Analysis and Enhancement of the Simulated Binary Crossover*

Joel Chacón, Carlos Segura

Center for Research in Mathematics, Jalisco S/N, Col. Valenciana CP: 36023 Guanajuato, Gto, México

joel.chacon@cimat.mx, carlos.segura@cimat.mx

***Abstract*—Most recombination operators are designed to provide an adaptive behavior with the principle of altering its exploration capabilities depending on the diversity of the population. However, depending only on the content of the population might be a drawback in long-term executions because diversity could not be large enough and the search process might prematurely converge. Based on the previous scenario, a novel recombination operator is designed for tackling continuous Multi-objective Optimization Problems (MOPs), which works effectively to enhance the search capability of Multi-Objective Evolutionary Algorithms (MOEAs). Particularly, this operator extends the Simulated Binary Crossover (SBX) by considering the stopping criterion to alter its internal operation. In order to validate the effectiveness of our proposal, it is studied by substituting the original recombination operators in three state-of-the-art MOEAs: NSGA-II, MOEA/D and SMS-EMOA. The popular DTLZ, WFG and UF benchmark problems are taken into account. Experimental validation shows a significant improvement in the performance of all the MOEAs when applying the novel crossover operator. Additionally, our proposal is also tested against state-of-the-art differential evolution operators, providing quite competitive results.**

I. Introduction

Evolutionary Algorithms (EA) have become a promising alternative in several real-world problems where the deterministic approches are not suitable. Multi-objective Optimization Problems (MOPs) involves the simultaneous optimization of two or more objective functions that are usually in conflict. A continuous minimization multi-objective problem can be defined as follows:

$$\begin{aligned} &minimize \quad F(x) = (f_1(x), f_2(x), ..., f_m(x)) \\ &subject \quad to \quad x \in \Omega \end{aligned} \tag{1}$$

where $x = (x_1, ..., x_n) \in R^n$ indicate a decision variable vector, n correspond to the number of decision variables, Ω is the feasible space, $F : \Omega \rightarrow R^m$ which consist of m objective functions and R^m is known as the *objective space*.

In a minimization MOP with m objective functions, and given two solutions x, $y \in \Omega$, x dominates y, denoted by $x \prec y$, if $f_i(x) \leq f_i(y)$ for all objectives $\{1, ..., m\}$, and $F(x) \neq F(y)$. This means that the solution x is not worse than y in any of the objectives and x is strictly better than y in at least one objective. The Pareto dominance definition states that the best solutions of a multi-objective optimization problem are those whose objective vectors are not dominated by any other feasible vector. A solution $x^* \in \Omega$ is known as Pareto optimal solution if no other solution $x \in \Omega$ dominates x^*. The Pareto set is the set of all the Pareto optimal solutions and the Pareto front are the images of the Pareto set. The goal of multi-objective optimization approaches is to obtain a proper approximation of the Pareto front. Particularly, a set of solutions that are diverse and close to the Pareto front are desired.

In the last decade several categories of Multi-Objective Evolutionary Algorithms (MOEAs) have been arising [1], [2], among them, the Non-Dominated Sorting Genetic Algortihm II (NSGA-II) [3], the MOEA based on decomposition (MOEA/D) [4], the S-metric Selection Evolutionary Multi-objective Optimization Algorithm (SMS-EMOA) [5], these representative methods are considered as the state-of-the-art.

Differential Evolution (DE) is a popular and efficient Evolutionary Algorithm (EA) which has yielded better results than Genetic Algorithms (GAs) [6], [7]. However, in the literature is well known that several componentes of DE induces an aggresive convergence leading to several drawbacks in different scenarios such as in long-term exetucions. Usually, the aggressive behaviour of DE provokes that the algorithm stagnates in some sub-optimal regions at the first stages of the execution.

To deal with these drawbacks, the most recent algorithms are designed with strategies where is considered the criteria stop, such as adaptive parameters [8], Linear Population Size Reduction (LPSR) [9] and several mutation strategies [10]. Principally, these adaptive strategies are oriented to deal with the substantial disadvantage of setting the control parameters. These are the crossover probability (CR) and the mutation factor scale (F). Also, is important to take into account the implications between single-objective and multi-objective problems which are related with the decision variable space diversity [11].

On the other hand, GAs imply a less aggressive and flexible behavior provoking lowest quality solutions than DE in short-term executions both in continuous and discrete domains [6], [7]. Nevertheless, the GAs have interesting behaviours when are considered in long-term executions. Principally, because several componentes are based in a probability density distribution which provides an extensive search process.

In this paper, a novel recombination operator, is designed for tackling continuous multi-objective optimization problems (MOPs), which works effectively to enhance the search capability, guiding to a gradual change between exploration to intensification. Therefore, several components of the Simulated Binary Crossover (SBX) are analyzed and modified, which aims to offer a suitable performance in long-term executions.

The rest of this paper is organized as follows. A brief description of the state-of-the-art MOEAs and detailed revision of the SBX operator is carried out in section II. Some empirical studies of the SBX operator are realized. Based in this stuides is proposed a dynamic variant of the SBX in section III. The experimental validation of the proposal and some DE variants are showed in Section IV. Finally, conclusions and some lines of future work are given in Section V.

