

Hybrid Crossover Operators with Multiple Descendants for Real-Coded Genetic Algorithms: Combining Neighborhood-Based Crossover Operators

Ana M. Sánchez,^{1,*} Manuel Lozano,^{2,‡} Pedro Villar,^{1,†} Francisco Herrera^{2,§}

¹*Department of Software Engineering, University of Granada, 18071, Granada, Spain*

²*Department of Computer Science and Artificial Intelligence, University of Granada, 18071, Granada, Spain*

Most real-coded genetic algorithm research has focused on developing effective crossover operators, and as a result, many different types of crossover operators have been proposed. Some forms of crossover operators are more suitable to tackle certain problems than others, even at the different stages of the genetic process in the same problem. For this reason, techniques that combine multiple crossovers, called hybrid crossover operators, have been suggested as alternative schemes to the common practice of applying only one crossover model to all the elements in the population. On the other hand, there are operators with multiple offsprings, more than two descendants from two parents, which present a better behavior than the operators with only two descendants, and achieve a good balance between exploration and exploitation. © 2009 Wiley Periodicals, Inc.

In this paper, we propose a model for the application of the crossover operators that generate multiple descendants from two parents and select the two best offspring to replace the parents in the new population. This has been done by means of hybrid real-parameter crossover operators, which generate the multiple descendants for every pair of parents, each two offsprings via a different crossover operator. These hybrid crossover operators are based on the combination of neighborhood crossover operators that have shown a good behavior using them for generating multiple offsprings. Experimental results show that these crossover operators improve the behavior of the classic crossover operators for real-coded genetic algorithms, and it is possible to achieve a suitable balance between diversity and selective pressure that it is responsible for improving performance with respect to the use of a single crossover operator.

* Author to whom all correspondence should be addressed: amlopez@ugr.es.

† e-mail: pvillarc@ugr.es.

‡ e-mail: lozano@decsai.ugr.es.

§ e-mail: herrera@decsai.ugr.es.

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 24, 540–567 (2009)

© 2009 Wiley Periodicals, Inc. Published online in Wiley InterScience
(www.interscience.wiley.com). • DOI 10.1002/int.20348

1. INTRODUCTION

Genetic algorithms (GAs)^{1,2} are adaptive methods based on natural evolution that may be used for search and optimization problems. They process a population of search space solutions with three operations: *selection*, *crossover*, and *mutation*.

In the initial formulation of the GAs, the candidate solutions were coded using the binary alphabet; however, other coding types, such as the *real coding*, have also been taken into account to deal with the representation of the problem. The real coding approach seems particularly natural when tackling optimization problems of parameters with variables in continuous domains.³ A chromosome is a vector of floating point numbers whose size is kept the same as the length of the vector, which is the solution to the problem. GAs based on real number representation are called *real-coded GAs* (RCGAs).^{3,4}

The crossover operator is a method for sharing information between chromosomes. It has always been regarded as the main search operator in GAs^{5,6} because it exploits the available information in previous samples to influence future searches. This is why most RCGA research has been focused on developing effective real-parameter crossover operators, and as a result, many different possibilities have been proposed.^{3,4} In Ref. 7, a taxonomy is introduced to classify the crossover operators for RCGAs. It groups the models for these operators into different categories according to the way in which they generate the genes of the offspring from the genes of the parents. Specifically, neighborhood-based crossover operators (NBCOs) are a family of operators that has currently received special attention. They determine the genes of the offspring extracting values from intervals defined on neighborhoods associated with the genes of the parents, throughout probability distributions. The degree of diversity induced by these operators may be easily adjusted by means of varying an associated crossover step size parameter. The greater its value is, the higher the variance (diversity) introduced into the population.

Each crossover operator directs the search toward a different zone in the neighborhood of the parents. The quality of the elements that belong to the visited region depends on the particular problem to be solved. This means that different crossover operators perform differently with respect to different problems, even at the different stages of the genetic process in the same problem. Thus, the simultaneous application of different crossover operators on the population could provide effective models that may be suited to many practical problems. In fact, some studies have been undertaken to examine the synergy produced by combining different styles of the traversal of solution space associated with various crossover operators.⁸⁻¹⁴ Their objective was to investigate whether or not a combination of crossovers performs better than the best single crossover amongst them. In Ref. 15, the synergy derived from the combination of different crossover operators of the taxonomy presented in Ref. 7 is studied and that reveal complementary properties that are required to build an effective coupling of real-parameter crossover operators.

Usually, the crossover operator is applied to pairs of chromosomes, generating two offspring for each one of them, which are introduced in the population.¹ However, *crossover operators with multiple descendents* (CX-MDs) have been presented,¹⁶⁻²³ which produce more than two offspring for each group of

parents. In this case, an offspring selection mechanism limits the number of offspring that will be population members. These operators are inspired in the nature to obtain more benefit of the parents by the sampling of a bigger number of possible solutions resulting of their recombination. All the offspring may be created using the same offspring generation mechanism^{17,21} or by means of different offspring generation mechanisms.¹⁹ In Ref.24, an empirical study of a simple model of CX-MDs for RCGAs is presented. For each pair of parents, n_d offspring are created applying repeatedly the crossover operator to them, and the two best offspring are selected as the children contributed by the mating. Three different instances of this model were implemented by considering three well-known neighborhood-based real-parameter crossover operators, BLX- α , FR, and PNX. The final goodness of the crossover operator depends of the balance between the diversity associated with the offspring generation mechanisms and the selective pressure derived from the offspring selection mechanisms. A number of parents and descendants as well as the nature of offspring generation and selection mechanisms are fundamental factors in this balance.

The aim of this paper is to design *hybrid real-parameter crossover operators with multiple descendants* (HCX-MDs), which generate multiple offspring for every pair of parents, each two with a different crossover operator. We analyze the positive synergy among real-parameter crossover operators that belong to the group of NBCOs.

