, "Performance assessment of generalized differential evolution 3 with a given set of constrained multi-objective test problems," in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*. IEEE, 2009, pp. 1943–1950.

[9] K. C. Tan, E. F. Khor, and T. H. Lee, *Multiobjective Evolutionary Algorithms and Applications (Advanced Information and Knowledge Processing)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[10] H. M. Pandey, A. Chaudhary, and D. Mehrotra, "A comparative review of approaches to prevent premature convergence in GA," *Applied Soft Computing*, vol. 24, pp. 1047 – 1077, 2014.

[11] M. Črepinšek, S.-H. Liu, and M. Mernik, "Exploration and Exploitation in Evolutionary Algorithms: A Survey," *ACM Computing Surveys*, vol. 45, no. 3, pp. 35:1–35:33, Jul. 2013.

[12] O. J. Mengshoel, S. F. Galán, and A. de Dios, "Adaptive generalized crowding for genetic algorithms," *Information Sciences*, vol. 258, pp. 140 – 159, 2014.

[13] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins, "A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows," *Computers & Operations Research*, vol. 40, no. 1, pp. 475 – 489, 2013.

[14] A. G. Slim Bechikh, Rituparna Datta, *Recent Advances in Evolutionary Multi-objective Optimization*. springer.

[15] S. Das and P. N. Suganthan, "Differential Evolution: A Survey of the State-of-the-Art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, Feb 2011.

[16] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr 2002.

[17] S. S. Bhagavatula, S. G. Sanjeevi, D. Kumar, and C. K. Yadav, "Multi-objective indicator based evolutionary algorithm for portfolio optimization," in *2014 IEEE International Advance Computing Conference (IACC)*, Feb 2014, pp. 1206–1210.

[18] H. Trautmann, T. Wagner, and D. Brockhoff, "R2-emoa: Focused multiobjective search using r2-indicator-based selection," in *International Conference on Learning and Intelligent Optimization*. Springer, 2013, pp. 70–74.

[19] T. C. Chiang and Y. P. Lai, "MOEA/D-AMS: Improving MOEA/D by an adaptive mating selection mechanism," in *2011 IEEE Congress of Evolutionary Computation (CEC)*, June 2011, pp. 1473–1480.

[20] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, Dec 2007.

[21] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," in *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, Jun 1994, pp. 82–87 vol.1.

[22] A. Toffolo and E. Benini, "Genetic diversity as an objective in multi-objective evolutionary algorithms," *Evolutionary computation*, vol. 11, no. 2, pp. 151–167, 2003.

[23] K. Deb and S. Tiwari, "Omni-optimizer: A procedure for single and multi-objective optimization." in *EMO*, vol. 3410. Springer, 2005, pp. 47–61.

[24] O. M. Shir, M. Preuss, B. Naujoks, and M. T. Emmerich, "Enhancing decision space diversity in evolutionary multiobjective algorithms." in *EMO*, vol. 9. Springer, 2009, pp. 95–109.

[25] K. Chan and T. Ray, "An evolutionary algorithm to maintain diversity in the parametric and the objective space," in *International Conference on Computational Robotics and Autonomous Systems (CIRAS), Centre for Intelligent Control, National University of Singapore*, 2005.

[26] T. Ulrich, J. Bader, and E. Zitzler, "Integrating decision space diversity into hypervolume-based multiobjective search," in *Proceedings of the 12th annual conference on Genetic and evolutionary computation*. ACM, 2010, pp. 455–462.

[27] H. Ishibuchi and Y. Shibata, *An Empirical Study on the Effect of Mating Restriction on the Search Ability of EMO Algorithms*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 433–447.

[28] G. Rudolph, B. Naujoks, and M. Preuss, "Capabilities of emoa to detect and preserve equivalent pareto subsets," in *Evolutionary Multi-Criterion Optimization*. Springer, 2007, pp. 36–50.

[29] R. B. Agrawal, K. Deb, K. Deb, and R. B. Agrawal, "Simulated binary crossover for continuous search space," Tech. Rep., 1994.

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

[31] K. Deb and M. Abouhawwash, "An optimality theory-based proximity measure for set-based multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 4, pp. 515–528, 2016.

[32] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, *Modified Distance Calculation in Generational Distance and Inverted Generational Distance*. Cham: Springer International Publishing, 2015, pp. 110–125. [Online]. Available: https://doi.org/10.1007/978-3-319-15892-1_8

[33] S. Huband, L. Barone, L. While, and P. Hingston, *A Scalable Multi-objective Test Problem Toolkit*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 280–295.