II. Literature Review

This section is devoted to review some of the most important works that are highly related to the research presented in this paper. Firstly, the most important MOEAs paradigms are defined. Thereafter, some relevant classifications of crossover operators are discussed. Finally, the popular Simulated Binary Crossover (SBX) operator is explained, which is used extensively in this paper.

A. Multi-objective Evolutionary Algorithms

Through the last years has been created a large number of MOEAs that follows different design principles. In order to better classify them, several taxonomies have been proposed [12]. Attending to the principles of design, MOEAs can be based on Pareto dominance, indicators and/or decomposition [13]. Reciently, none of them have reported a clear advantage over the other ones. Particularly, the experimental validation has been carried out by including the Non-Dominated Sorting Genetic Algorithm (NSGA-II) [3], the MOEA based on Decomposition [4], and the S-Metric Selection Evolutionary Multi-objective Optimization Algorithm (SMS-EMOA) [5]. Therefore, they are representative methods of the domination-based, decomposition-based and indicator-based paradigms, respectively. The following subsections briefly describe each one of these paradigms, as well as the selected methods.

1) Domination Based MOEAs - NSGA-II: One of the most recognized paradigms are the domination based algorithms, particularly this family are based on the application of the dominance relation to design different components of the EAs. Since that the dominance relation does not inherently promotes diversity in the objective space, auxiliary techniques such as niching crowding and/or clustering are usually integrated to obtain an acceptable spread and diversity in the objective space. A critical drawback of the dominance relation is caused by the dimensionality of the objective space. In fact the selection pressure is decremented substantially as the number of objectives increase. Since this, some strategies have been developed to deal with this issue [14].

One of the most popular techniques of this group is the NSGA-II. This algorithm [3] implements a special parent selection operator. This operator is based on two mechanisms: fast-non-dominated-sort and crowding. The first one tends to provide a convergence to the Pareto front and the second one promotes the preservation of diversity in the objective space. A recent version is the NSGA-III, which is designed to deal with many-objective problems [15].

2) Decomposition Based MOEAs - MOEA/D: Decomposition-based MOEAs [4] transform a MOP in a set of single-objective optimization problems that are considered simultaneously. This transformation can be achieved through several approaches. The most popular of them is applying a weighted Tchebycheff function, therefore it is necessary provide weight vectors that are well distributed in the $m-1$ simplex that aims well-spread solutions. However, an important drawback of this kind of approaches resides in the Pareto front geometry. Particularly, the weight vectors could not be favorably to irregular Pareto front shapes.

MOEA/D [4] is a recently designed decomposition-based MOEA. It includes several principle such as problem decomposition, weighted aggregation of objectives and mating restrictions with neighborhoods definitions. Particularly, the neighborhoods are considered in the variation operators. A very used variant of the MOEA/D is the MOEA/D-DE, which use the DE operators [16] and the polynomial mutation operator [17] in the reproduction phase. Additionally, it has two extra measures for maintaining the population diversity [18].

3) Indicator Based MOEAs - SMS-EMOA: In multi-objective optimization several quality indicators have been developed to compare the performance of MOEAs. Since these indicators measure the quality of the approximations attained by MOEAs, a paradigm based on the application of these indicators was proposed. Particularly, instead of dominance concept, the indicators are used in the MOEAs to guide the optimization process. Among the different indicators, hypervolume is a widely accepted Pareto-compliance quality indicator [19]. The principal advantage of this algorithms is that the indicator usually takes into account both the quality and diversity of the solutions.

A popular and extensively used indicator-based algorithm is the SMS-EMOA [5]. This algorithm might be considered as hybrid, since it involves an indicator and dominance concepts. Essentially, it integrates the non-dominated sorting method with the use of the hypervolume metric. Thus, SMS-EMOA uses the hypervolume as a density estimator which results in a computationally extensive task. Particularly, the replacement phase erases the individual of the worst ranked front with the minimum contribution to the hypervolume. Taking into account the promising behavior of SMS-EMOA, it has been used in our experimental validation.

B. Crossover operators

The crossover operators are designed to generate offspring solutions using information of the parent solutions. They combine the feature of two or more parent solutions to form children solutions. Since several crossover operators have been proposed, some taxonomies have also been provided. The

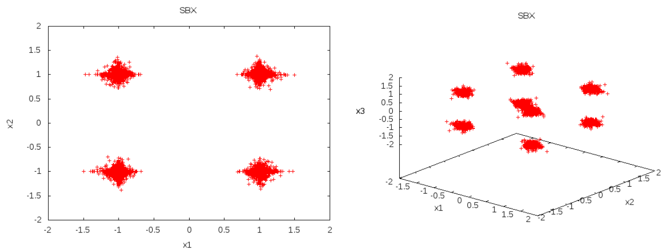

Fig. 1. Simulations of the SBX operator with a distribution index of 20, the parents are located in $P_1 = (-1.0, -1.0)$ and $P_2 = (1.0, 1.0)$ and $P_1 = (-1.0, -1.0, -1.0)$ and $P_2 = (1.0, 1.0, 1.0)$ for two and three variables respectively.