The paper is set out as follows: In Section 2, we introduce relevant issues related to the taxonomy of real-parameter crossover operators as proposed in Ref. 7 (Section 2.1), the characteristics of the multiple descendants (Section 2.2), and the hybrid crossover operators (Section 2.3). In Section 3, we design hybrid crossover operators with multiple descendants, which combine the NBCOs. In Section 4, we introduce the experimental framework (Section 4.1) and the statistical tests that we have used to compare algorithms (Section 4.2). In addition, we describe the experimental study aimed at determining the goodness associated with the hybrid crossover operators with multiple descendants (Section 4.3) and analyze the synergetic effects produced among their constituent crossover operators (Section 4.4). In Section 4.5, we compare the best operator of the previous sections with other hybrid operators, which combine the different groups of the taxonomy. In Section 5, we present some concluding remarks and summarize some possible future research areas related to this topic. Finally, two appendixes are included containing the test functions and the results of the experiments.

2. CROSSOVER OPERATORS FOR RCGAs: PRELIMINARIES

In this section, we deal with the main aspects of the crossover operators for RCGAs. In Section 2.1, we introduce a taxonomy that groups real-parameter crossover operators in different categories attending to the way they follow to generate the genes of the offspring from the genes of the parents. In Section 2.2, we outline the characteristics of the crossover operators with multiple descendants, and, finally, in Section 2.3 we point out different approaches for combining crossover features.

2.1. Taxonomy

In Ref. 7, a taxonomy is presented, which classifies the crossover operators for RCGAs (those applied only on two parents) into different groups, focusing on the features associated with the offspring generation mechanisms that are applied to the parents to obtain the offspring. This includes whether they preserve the genes of the parents in the offspring, whether the genes of the offspring are obtained from an aggregation function in which its arguments are the genes of the parents, or whether the genes in the offspring are generated from a probability distribution defined in the neighborhoods of the genes of the parents. The taxonomy includes the following groups:

- *Discrete crossover operators* (DCOs). This category groups all the crossover operators proposed for binary coding, which are directly applicable to real coding. It includes the two-point and uniform crossover operators. With these crossovers, the value of each gene in the offspring coincides with the value of this gene in one of the parents ($h_i \in \{c_i^1, c_i^2\}$), i.e., the values of the genes in the parents are not transformed numerically to obtain the values of the genes in the offspring. Geometrically, DCOs generate a corner of the hypercube defined by the component of the two parents. The effect of these operators, according to the intervals of the generation of genes, is shown in Figure 1.
- *Aggregation-based crossover operators* (ABCOs). These include operators that use an aggregation function that numerically combines the values of the genes of the parents to generate the value of the genes of the offspring. If $[a_i, b_i]$ is the action interval for the i th gene, an aggregation function, $f_i : [a_i, b_i] \rightarrow [a'_i, b'_i] ([a'_i, b'_i] \subseteq [a_i, b_i])$ should be provided. Then, the value for the i th gene of the offspring is calculated as $f_i(c_i^1, c_i^2)$. The arithmetical and geometrical crossover operators are examples of ABCOs. In the case of the arithmetical crossover, the aggregation function is a linear combination of c_i^1 and c_i^2 . Graphically, ABCOs act as shown in Figure 2. As seen in this figure the ABCOs may generate genes in the exploitation interval or in the exploration interval.
- *Neighborhood-based crossover operators* (NBCOs). This group includes crossovers that determine the genes of the offspring, extracting values from intervals defined in neighborhoods associated with the genes of the parents throughout probability distributions.



Figure 1. Effects of the DCOs.

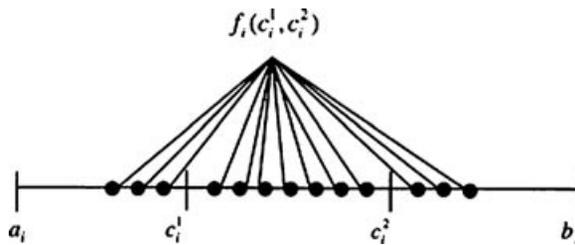


Figure 2. Possible gene values calculated by ABCOs from c_i^1 and c_i^2 .

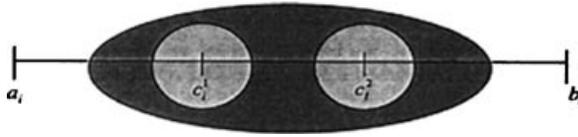


Figure 3. Neighborhoods taken into account by NBCOs.

Examples of NBCOs are BLX- α , simulated binary crossover, fuzzy recombination, and PNX, which are based on uniform, polynomial, triangular, and normal probability distributions, respectively. Figure 3 represents graphically the neighborhoods considered by NBCOs.

NBCOs with multiple descendants are crossover-based local search techniques. With the passing of generations, the RCGA loses diversity due to the selective pressure. Under this circumstance, the self-adaptive nature of NBCOs (they can generate offspring adaptively according to the distribution of parents without any adaptive parameter) allows the creation of offspring distributed densely around the parents, favoring an effective local tuning.

The main difference between ABCOs and NBCOs is that ABCOs are deterministic crossovers, i.e., given two parents, the resultant offspring shall always be the same, whereas NBCOs include a random component, i.e., they are nondeterministic.

Tables I and II contain the real-parameter crossover operators used in this paper, along with the category to which they belong: NBCOs in Table I and DCO and ABCO in Table II. These operators have been selected from the operators studied in Ref. ^{7,24} according to their good performance.

Table I. Neighborhood-based crossover operators.

Crossover operator	Taxonomy group
Blend crossover (BLX- α) ($\alpha = 0.1, 0.3, 0.5$ and 0.7) ^{25,26}	NBCO
Simulated binary crossover (SBX- η) ($\eta = 2$ and $\eta = 5$) ^{4,27}	
Fuzzy recombination crossover (FR-d) ($d = 0.5$) ²⁸	
Parent-centric normal crossover (PNX- η) ($\eta = 1, 2, 3,$ and 4) ²⁹	

Numbers in superscript indicate the references.

Table II. Discrete and aggregation crossover operators.