[34] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," *Evolutionary Multiobjec-*

tive Optimization. Theoretical Advances and Applications, pp. 105–145, 2005.

[35] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the cec 2009 special session and competition," *University of Essex, Colchester, UK and Nanyang technological University, Singapore, special session on performance assessment of multi-objective optimization algorithms, technical report*, vol. 264, 2008.

[36] jMetalcpp. [Online]. Available: http://jmetalcpp.sourceforge.net/

[37] MOEA/D. [Online]. Available: http://www3.ntu.edu.sg/home/epnsugan/index_files/CEC09-MOEA/CEC09-MOEA.htm

[38] R2-EMOA. [Online]. Available: http://inriadortmund.gforge.inria.fr/r2emoa/

[39] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of moea/d on cec09 unconstrained mop test instances," in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*. IEEE, 2009, pp. 203–208.

[40] X. Ma, Y. Qi, L. Li, F. Liu, L. Jiao, and J. Wu, "Moea/d with uniform decomposition measurement for many-objective problems," *Soft Comput.*, vol. 18, no. 12, pp. 2541–2564, Dec. 2014. [Online]. Available: http://dx.doi.org/10.1007/s00500-014-1234-8

[41] J. A. Molinet Berenguer and C. A. Coello Coello, *Evolutionary Many-Objective Optimization Based on Kuhn-Munkres' Algorithm*. Cham: Springer International Publishing, 2015, pp. 3–17. [Online]. Available: https://doi.org/10.1007/978-3-319-15892-1_1

[42] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Reference point specification in hypervolume calculation for fair comparison and efficient search," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2017, pp. 585–592.

[43] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition." *IEEE Trans. Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, 2015.

[44] J. J. Durillo, A. J. Nebro, C. A. C. Coello, J. Garcia-Nieto, F. Luna, and E. Alba, "A Study of Multiobjective Metaheuristics When Solving Parameter Scalable Problems," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 4, pp. 618–635, Aug 2010.

[45] S. Huband, L. Barone, L. While, and P. Hingston, "A scalable multi-objective test problem toolkit," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2005, pp. 280–295.

[46] C. Segura, S. Botello Rionda, A. Hernández Aguirre, and S. I. Valdez Peña, "A novel diversity-based evolutionary algorithm for the traveling salesman problem," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*. ACM, 2015, pp. 489–496.

Joel Chacón was born in Irapuato, México, in 1991 and, received his Bachelor's degree in Computer Science in the field of bioinformatic from the Instituto Tecnológico Superior de Irapuato, in 2014. In 2017 recieved the master's degree in computer science in the field of multi-objective optimization from the Center for Reseach in Mathematics, México.

He has participated in several international congress with works highly related to the diversity topics in evolutionary algorithms.

He is currently working toward the Ph.D. degree, and his current research interests are multi-objective optimization, memetic algorithms, and machine learning.

Carlos Segura (M'13) was born in Santa Cruz de Tenerife, Spain, on August 03, 1983. He received the M.S. degree in computer science from the Universidad de La Laguna, in 2009 and the Ph.D. degree in computer science from the Universidad de La Laguna, in 2012.

He has authored and co-authored over 75 technical papers and book chapters, including more than 25 journal papers. His publications currently report over 700 citations in *Google Scholar* and his h-index is 17. Currently he serves in the editorial board of several international conferences. His main research interests are: design of evolutionary algorithms, diversity management and problem solving paradigms.

Dr. Segura is a *Member* of the IEEE and a member of the ACM. He is currently an Associate Researcher of the Computer Science area at the Center for Research in Mathematics.

Carlos A. Coello Coello received PhD degree in computer science from Tulane University, USA, in 1996. He is currently Professor (CINVESTAV-3F Researcher) and Chair of the Computer Science Department of CINVESTAV-IPN, in Mexico City, México.

Dr. Coello has authored and co-authored over 450 technical papers and book chapters. He has also co-authored the book *Evolutionary Algorithms for Solving Multi-Objective Problems* (Second Edition, Springer, 2007). His publications currently report over 29,000 citations in *Google Scholar* (his h-index is 67). Currently, he is associate editor of the *IEEE Transactions on Evolutionary Computation* and serves in the editorial board of 12 other international journals. His major research interests are: evolutionary multi-objective optimization and constraint-handling techniques for evolutionary algorithms.

He received the *2007 National Research Award* from the Mexican Academy of Sciences in the area of *Exact Sciences* and the *Medal to the Scientific Merit 2009*, granted by Mexico City's congress. He is a *Fellow* of the IEEE, and a member of the ACM, Sigma Xi, and the Mexican Academy of Science.