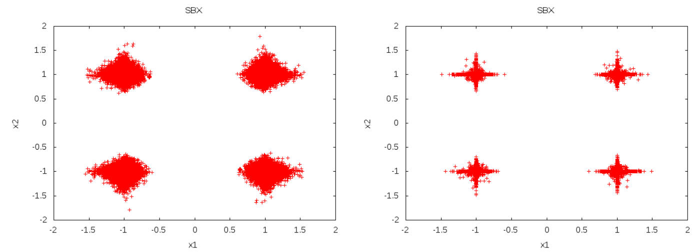

Fig. 3. Simulations of the SBX operator with a distribution index of 20, the parents are located in $P_1 = (-1.0, -1.0)$ and $P_2 = (1.0, 1.0)$. The left simulation corresponds to a probability of altering a variable (δ_1 in Algorithm 1) to 1.0 and in the right corresponds to 0.1.

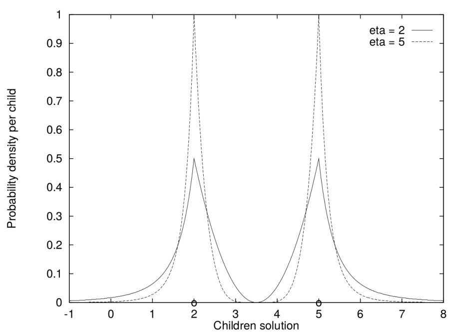

Fig. 2. Probability density function of the SBX operator with indexes of distribution 2 and 5. The parents are located in 2 and 5 respectively.

taxonomies are based on features such as the location of new generated solutions or the relation among the variables.

A popular taxonomy classifies crossover operators into variable-wise operators and vector-wise operators. In the variable wise category, each variable from parent solutions is recombined independently with a certain pre-specified probability to create new values. These operators are specially suitable to deal with separable problems. Some operators belonging to this category are the Blend Crossover (BLX) [20], and the SBX [21]. On the other hand, the vector-wise recombination operators are designed to take into account the linkage among variables. They usually perform a linear combination of the variable vectors. Some operators belonging to this category are the Unimodal Normally Distributed Crossover (UNDX) [22], and the simplex crossover (SPX) [23]. Additionally, the crossover operators can be classified as Parent-Centric and Mean-Centric [24]. In Parent-Centric operators, children solutions are created around one of the parent solutions, whereas in Mean-Centric operators, children solutions are created mostly around the mean of the participating parent solutions. Among the crossover operators, SBX is probably the most frequently used operator, so this research focuses on this crossover.

1) The Simulated Binary Crossover - SBX: The reproduction operators are one of the most relevant components that influence the search process of the GAs. Specifically, the crossover and mutation operators are highly related with the diversity of solutions. Hence, the quality solutions can be affected.

The Simulated Binary Crossover (SBX) [25] is popularly implemented in GAs [3], [5] and is classified as Parent-Centric, meaning that two children values (c_1 and c_2) are created around the parent values (p_1 and p_2). Also the process of generate the children values is based in a probability distribution. This distribution is defined by a non-dimensional variable, better known as the spread factor $\beta = |c_1 - c_2|/|p_1 - p_2|$, indicating the ratio of the spread children values related with the parent values.

Additionally, this density function uses a distribution index η_c (user-defined control parameter) that alters the exploration capability of the operator. Specifically, a small index induces a larger probability of building children values more dissimilar than parents values, whereas with a high index are generated solutions more similar to the parents as is showed in the Figure 2.

Principally, the SBX has non-zero probability of creating any number in the search space by recombining any two parent values from the search space. The probability distribution to create an offspring value is defined as a function of a non-dimensionalized parameter $\beta \in [0, \infty]$ as follows:

$$P(\beta) = \begin{cases} 0.5(\eta_c + 1)\beta^{\eta_c}, & \text{if} \quad \beta \leq 1 \\ 0.5(\eta_c + 1)\frac{1}{\beta^{\eta_c+2}}, & \text{otherwise} \end{cases} \tag{2}$$

Based in the mean-preserving property of children values and parent values, the distribution probability has the following properties:

- Both offspring values are equi-distant from parent values.
- There exist a non-zero probability to create offspring solutions in the entire feasible space from any two parent values.
- The overall probability of creating a pair offspring values within the range of parent values is identical to the overall probability of creating two offspring values outside the range of parent values.

Therefore, considering two participating parent values (p_1 and p_2), two offspring values (c_1 and c_2) can be created as linear combination of parent values with a random number $u \in [0, 1]$, as follows:

$$\begin{aligned} c_1 &= 0.5(1 + \beta(u))p_1 + 0.5(1 - \beta(u))p_2 \\ c_2 &= 0.5(1 - \beta(u))p_1 + 0.5(1 + \beta(u))p_2 \end{aligned} \tag{3}$$

Algorithm 1 Simulated Binary Crossover (SBX)

1: Input: Parents (P_1, P_2), Distribution index (η_c), Probability distribution (P_c).
2: Output: Children (C_1, C_2).
3: **if** $U[0,1] \leq P_c$ **then**
4:     **for** each variable d **do**
5:         **if** $U[0,1] \leq \delta_1$ **then**
6:             Generate $C_{1,d}$ with Equations (5) and (6).
7:             Generate $C_{2,d}$ with Equations (5) and (7).
8:             **if** $U[0,1] \geq \delta_2$ **then**
9:                 Swap $C_{1,d}$ with $C_{2,d}$.
10:         **else**
11:             $C_{1,d} = P_{1,d}$.
12:             $C_{2,d} = P_{2,d}$.
13: **else**
14:     $C_{1,d} = P_{1,d}$.
15:     $C_{2,d} = P_{2,d}$.