Crossover operator	Taxonomy group
Two-point crossover (2P) ³⁰	DCO
Arithmetical crossover (A) ($a = 0.5$) ⁵⁰	ABCO
Geometric crossover (G) ($a = 0.5$) ³¹	
Dynamic heuristic crossover (DH) ³²	

Numbers in superscript indicate the references.

2.2. Multiple Descendants

The crossover operator can provide diversity by the generation of multiple descendants. Obviously, if the number of descendants grows, the exploration on the search space will be bigger because the number of possible solutions will increase and will be located anywhere in this space. A bigger exploration helps to the diversity.

On the other hand, the generation of multiple descendants implies the use of some selection mechanism to decide which of these descendants will be population members. This selection mechanism has associated a selective pressure that can derive in premature convergence.

Figure 4 shows the idea of CX-MDs. In this case, from two parents, four crossover are applied. Each crossover generates two offspring and, finally, the two best ones are selected.

RCGAs with CX-MDs may be seen as a kind of real-coded memetic algorithm.³³ The justification is as follows: Once a standard RCGA has found fit areas of the search space, it only searches over a small fraction of the neighborhood around each search point. It must derive its power from integrating multiple single neighborhood explorations in parallel over successive generations of a population. This “many points, few neighbors” strategy is in direct contrast with a hill climber, which potentially focuses effort on a greater fraction of the search neighborhood of one point but only around one point at a time. This strategy might be called “few points,

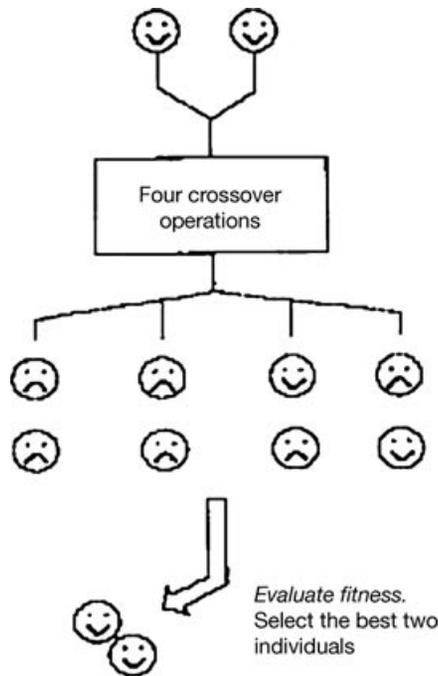


Figure 4. Multiple descendants.

many neighbours.”³⁴ Precisely, with the use of crossover operators with multiple descendants, RCGAs may follow this strategy, with the aim of inducing an effective local search on the neighborhood of the parents involved in crossover.

One of the first versions of CX-MDs is brood recombination, which was studied in genetic programming (GP). Tackett³⁵ devised this method to compensate for the highly disruptive type of crossover used in genetic programming. With this mechanism, Tackett attempted to model the observed fact that many animal species produce far more offspring than are expected to live. Although there are many different mechanisms, the excess offspring die. This is a hard but effective way to cull out the results of bad crossover.

In general, CX-MDs promote the idea that by exploring more combinations of an individual crossover operation, there is more opportunity of success. Even though with a less efficient evaluation, the diversity will grow.

In general, points of crossover are randomly selected; however, with CX-MDs, it is selected the best of a big set of random recombinations. Thus, there is a bigger probability that the offspring are better than their parents.

In the following, we revise the proposals of multiple descendants from two parents presented in the specialized literature:

- In Refs. 36 and 18, multiple descendants are generated from two parents, using different crossover operators, and introducing all the descendants in the new population. In Ref.²¹, multiple descendants are generated from two individuals selected randomly from the parents' population. Two individuals are selected among the family (the parents and their children), selecting one of them with the elitist and the other with the roulette wheel, to gather the population for the next generation.
- A generalization of this last model is proposed in Ref.⁴. The multiple descendants are generated from the best chromosome of the population and other that has been selected by tournament. Finally, the two best chromosomes of the family are selected.
- In Refs.^{37,38}, a family competition model is shown. In this model, each individual in the population sequentially becomes the “family parent.” With a probability p_c , this family father and another solution that is randomly chosen from the rest of the parent population are used as parents for a recombination operation. Then the new offspring or the family father is operated on by a mutation. For each family father, such a procedure is repeated L times. Finally, L children are produced, but only the one with the lowest objective value survives. After the family competition, there are N parents and children left. Two methods are proposed to obtain a new population with N individuals: “family selection” and “population selection.”
- In Ref. 19, the Max-min-arithmetical crossover operator generates four descendants, and the two best chromosomes are selected as final descendants for the new population.
- In Ref. 20, multiple descendants are generated from two parents and the two best offspring replace the parents in the new population. In this model, different instances are presented based on the BLX- α crossover operator for real-coded genetic algorithms.
- In Ref. 24, this idea is extended to NBCOs. It is presented an empirical study of a simple model of CX-MDs for RCGAs. For each pair of parents, n_d offspring are created, applying repeatedly the crossover operator to them, and the two best offspring are selected as the children contributed to by the mating. Three different instances of this model were implemented by considering BLX- α , FR, and PNX crossover operators. The experimental results obtained confirm that the generation of multiple descendants along with the offspring selection mechanism that chooses the two best offspring may enhance the operation of these three crossover operators.

2.3. Different Approaches for Combining Crossover Features

The idea behind crossover is that by combining features from two good parents crossover will often produce even better offspring.³⁹ However, the efficiency of crossover for genetic search is governed by the relationship between the crossover biases (its traversal style of solution space) and the search problem itself. A particular crossover operator becomes more effective when its search bias is adjusted to the structure of the problem to be solved. Thus, some forms of crossover operators are more suitable for solving certain problems than others, even at the different stages of the genetic process in the same problem. The *no free lunch* theorems confirm this fact.⁴⁰

An interesting idea to devise crossover-based techniques, which may be suited to most practical problems, would consist in the simultaneous application of diverse crossover operators on the population. In fact, some studies have been conducted in which the synergy produced by combining the different styles of the traversal of solution space associated with various crossover operators has been examined.^{8–15} Their objective was to investigate whether or not a combination of crossovers performs better than the best single crossover amongst them.

There have been different attempts to find synergetic crossover operators:

1. *Hybrid crossover operators (HCXs)*. These crossovers use different kinds of crossover operators to produce diverse offspring from the same parents. For example, in Ref. 19, an hybrid real-parameter crossover operator is presented, which generates four offspring for each pair of parents, applying two explorative crossovers and two exploitative crossovers to them. The two most promising offspring of the four substitute their parents in the population. In Ref. 15, it is made a study of the synergy derived from the combination of different crossover operators that reveal complementary properties that are required to build an effective coupling of real-parameter crossover operators. This has been done by means of HCXs, which generate two offspring for every pair of parents, each one with a different crossover operator. Experimental results show that synergy is possible among real-parameter crossover operators, and in addition, that it is responsible for improving performance with respect to the use of a single crossover operator.
2. *Gradual distributed GAs*. In Ref. 9, a distributed RCGA model maintains, in parallel, several subpopulations that are processed by independent GAs that apply different forms of crossover operators. These operators are differentiated according to their associated exploration and exploitation properties and the degree thereof. Other distributed GA models that make distinctions between the subpopulations by applying GAs with different crossovers are described in Refs. 41 and 42. In this case, each subpopulation competes with other subpopulations, in such a way that it gains or loses individuals depending on its evolution quality in relation to the others.
3. *Adaptive crossover operator probabilities*. A set of crossover operators is available, each with a probability of being used. For each reproduction event, a single operator is selected probabilistically according to the set of operator probabilities. In addition, an adaptive process dynamically adjusts the operator probabilities during the course of evolving a solution. For example, in Refs. 8 and 43, those operators that create and cause the generation of better chromosomes are given higher probabilities. Another approach for adaptation involves self-adaptation^{13,44}; operator probabilities are directly coded onto each member of the population, and this allows them to evolve, i.e., they undergo mutation and recombination. In Ref. 45 is presented a new adaptive crossover operator based on the rough set theory that uses the attribute reduction to find candidate schemata with promising performance.

Two models of adaptive real-parameter crossover operator probabilities are described in Refs. 46 and 47. In Ref. 46, two complementary crossover operators are considered, UNDX⁴⁸ and uniform crossover.⁴⁹ In Ref. 47, an RCGA applies two different crossover operators; one with exploitation properties and another with exploration properties. An operator probability parameter defines the frequency of the application of the exploitative operator.

3. HYBRID REAL-PARAMETER NEIGHBORHOOD-BASED CROSSOVER OPERATORS WITH MULTIPLE DESCENDANTS

The crossover operators that belong to the NBCO's group of taxonomy are expected to provide different traversal styles of search space. NBCOs with multiple descendants are crossover-based local search techniques. With the passing of generations, the RCGA loses diversity due to the selective pressure. Under this circumstance, the self-adaptive nature of NBCOs (they can generate offspring adaptively according to the distribution of parents without any adaptive parameter) allows the creation of offspring distributed densely around the parents, favoring an effective local tuning.

Thus, to carry out a study of the synergy among different operators, we present several proposals of *hybrid real-parameter with multiple descendants neighborhood-based crossover operators* (HCX-MDs-NBCOs). They generate multiple offspring (six or eight) for every pair of parents by applying a different crossover operator for every pair of descendants. The two most promising offspring substitute their parents in the population. In Figure 5, this idea is represented.

HCXs are a simple way of combining crossover operators and, therefore, constitute a framework that facilitates the study of the synergetic effects of different real-parameter crossover operators.

We have built two kinds of HCX-MDs-NBCOs: S-NBCO (all the descendants are generated with the same NBCO, with different values of parameter for each two offspring) and D-NBCO (a combination of different NBCOs, so that each two descendants are generated with a different NBCO), as shown in Tables III and IV (the number before the operator indicates the amount of descendants generated by it, two or four). These two types of hybridization allow us to analyze the effects derived from the union between different exploration and exploitation characteristics:

- With the first type, S-NBCO, all the descendants are obtained with the same NBCO, but with a different parameter value for each two offspring. The degree of diversity induced by these operators may be adjusted by means of varying these parameters.
- The second one, D-NBCO, combines different neighborhood crossover operators, allowing us to generate each two descendants with a different probability distribution for creating the genes of the offspring in restricted search spaces around the regions marked by the genes of the parents. We have chosen parameter values that have achieved good results in previous works.