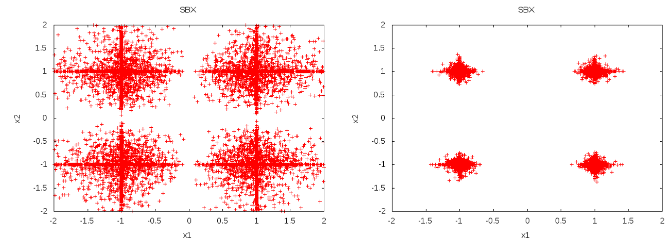

Fig. 4. Simulation of the SBX operator sampling $10,000$ children values, the parents are located in $P_1 = (-1.0, -1.0)$ and $P_2 = (1.0, 1.0)$. The left and right are with a distribution index of 2 and 20 respectively.

The parameter $\beta(u)$ depends on the random number u, as follows:

$$\beta(u) = \begin{cases} (2u)^{\frac{1}{\eta_c+1}}, & \text{if} \quad u \leq 0.5, \\ (\frac{1}{2(1-u)})^{\frac{1}{\eta_c+1}}, & \text{otherwise} \end{cases} \tag{4}$$

The above equation only considers a optimization problem having no variable bounds. In most practical problems, each variable is bounded within a lower and upper bound. Thus, Deb and Beyer in 1999 [26] proposed a modification of the probability distribution showed in the Equation (5).

$$\beta(u) = \begin{cases} (2u(1-\gamma))^{\frac{1}{\eta_c+1}}, & \text{if} \quad u \leq 0.5/(1-\gamma), \\ (\frac{1}{2(1-u(1-\gamma))})^{\frac{1}{\eta_c+1}}, & \text{otherwise} \end{cases} \tag{5}$$

$$c_1 = 0.5(1+\beta(u))p_1 + 0.5(1-\beta(u))p_2 \tag{6}$$

$$c_2 = 0.5(1+\beta(u))p_1 + 0.5(1-\beta(u))p_2 \tag{7}$$

In this fashion, the child c_1 which is nearest to p_1 is calculated according the Equation (6). Therefore, for $p_1 < p_2$ and the lower bound a is closer to p_1 than to p_2, thus $\gamma = 1/(\alpha^{\eta_c+1})$, where $\alpha = 1 + (p_1 - a)/(p_2 - p_1)$. Similarly, the second child c_2 is computed with $\alpha = 1 + (b - p_2)/(p_2 - p_1)$, where b correspond to the upper bound. Then, the second child is computed as is indicated in the Equation (7). In the literature [21] is not entirely studied the SBX extension to multi-variables problems, in fact the authors implemented a similar mechanism to the uniform crossover for multiple variables [22] for choosing which variables to cross. However those authors recognized the important implications with the linkage issues, therefore it does not alleviate the linkage problem of some MOPs.

2) Implementation and analyses of SBX operator: This section discusses the principal characteristics of SBX operator. Essentially, the behavior of this operator is directly affected by three key components. Firstly, it applies a probability of altering each variable which is fixed to 0.5, therefore in average the half of each parent is duplicate in each child. If this probability value is increased, the children values are more dissimilar, since that in average are modified more variables. An appropriate setting of this probability is related with the MOP, therefore a high probability value is better with objective functions that have high dependence level in the parameters. The behavior of variating this probability can be noticed in the Figure 3, where a low probability provokes a bias to the axis (right side), being a suitable aproach for separable problems. On the other hand, a high probability (left side) could be better in non-separable problems. Additionally, this probability value is related with the distribution index, in fact both components have a direct effect in the similarity between parents and children. The second key component resides in that two child values are interchanged given a defined probability (usually 0.5). In some contexts this probability is known as “Variable uniform crossover probability” [6] or “Discrete Recombination” [27]. Although that in single-objective this action provides an uto-adaptive behavior, in multi-objective optimization could not provide a desirable effect at first stages, the principal reason is that it could be highly disruptive. This components has serial implications, since that interchange variables between the children has the effect of multiple “reflections” in the feasible space. However, increasing the dimensions of the decision variables has the effect of increase exponentially the number of reflections ($2^n - 2$) as is showed in the Figure 1 where are considered two and three decision variables.

Finally, the last component is the distribution index, which plays an important role, since a low index results in a greater exploration levels. In fact a distribution index of the unity has the similar effect of the Fuzzy Recombination Operator [28], the behaviour of the different indexes can be analyzed in the Figure 4 where in the left is showed a simulation which consider a low index value, whereas in the right is used a high index that create similar values of the parents. Also can be noticed that it has a bias of create children according to the axis.

For simplicity the SBX implementation is showed in the Algorithm 1, which is based in the most used implementation and is integrated in the NSGA-II code published by Deb et al. [3]. It requires two parents (P_1 and P_2) and create two children (C_1 and C_2).

Principally, the first and second key components correspond to the lines 5 and 8 respectively. As is usual, the SBX is configured with $\delta_1 = \delta_2 = 0.5$ and $\eta_c = 20$. It is important take into account that this configuration neither considers the dimension of the decision variables space or the criteria stop.

TABLE I

STATISTICAL INFORMATION OF METRICS WITH TWO OBJECTIVES

	NSGA-II						MOEA/D						SMS-EMOA					
	1	2	3	4	5	DE	1	2	3	4	5	DE	1	2	3	4	5	DE
Average HV	0.88	0.90	0.90	0.91	0.93	**0.94**	0.87	0.87	0.87	0.90	**0.91**	**0.91**	0.88	0.89	0.87	0.91	0.92	**0.93**
Best Counts HV	2	1	0	1	8	**11**	2	0	2	2	8	**9**	0	1	1	5	6	**10**
Average Best Difference HV	0.068	0.057	0.053	0.039	0.019	**0.017**	0.053	0.048	0.049	0.024	**0.013**	0.014	0.074	0.064	0.081	0.045	0.028	**0.019**
Average IGD+	0.12	0.09	0.11	0.07	0.06	**0.05**	0.14	0.12	0.14	0.09	0.08	**0.07**	0.13	0.11	0.14	0.08	0.07	**0.05**
Best Counts IGD+	2	1	1	1	8	**10**	3	0	2	3	6	**9**	0	2	0	3	**9**	**9**
Average Best Difference IGD+	0.086	0.052	0.077	0.035	0.021	**0.016**	0.075	0.059	0.072	0.025	0.019	**0.008**	0.093	0.071	0.101	0.038	0.030	**0.017**

III. PROPOSAL

A. Implementing the linear dynamic behavior

Based in the previous analysis and to induce an appropiate balance between exploration and intensification, the following modifications are proposed. Firstly, the probability to independently alterate a variable (δ_1) changes among the execution, thus in the first optimization stages almost all the variables are modified or sampled by the determined distribution (Equation 5). Therefore in the last optimization stages less variables are modified. Principally, this change is based in a linear decreasing model, where initially is fixed to 1.0 and it is decreasing to the half of total generations to 0.5 which is maintained until the end of the execution. In this way a half of total executions applied the standard SBX as is showed in the Equation (8), where $T_{Elapsed}$ is the current generation (or elapsed time) and T_{End} is the total number of generations (or total time). In a similar fashion, the second change is related with the "variable uniform crossover probability" which is incremented from 0.0 to 0.5 as is indicated in the Equation (8). This modification is motivated to avoid the disruptive behavior of interchange the variables at first generations, once that the individuals converged to certain grade (to half of total generations) this probability is fixed to 0.5.

$$\delta_1 = \delta_2 = max\left(0.5, 1.0 - \frac{T_{Elapsed}}{T_{End}}\right) \tag{8}$$

Finally, the distribution index changes among the execution, where at the first stages a low distribution index is induced and it is linearly incremented closing the distribution curve, this linear increment is indicated in the Equation (9). Although, that these modifications are based in several works [29], [17] no one has considered to apply them simultenously and with a dynamic behavior.

$$\eta_c = 2 + 20 \times \left(\frac{T_{Elapsed}}{T_{End}}\right) \tag{9}$$

B. The relevance of isolated components

In this section we show the independent effect of each component previously mentioned. In general is applied the same configuration as is indicated in the section IV. Particularly, the jMetalcpp framework [30] was used to perform our executions. Taking into account the stochastic behavior of MOEAs, 35 independent executions were run. In all of them, the stopping criteria was set to $25,000$ generations and the size of the population was fixed to 100. The effect of each component is analyzed through four cases, based in the Algorithm 1 each case is described as follows:

- **Case 1**: The standard SBX operator where $\delta_1 = \delta_2 = 0.5$ and $\eta_c = 20$.
- **Case 2**: The value δ_1 changes according the Equation (8), $\delta_2 = 0.5$ and $\eta_c = 20$.
- **Case 3**: The value δ_2 changes according the Equation (8), $\delta_1 = 0.5$ and $\delta_c = 20$.
- **Case 4**: The distribution index changes according the Equation (9), $\delta_1 = \delta_2 = 0.5$.

In order to analyze the performance of each Case (Case 5 is discussed in section IV), the Tables I, II and III shows information of the Normalized Hyper-volume (HV) [29] and the Inverted Generational Distance Plus (IGD+) [19]. In the Table III is showed a summary of the statistical tests, where are considered the HV and IGD+ both in two and three objectives. Based in the statistical tests (Table III), the Case 4 which correspond to the dynamic distribution index, yield better results than Case 1, Case 2 and Case 3. Therefore, increasing the distribution index might provokes an suitable behavior to aim a balance between exploration and intensification. This occurs since that initially an open distribution curve leads to explorate more adequatly the feasible space. Hence are generated more dissimilar individuals and is induced a level of diversity in the population. At the last stages the distribution curve tends to be close, leading to an exploitation effect. In the second place is the Case 3, it increases the probability of interchange a variable based in a linear model, this might occurs since that at the first stages are avoided disruptive modifications. Therefore, the promising regions are not adequatly explored, although that this case does not yeld the best results, it still outperforms the standard SBX (Case 1). The average of HV and IGD+ for two and three objectives are showed in the Tables I and II, it shows that the Case 4 outperforms the Case 1, Case 2 and Case 3 with two and three objectives. On the other hand, just considering the averages, the Case 2 is worse than the Case 1, this might occurs because altering almost all the variables with the SBX distribution could induce very dissimilar and distant children. Perhaps, modifying this linear model (i.e. from 0.4 to 0.5) could provide a better behavior. It is important to take in consideration that the statistical tests do not qualify the results. Therefore, some MOEAs could have a high statistical score and in average a poorly performance. This occurs since that for some instances a MOEA provides high quality solutions, and for other problems are produced low quality solutions, indicating that the MOEA has stability issues.