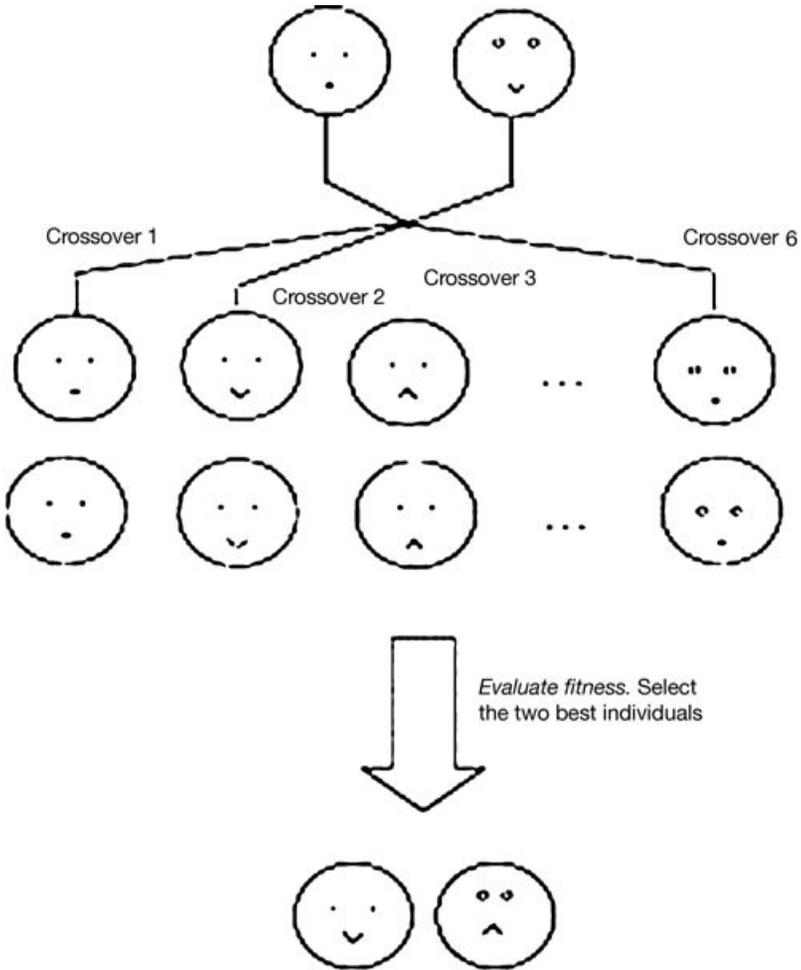


Figure 5. Hybrid crossover operators with multiple descendants.

Table III. S-NBCO hybrid real-parameter crossover operators.

2BLX0.3-4BLX0.5-2BLX0.7
2FR0.3-4FR.5-2FR0.7
2PNX2-4PNX3-2PNX4
2BLX0.1-2BLX0.3-2BLX0.5-2BLX0.7
2FR0.1-2FR0.3-2FR0.5-2FR0.7
2PNX1-2PNX2-2PNX3-2PNX4
2SBX0.01-2SBX1-2SBX2-2SBX5
2BLX0.3-2BLX0.5-2BLX0.7
2FR0.3-2FR.5-2FR0.7
2PNX2-2PNX3-2PNX4
2SBX0.01-2SBX1-2SBX2

Table IV. D-NBCO hybrid real-parameter crossover operators.

2BLX0.1-2FR0.3-2PNX4
2BLX0.5-2FR0.5-2PNX3
2BLX0.5-2FR0.5-2SBX0.01
2BLX0.5-2PNX3-2SBX0.01
2FR0.5-2PNX3-2SBX0.01
2BLX0.5-2FR0.5-2PNX3-2SBX0.01

4. EXPERIMENTAL STUDY

Minimization experiments on the test suite were carried out with the aim of determining whether differences on performance exist between classic crossover operators and the same operators, producing multiple descendants in combination with others crossover operators. This section is set out as follows:

- In Section 4.1, we describe the algorithms built to do this and the tests functions.
- In Section 4.2, we present nonparametric tests to carry out a statistical analysis of the results.
- In Section 4.3, we show the results of the hybrid crossover operators S-NBCOs and D-NBCOs and discuss some conclusions about them.
- In Section 4.4, we analyze the synergetic effects produced among the best HCX-MDs-NBCO operator and his constituent crossover operators, comparing these operators when they generate two, four, and eight offspring, with the hybrid operator.
- Finally, in Section 4.5, we compare the best HCX-MDs-NBCO operator with other HCXs-MDs operators.

4.1. Experimental Framework

For experiments, we have considered a *generational* RCGA model that applies the *nonuniform* mutation operator.⁵⁰ This operator has been widely used, reporting good results.³ The selection probability calculation follows linear ranking⁵¹ ($\eta_{\min} = 0.75$), and the sampling algorithm is the *stochastic universal sampling*.⁵² The *elitist strategy*⁵³ is considered as well, which makes sure that the best performing chromosome always survives intact from one generation to the next.

The population size is 61 individuals, the probability of updating a chromosome by mutation is 0.125, and the crossover probability is 0.6. We run all the algorithms 30 times, each one with a maximum of 100.000 evaluations.

Minimization experiments on the test suite (see Table V),^{15,24} were carried out to study the behavior of the RCGA with the HCX-MDs-NBCOs model presented in Section 3.3.

4.2. Statistical Tests

We have used four nonparametric tests, Friedman, Iman–Davenport, Holm, and Wilcoxon, to carry out a statistical analysis of the results.^{54–56}

There is no established procedure for comparing algorithms over multiple problems. Researchers adopt different statistical and commonsense techniques to

Table V. Tests functions.

Test functions	<i>n</i>	<i>f(x*)</i>
Sphere model ⁵³	25	$f_{Sph}(x^*) = 0$
Schwefel's function 1.2 ⁵⁷	25	$f_{Sch}(x^*) = 0$
Generalized Rastrigin's function ⁵⁸	25	$f_{Ras}(x^*) = 0$
Griewangk's function ⁵⁹	25	$f_{Gri}(x^*) = 0$
Expansion of F10 ⁶⁰	25	$f_{F10}(x^*) = 0$
Generalized Rosenbrock's function ⁵³	25	$f_{Ros}(x^*) = 0$
Systems of linear equations ⁶¹	10	$P_{Sle}(x^*) = 0$
Frequency modulation sounds parameter identification ⁶²	6	$P_{fms}(x^*) = 0$
Polynomial fitting problem ⁶³	9	$P_{Chev}(C^*) = 0$
Ackley's function ⁶⁴	25	$f_{Ack}(x^*) = 0$
Bohachevsky's function ⁶⁵	10	$f_{Boh}(x^*) = 0$

Numbers in superscript indicate the references.

decide whether the differences between the algorithms are real or random. Statistical analyses have been carried out to find significant differences among the results obtained by the studied methods. However, on the use of statistical tests, there are some assumptions to take into account, depending on the test applied. There are mainly two groups of statistical tests, parametric and nonparametric ones.

The most used statistical tests in comparisons are the parametric tests that use for each algorithm and function the mean error achieved from a set of executions. Using these values, these tests can indicate when a difference between two algorithms is statistically significant. An example of these tests is *t*-Student test (for comparisons one to one) and ANOVA (for multiple comparisons). To be able to apply parametric tests, results must assume normal distribution and homogeneity of variance. When these constraints are satisfied, a parametric statistical analysis of results will be right and safe, and more sensitive than nonparametric tests. But, when these conditions are not satisfied, parametric tests are not robust and nonparametric should be applied.

We will use a set of simple, safe, and robust nonparametric tests for statistical comparisons:

- *Friedman's test*.⁶⁶ It is equivalent to the repeated-measures ANOVA. Friedman assigns for function to each algorithm an order position (r_j for algorithm j with k algorithms, $r_j \in [1, \dots, k]$, 1 to the best and k to the worst value, respectively). The null hypothesis of Friedman implies that all algorithms are equivalent. If this condition is true, the Friedman's statistic value

$$X_F^2 = \frac{12N}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right]$$

follows an X_F^2 distribution with $k - 1$ degree of freedom, with $R_j = 1/N \sum_i r_i^j$, and N the number of functions.

- *Iman-Davenport's test*. It is a variation of the Friedman test. Iman and Davenport showed that Friedman's X_F^2 is undesirably conservative and derived a better

statistic

$$F_F = \frac{(N - 1) X_F^2}{N(k - 1) - X_F^2}$$

which is distributed according to the F -distribution with $k - 1$ and $(k - 1)(N - 1)$ degrees of freedom.

- *Holm's test.* This test sequentially tests the hypotheses ordered by their significance. We will denote the ordered p values by p_1, p_2, \dots , so that $p_1 \leq p_2 \leq \dots \leq p_{k-1}$. The test compares each p_i with $\alpha/(k - i)$. Holm's step-down procedure starts with the most significant p value. If p_1 is below $\alpha/(k - 1)$, the corresponding hypothesis is rejected and we are allowed to compare p_2 with $\alpha/(k - 2)$. If the second hypothesis is rejected, the test proceeds with the third, and so on. As soon as a certain null hypothesis cannot be rejected, all the remaining hypotheses are retained as well.
- *Wilcoxon's test.* This is a nonparametric alternative to the paired t -test, which ranks the differences in performances of two classifiers for each data set, ignoring the signs and compares the ranks for the positive and the negative differences. For each function, the difference is calculated between the values obtained by the algorithm to compare and the reference algorithm. The differences are ranked according to their absolute values; Average ranks are assigned in the case of ties. Let R^+ be the sum of ranks for the functions on which the second algorithm outperformed the first, and R^- is the sum of the ranks for the opposite. If the smaller of R^+ and R^- is smaller than or equal to the value of the T distribution of Wilcoxon for N degrees of freedom (Table B.12 in Ref. 56) the null hypothesis is rejected.

4.3. Analysis of the Results for Hybrid NBCOs with Multiple Descendants

In this section, we analyze the behavior of the HCX-MDs-NBCOs. First, in Section 4.3.1, we analyze D-NBCO operators (introduced in Table IV) and, second, in Section 4.3.2, we analyze S-NBCO operators (introduced in Table III). In Section 4.3.3, we compare the best crossovers obtained in the previous sections between them.

4.3.1. D-NBCO Hybrid Real-Parameter Crossover Operators

To apply the nonparametric test, for each test suite, the crossover operators have been ordered according to the algorithm behaviors in the different test suites (see Table A.3). The operator with the smallest value is the best. Then, we have obtained the rank of each crossover operator for the total of test suites. Table VI shows the average ranking of each crossover for the 11 test suites.

Friedman test compares the rank of the algorithms. Under the null hypothesis, which states that all the algorithms are equivalent and so their ranks should be equal, Friedman's statistics is distributed according to a value obtained according to the number of test suites and algorithms, with a degree of freedom equal to the number of algorithms -1 . In Table VII, we can see the results of Friedman and Iman-Davenport with $\alpha = 0.05$. Clearly, there are significant differences between them.

Holm's test, with a standard error $SE = 0.7937$, compares the algorithm with the best rank, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01, with each one of the five remaining

Table VI. Ranking of the algorithms with the different D-NBCOs.

	Rank
2BLX0.1-2FR0.3-2PNX4	4.59
2FR0.5-2PNX3-2SBX0.01	4.27
2BLX0.5-2FR0.5-2SBX0.01	4.05
2BLX0.5-2FR0.5-2PNX3	3.82
2BLX0.5-2PNX3-2SBX0.01	2.50
2BLX0.5-2FR0.5-2PNX3-2SBX0.01	1.77

Table VII. Friedman and Iman–Davenport’s test.

Friedman value	χ^2_F value	Iman-Davenport value	F_F value
17.41	11.070	4.63	2.40

Table VIII. Holm’s test with control algorithm 2BLX0.5-2FR0.5-2PNX3-2SBX0.01.

i	Algorithm	$Z = (R_0 - R_i)/SE$	P	α/i
5	2BLX0.1-2FR0.3-2PNX4	$(1.77 - 4.59)/0.7937 = 3.55$	0.0004	0.01
4	2FR0.5-2PNX3-2SBX0.01	$(1.77 - 4.27)/0.7937 = 3.14$	0.0016	0.0125
3	2BLX0.5-2FR0.5-2SBX0.01	$(1.77 - 4.05)/0.7937 = 2.87$	0.0042	0.17
2	2BLX0.5-2FR0.5-2PNX3	$(1.77 - 3.82)/0.7937 = 2.58$	0.0098	0.025
1	2BLX0.5-2PNX3-2SBX0.01	$(1.77 - 2.5)/0.7937 = 0.91$	0.3628	0.05

ones. There are significant differences with four of them because the P values are smaller than the α value. Table VIII shows the results of Holm test. The algorithms are ordered in rank from better (1) to worse (5).

As we can observe, there are not significant differences between the operators 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 and 2BLX0.5-2PNX3-2SBX0.01.

4.3.2. S-NBCO Hybrid Real-Parameter Crossover Operators

Once more, to apply the nonparametric test, for each test suite, the crossover operators have been ordered according to the average of each algorithm in the different test suites (see Table A.1). Then, we have obtained the rank of each crossover operator for the total of test suites. Table IX shows the ranking of each crossover for the test suites.

In Table X, we can see the results of Friedman and Iman–Davenport with $\alpha = 0.05$. Clearly, there are significant differences between the S-NBCOs.