Based in the previously analyzes, a variant of the SBX is proposed, where the Case 3 and Case 4 are mixed. Our proposal (Algorithm 1) is configured as follows. As is usual the parameter δ_1 is fixed to 0.5, following the Case 3 the δ_2 is changed according the Equation (8) and based in the Case 4 the distribution index (η_c) is changed according the Equation (9). An important resulta is that our proposal does not requires a distribution index.

IV. Experimental validation

This section is devoted to validate our proposal (Case 5). The WFG [31], DTLZ [32] and UF [18] test problems have been used for our purpose. Our experimental validation includes the Differential Evolution based variants DEMO such strategy is explained in [6]. Given that all of them are stochastic algorithms, each execution was repeated 35 times with different seeds. The common configuration in all of them was the following: the stopping criterion was setted to $25,000$ generations, the population size was fixed to 100, the WFG test problems were configured with two and three objectives, setting 24 parameters, where 20 of them are distance parameters and 4 are position parameters. Specifically, in the DTLZ test instances, the number of decision variables is set to $n = M + r - 1$, where $r = \{5, 10, 20\}$ for DTLZ1, DTLZ2 to DTLZ6 and DTLZ7 respectively, as is suggested by the authors [32]. The UF benchmark is composed of ten test instances, where the first seven are of two objectives and the rest with three objectives, the number of decision variables is assigned to $n = 10$. In general, the crossover and mutation operators are SBX and polynomial respectively, with a crossover probability of 0.9 and mutation probability of $1/n$, also the crossover and mutation distribution indexes were assigned to 20 and 50 respectively. The extra-parametrization of each algorithm is as follows:

- **DE-Variants**: CR = 0.3 and F = 0.5.
- **SMS-EMOA**: offset = 100.
- **MOEA/D**: size of neighborhood = 10, max updates by sub-problem (nr) = 2 and $\delta = 0.9$.

Our experimental analysis has been performed in base of the Normalized HV and IGD+. The reference points implemented in the hypervolume indicator are showed in the Table IV as used in [33], [34].

In order to statistically compare the hypervolume results, a similar guideline than the proposed in [35] was used. First a Shapiro-Wilk test was performed to check whatever or not 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 samples had equal variance, an ANOVA test was done; if not, a Welch test was performed. For non-Gaussian distributions, the nonparametric Kruskal-Wallis test was used to test whether samples are drawn from the same distribution. An algorithm X is said to win algorithm Y when the differences between them are statistically significant, if the mean and median obtained by X are higher than the mean and median achieved by Y.

In addition, pair-wise statistical tests with HV and IGD+ were performed (Table III). For each instance, the column "$\uparrow$" reports the number of comparisons where the statistical tests confirmed the superiority of the MOEA listed in the corresponding group, whereas the column "$\downarrow$" reports the number of cases where it was inferior and "$\longleftrightarrow$" indicates that are not significantly different and are considered as a tie.

Based in the statistical tests[1] (Table III), generally speaking our proposal (Case 5) has the best scores. Particularly, considering three objective and with the NSGA-II, the Case 4 is better than our proposal. This might occurs for the ranking mechanism of this MOEA. Since that the NSGA-II applies a binary tournament selection based in dominance concept and crowding procedure, thus an apropiate level of diversity is maintained. As result some promising regions are reached.

Although that with three objective the Case 4 and our proposal have similar average results. The average difference with the best indicates that our proposal is mostly closest to the best values. This since that it shows 0.020 in the "Average Best Difference" IGD+ against the Case 4 with 0.023. Also, inspecting the statistical tests, the Case 4 has more loses than our proposal (10 against 9 and 15 against 9) of the HV and IGD+ respectively. However, the MOEA/D and SMS-EMOA which are more elitist algorithms and do not consider the dominance concept at all, are not affected in this sense. Anyway, our proposal and the Case 3 reports significantly better results that the normal SBX (Case 1), in fact this could evidence that variate the distribution index among the execution has an important effect.

In order to better understand the behavior of our proposal, the Tables I and II shows a summary average of Normalized HV and IGD+ considering all the instances[2]. Also is counted the number of instances which attained the best results labeled by "Best Counts". Additionally, the average of difference with the best result is showed, this aims to provide a metric which indicates the difference with the best result attained in each problem.

Although that the DE variants provides better average results than our proposal considering two objective, our proposal provides near solutions to the best, showing its stability. Also according the number of objectives increase to three our proposal provides the best results. The main reason is that DE is seriously deteriorated as the number of objectives increases, also it converges in a fast way, since it incorporates an agressive selection operator.