Holm’s test, with a standard error $SE = 1.4142$, compares the algorithm with best rank, 2BLX0.3-4BLX0.5-2BLX0.7, with each one of the 10 remaining S-NBCOs. There are significant differences with four of them because the P values are smaller than the α value. Table XI shows the results of Holm test. The algorithms are ordered in rank from better (1) to worse (10).

Because Holm’s test shows enough equality between seven S-NBCO operators, in the next section we will compare these operators with the best one of the previous section.

Table IX. Ranking of the algorithms with the different S-NBCOs.

	Average rank
2SBX0.01-2SBX1-2SBX2	8.82
2SBX0.01-2SBX1-2SBX2-2SBX5	8.18
2PNX2-2PNX3-2PNX4	7.45
2PNX1-2PNX2-2PNX3-2PNX4	7.36
2PNX2-4PNX3-2PNX4	6.55
2BLX0.1-2BLX0.3-2BLX0.5-2BLX0.7	5.45
2FR0.3-4FR.5-2FR0.7	5.09
2FR0.1-2FR0.3-2FR.5-2FR0.7	5.00
2FR0.3-2FR.5-2FR0.7	4.91
2BLX0.3-2BLX0.5-2BLX0.7	3.73
2BLX0.3-4BLX0.5-2BLX0.7	3.45

Table X. Friedman and Iman–Davenport’s test.

Friedman value	X_F^2 value	Iman–Davenport value	F_F value
31.81	18.307	4.06	1.93

Table XI. Holm’s test with control algorithm 2BLX0.3-4BLX0.5-2BLX0.7.

i	Algorithm	$Z = (R_0 - R_i/SE)$	P	α/i
10	2SBX0.01-2SBX1-2SBX2	$(3.45 - 8.82)/1.4142 = 3.79$	0.0002	0.005
9	2SBX0.01-2SBX1-2SBX2-2SBX5	$(3.45 - 8.18)/1.4142 = 3.34$	0.0008	0.0055
8	2PNX2-2PNX3-2PNX4	$(3.45 - 7.45)/1.4142 = 2.82$	0.0048	0.00625
7	2PNX1-2PNX2-2PNX3-2PNX4	$(3.45 - 7.36)/1.4142 = 2.76$	0.0058	0.07
6	2PNX2-4PNX3-2PNX4	$(3.45 - 6.55)/1.4142 = 2.19$	0.0286	0.08
5	2BLX0.1-2BLX0.3-2BLX0.5-2BLX0.7	$(3.45 - 5.45)/1.4142 = 1.41$	0.1616	0.01
4	2FR0.3-4FR.5-2FR0.7	$(3.45 - 5.09)/1.4142 = 1.15$	0.2502	0.0125
3	2FR0.1-2FR0.3-2FR.5-2FR0.7	$(3.45 - 5.0)/1.4142 = 1.09$	0.2758	0.17
2	2FR0.3-2FR.5-2FR0.7	$(3.45 - 4.91)/1.4142 = 1.03$	0.3030	0.0125
1	2BLX0.3-2BLX0.5-2BLX0.7	$(3.45 - 3.73)/1.4142 = 0.19$	0.8494	0.05

4.3.3. Comparison between D-NCBOs and S-NBCOs

With the aim of obtaining the best HCX-MDs-NBCO operators, we have compared the best operators of Section 4.3.1 and 4.3.2. Because in Section 4.3.1, the two best operators have similar behavior, we have chosen only one of them, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 operator, for the study because it presented the best rank. Table XII shows the ranking of each crossover for the 11 test suites.

In Table XIII, we can see the results of Friedman and Iman–Davenport with $\alpha = 0.05$. Clearly, there are significant differences between the eight best crossover operators of the two previous sections.

Holm’s test, with a standard error $SE = 1.0219$, compares the algorithm with best rank, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01, with each one of the seven remaining operators. There are significant differences with all of them. Table XIV shows

Table XII. Ranking of the algorithms with different crossover operators.

	Average rank
2PNX2-4PNX3-2PNX4	6.09
2BLX0.1-2BLX0.3-2BLX0.5-2BLX0.7	5.55
2FR0.3-2FR.5-2FR0.7	4.82
2FR0.1-2FR0.3-2FR.5-2FR0.7	4.73
2FR0.3-4FR.5-2FR0.7	4.64
2BLX0.3-2BLX0.5-2BLX0.7	4.36
2BLX0.3-4BLX0.5-2BLX0.7	4.00
2BLX0.5-2FR0.5-2PNX3-2SBX0.01	1.82

Table XIII. Friedman and Iman–Davenport’s test.

Friedman value	X_F^2 value	Iman–Davenport value	F_F value
20.76	15.507	3.69	2.14

Table XIV. Holm’s test with control algorithm 2BLX0.5-2FR0.5-2PNX3-2SBX0.01.

i	Algorithm	$Z = (R_0 - R_i/SE)$	P	α/i
7	2PNX2-4PNX3-2PNX4	$(1.82 - 6.09)/1.0219 = 4.17$	<0.0002	0.07
6	2BLX0.1-2BLX0.3-2BLX0.5-2BLX0.7	$(1.82 - 5.55)/1.0219 = 3.65$	0.0002	0.08
5	2FR0.3-2FR.5-2FR0.7	$(1.82 - 4.82)/1.0219 = 2.93$	0.0034	0.01
4	2FR0.1-2FR0.3-2FR.5-2FR0.7	$(1.82 - 4.73)/1.0219 = 2.84$	0.0046	0.0125
3	2FR0.3-4FR.5-2FR0.7	$(1.82 - 4.64)/1.0219 = 2.75$	0.0060	0.017
2	2BLX0.3-2BLX0.5-2BLX0.7	$(1.82 - 4.36)/1.0219 = 2.48$	0.0132	0.025
1	2BLX0.3-4BLX0.5-2BLX0.7	$(1.82 - 4.00)/1.0219 = 2.13$	0.0332	0.05

the results of Holm’s test with the algorithms ordered in rank from better (1) to worse (7).

According to these results, we can conclude that the crossover operators, which generate multiple descendants by the combination of different NBCOs, show a better behavior than the crossover operators that generate all the descendants with the same operator even if with different parameters.