Despite the fact that DE variants have high "Best Count" values, in average it is improved by the Case 4 and Case 5, this might occurs because DE is directly influenced by the probability crossover (CR) and the mutation factor (F), therefore in some instances the DE variants attained the best

[1]DE is not considered in the statistical tests.

[2]The detailed results can be consulted in the web page https://github.com/joelchaconcastillo/SBX_CEC2018.

TABLE II
STATISTICAL INFORMATION OF METRICS WITH THREE OBJECTIVES

	NSGA-II						MOEA/D						SMS-EMOA					
	1	2	3	4	5	DE	1	2	3	4	5	DE	1	2	3	4	5	DE
Average HV	**0.87**	0.84	**0.87**	**0.87**	**0.87**	0.85	0.84	0.84	0.84	**0.86**	**0.86**	0.85	0.90	0.89	0.88	**0.91**	**0.91**	**0.91**
Best Counts HV	1	2	1	4	4	**7**	1	2	1	2	5	**8**	3	2	0	2	5	**7**
Average Best Difference HV	0.019	0.047	0.020	**0.014**	**0.014**	0.032	0.036	0.041	0.038	0.016	**0.013**	0.027	0.038	0.038	0.049	**0.019**	0.027	**0.019**
Average IGD+	0.13	0.16	0.13	**0.12**	**0.12**	0.13	0.15	0.14	0.15	**0.11**	**0.11**	0.13	0.11	0.11	0.13	**0.09**	**0.09**	0.13
Best Counts IGD+	0	2	2	4	3	**8**	2	2	0	2	4	**9**	1	3	0	3	5	**7**
Average Best Difference IGD+	0.029	0.061	0.027	0.023	**0.020**	0.032	0.053	0.048	0.053	**0.015**	**0.015**	0.030	0.047	0.040	0.062	**0.020**	0.024	0.069

TABLE III
SUMMARY OF STATISTICAL TESTS

NSGA-II															
	1			2			3			4			5		
	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷
HV-2obj	16	29	47	6	61	25	28	19	45	31	23	38	**54**	3	35
HV-3obj	15	19	42	12	50	14	17	15	44	**33**	10	33	26	9	41
IGD-2obj	14	30	48	4	60	28	25	17	50	33	19	40	**52**	2	38
IGD-3obj	14	18	44	13	44	19	18	15	43	**33**	15	28	23	9	44
MOEA/D															
	1			2			3			4			5		
	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷
HV-2obj	15	33	44	10	60	22	25	26	41	39	18	35	**57**	9	26
HV-3obj	10	22	44	12	39	25	11	19	46	24	10	42	**38**	5	33
IGD-2obj	16	31	45	9	60	23	23	27	42	37	17	38	**57**	7	28
IGD-3obj	12	22	42	13	43	20	13	24	39	30	9	37	**40**	10	26
SMS-EMOA															
	1			2			3			4			5		
	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷	↑	↓	⟷
HV-2obj	9	35	48	7	43	42	16	31	45	41	9	42	**53**	8	31
HV-3obj	7	21	48	9	35	32	13	21	42	27	6	43	**31**	4	41
IGD-2obj	10	34	48	15	48	29	12	33	47	41	12	39	**55**	6	31
IGD-3obj	8	20	48	13	30	33	9	19	48	22	5	49	**27**	5	44

TABLE IV
REFERENCES POINTS FOR THE HV INDICATOR

Instances	**Reference Point**
WFG1-WFG9	$[2.1, ..., 2m + 0.1]$
DTLZ 1, 2, 4	$[1.1, ..., 1.1]$
DTLZ 3, 5, 6	$[3, ..., 3]$
DTLZ7	$[1.1, ..., 1.1, 2m]$
UF 1-10	$[2, ..., 2]$

results, and in other instances the solutions are far from the Pareto front.

On the other hand, our proposal shows a robust behavior in the state-of-the-art MOEAs, in fact the "Average Best Difference" is low both with two objectives and three objectives that confirms the stability and superiority of our proposal.

V. CONCLUSIONS

In the usual scheme of EAs, the crossover is one of the most important operators. In most cases, crossover operators provide a large degree of exploration when the content of the population is diverse. However, when a low diversity degree is reached, they tend to promote intensification. Since the behavior of operators does not usually depend on the stopping criterion, losing diversity in a gradual way might be a complex task. Thus, some parameterizations might be suitable for some stopping criterion but not for others. This paper proposes extending the well-known SBX to incorporate the stopping criterion and elapsed generations as one of its inputs. First, the standard version of SBX and some actions that can be done to alter its exploration capabilities are identified. Specifically, the SBX is composed by three key components that are related with diversity issues. The first controls the amount of variables that are inherited intact. The second is the probability of interchanging a given variable between offspring. Finally, the last component is related with the aperture of the distribution index. These tree features are adapted by taking into account both the stopping criterion and elapsed generations, with the aim of inducing an additional degree of exploration in the first stages of the optimization. The experimental validation is carried out with long-term executions and the popular WFG, DTLZ and UF problems. This validation shows that a dynamic distribution index provides significantly better and more robust results than the standard SBX with all the tested MOEAs. Additionally, adapting the probability of interchanging variables between offspring provides benefits and by adapting this probability simultaneously with the distribution index performance can be further improved. In the case of three objectives, state-of-the-art DE operators could also be outperformed by our proposal. No additional parameterizations were required to devise the proposed operators, meaning that robust results could be obtained with the additional user efforts.

Several lines of the future work might be explored. First, we would like to devise some strategies to adaptively manage the distribution index. Meassuring the diversity of the population to alter the behavior of the operators seems a plausible approach. Additionally, using these kinds of operators together with some of the specific methods that has been devised to avoid premature convergence might bring additional benefits. Finally, we plan to use the principles that governed the design of our proposals to devise novel vector-wise operators.

REFERENCES

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

[2] C. A. C. Coello, G. B. Lamont, and D. A. V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[3] 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.

[4] 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.

[5] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume ," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653 – 1669, 2007.

[6] T. Tušar and B. Filipič, "Differential evolution versus genetic algorithms in multiobjective optimization," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2007, pp. 257–271.

[7] M. Mi, X. Huifeng, Z. Ming, and G. Yu, "An improved differential evolution algorithm for tsp problem," in *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*, vol. 1. IEEE, 2010, pp. 544–547.

[8] J. Brest, M. S. Maučec, and B. Bošković, "Single objective real-parameter optimization: Algorithm jso," in *Evolutionary Computation (CEC), 2017 IEEE Congress on*. IEEE, 2017, pp. 1311–1318.

[9] J. Brest and M. S. Maučec, "Population size reduction for the differential evolution algorithm," *Applied Intelligence*, vol. 29, no. 3, pp. 228–247, 2008.

[10] 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, 2011.

[11] S. Kukkonen, "Performance assessment of generalized differential evolution 3 (gde3) with a given set of problems," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. IEEE, 2007, pp. 3593–3600.

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

[13] M. Pilát, "Evolutionary multiobjective optimization: A short survey of the state-of-the-art," *Proceedings of the Contributed Papers Part I-Mathematics and Computer Sciences, WDS, Prague, Czech*, pp. 1–4, 2010.

[14] C. Horoba and F. Neumann, "Benefits and drawbacks for the use of epsilon-dominance in evolutionary multi-objective optimization," in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*. ACM, 2008, pp. 641–648.

[15] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[16] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media, 2006.

[17] M. Hamdan, "The distribution index in polynomial mutation for evolutionary multiobjective optimisation algorithms: An experimental study," in *International Conference on Electronics Computer Technology, Kanyakumari, India*, 2012.

[18] 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.

[19] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, "Modified distance calculation in generational distance and inverted generational distance," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2015, pp. 110–125.

[20] L. J. Eshelman, "Real coded genetic algorithms and interval-schemata," *Foundations of genetic algorithms*, vol. 2, pp. 187–202, 1993.

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

[22] I. Ono, H. Kita, and S. Kobayashi, "Advances in evolutionary computing," A. Ghosh and S. Tsutsui, Eds. New York, NY, USA: Springer-Verlag New York, Inc., 2003, ch. A Real-coded Genetic Algorithm Using the Unimodal Normal Distribution Crossover, pp. 213–237. [Online]. Available: http://dl.acm.org/citation.cfm?id=903758.903767

[23] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *J. of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec. 1997. [Online]. Available: http://dx.doi.org/10.1023/A:1008202821328

[24] H. Jain and K. Deb, "Parent to mean-centric self-adaptation in sbx operator for real-parameter optimization," *Swarm, Evolutionary, and Memetic Computing*, pp. 299–306, 2011.

[25] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, no. 3, pp. 1–15, 1994.

[26] K. Deb and H.-G. Beyer, *Self-adaptive genetic algorithms with simulated binary crossover*. Secretary of the SFB 531, 1999.

[27] H. Mühlenbein and D. Schlierkamp-Voosen, "Predictive models for the breeder genetic algorithm i. continuous parameter optimization," *Evolutionary computation*, vol. 1, no. 1, pp. 25–49, 1993.

[28] H.-M. Voigt, H. Mühlenbein, and D. Cvetkovic, "Fuzzy recombination for the breeder genetic algorithm," in *Proc. Sixth Int. Conf. on Genetic Algorithms*. Citeseer, 1995.

[29] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[30] E. López-Camacho, M. J. García Godoy, A. J. Nebro, and J. F. Aldana-Montes, "jmetalcpp: optimizing molecular docking problems with a c++ metaheuristic framework," *Bioinformatics*, vol. 30, no. 3, 2013.

[31] 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.

[32] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, *Scalable test problems for evolutionary multiobjective optimization*. Springer, 2005.

[33] J. A. M. Berenguer and C. A. C. Coello, "Evolutionary many-objective optimization based on kuhn-munkres' algorithm," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2015, pp. 3–17.

[34] Q. Zhu, Q. Lin, Z. Du, Z. Liang, W. Wang, Z. Zhu, J. Chen, P. Huang, and Z. Ming, "A novel adaptive hybrid crossover operator for multiobjective evolutionary algorithm ," *Information Sciences*, vol. 345, pp. 177 – 198, 2016.

[35] 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.