4.4. Synergy of 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 with Their Components

With the aim to study the synergy between the crossover operators, we have compared the best HCX-MDs-NBCO, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01, with each one of the four classic operators that compose it (BLX0.5, FR0.5, PNX3, SBX0.01).

For that purpose, in Section 4.4.1, we compare this operator with each one of the classic operators when they generate two offspring, and in Section 4.4.2, with all the combinations of two crossover (of the four classic operators) generating four

descendants, two for each operator. In Section 4.4.3, we compare the HCX-MDs-NBCO with each one of its components when they generate eight descendants in a homogeneous way, that is to say, all the descendants are generated with the same operator.

4.4.1. Analysis of 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 versus the Classic Simple Operators

Table XV shows the ranking of each crossover for the 11 test suites (see Table A.5). The HCX-MDs-NBCO operator has the best rank.

In Table XVI, we can see the results of Friedman and Iman–Davenport with $\alpha = 0.05$. Clearly, there are significant differences between the five crossover operators.

Holm’s test, with a standard error $SE = 0.6708$, shows that 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 is better statistically, although this operator does not present significant differences between FR-0.5 and $\alpha = 0.05$ (it shows differences with $\alpha = 0.1$). Table XVII shows the results of Holm’s test with the algorithms ordered in rank from better (1) to worse (4).

We have applied the Wilcoxon’s test, Table XVIII, to compare the 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 operator with the 2FR-0.5 operator. The value obtained, 6, is smaller than the critical value for $\alpha = 0.05$, 11. Then, we can conclude that HCX-MDs-NBCOs is better than the classic operator.

Table XV. Ranking of the algorithms with different crossover operators.

	Average rank
2SBX0.01	4.55
2PNX3	3.64
2BLX0.5	3.45
2FR0.5	2.27
2BLX0.5-2FR0.5-2PNX3-2SBX0.01	1.09

Table XVI. Friedman and Iman–Davenport’s test.

Friedman value	X_F^2 value	Iman–Davenport value	F_F value
31.66	11.070	25.65	5.72

Table XVII. Holm’s test with control algorithm 2BLX0.5-2FR0.5-2PNX3-2SBX0.01.

i		$Z = (R_0 - R_i) / SE$	P	α/i
4	2SBX0.01	$(1,09 - 4,55)/0.6708 = 5,15$	<0.0002	0.0125
3	2PNX3	$(1,09 - 3,64)/0.6708 = 3.80$	0.0002	0.17
2	2BLX0.5	$(1,09 - 3,45)/0.6708 = 3.51$	0.0004	0.025
1	2FR0.5	$(1,09 - 2,27)/0.6708 = 1.75$	0.0802	0.05

Table XVIII. Wilcoxon’s test comparing 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 and 2FR0.5.

	R^+	R^-	Hyp.
2FR0.5	6	60	Reject

4.4.2. *Analysis of 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 versus Its Components Generating Four Descendants*

In this section, we have compared 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 operator with each one of the possible combinations of its components, in couples, to generate four descendants. Table XIX shows the ranking of each crossover for the 11 test suites (see Table A.2). The HCX-MDs-NBCO operator has the best rank.

In Table XX, we can see the results of Friedman and Iman–Davenport with $\alpha = 0.05$. Clearly, there are significant differences between 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 and the combinations in couples of its components.

Holm’s test, with a standard error $SE = 0.9210$, compares the algorithm with best rank, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01, with each one of the possible combinations of its components, in couples. There are significant differences between all of them. Table XXI shows the results of Holm’s test with the algorithms ordered in rank from better (1) to worse (6).

Table XIX. Ranking of the algorithms with different crossover operators.

	Average rank
2PNX3-2SBX0.01	5.00
2BLX0.5-2PNX3	4.00
2BLX0.5-2SBX0.01	4.00
2FR0.5-2PNX3	3.91
2BLX0.5-2FR0.5	3.64
2FR0.5-2SBX0.01	3.55
2BLX0.5-2FR0.5-2PNX3-2SBX0.01	1.18

Table XX. Friedman and Iman–Davenport’s test.

Friedman value	X_F^2 value	Iman–Davenport value	F_F value
31.66	11.070	9.21	2.25

Table XXI. Holm’s test with control algorithm 2BLX0.5-2FR0.5-2PNX3-2SBX0.01.

<i>i</i>		$Z = (R_0 - R_i) / SE$	<i>P</i>	α/i
6	2PNX3-2SBX0.01	$(1.18 - 5) / 0.9210 = 4.1476$	<0.0022	0.08
5	2BLX0.5-2PNX3	$(1.18 - 4) / 0.9210 = 3.0618$	0.0022	0.01
4	2BLX0.5-2SBX0.01	$(1.18 - 4) / 0.9210 = 3.0618$	0.0022	0.0125
3	2FR0.5-2PNX3	$(1.18 - 3.91) / 0.9210 = 2.9641$	0.0030	0.17
2	2BLX0.5-2FR0.5	$(1.18 - 3.64) / 0.9210 = 2.6710$	0.0076	0.025
1	2FR0.5-2SBX0.01	$(1.18 - 3.55) / 0.9210 = 2.5732$	0.0102	0.05

4.4.3. Analysis of 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 versus Homogeneous Operators with Eight Descendants

In this section, we have compared 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 with its components when they generate eight descendants. Table XXII shows the ranking of each crossover for the 11 test suites (see Table A.6).

In Table XXIII, we can see the results of Friedman and Iman–Davenport with $\alpha = 0.05$. Clearly, there are significant differences between the five operators.

Holm’s test, with a standard error $SE = 0.6708$, compares the algorithm with the best rank, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01, with each one of homogeneous operators. There are significant differences between all of them. Table XXIV shows the results of Holm test with the algorithms ordered in rank from better (1) to worse (4).

Then, we can conclude that in the generation of multiple descendants, it is interesting to use HCXs that combine the features of the classic operators to improve their efficiency.

Table XXII. Ranking of the algorithms with different crossover operators.

	Average rank
8BLX0.5	2.64
8FR0.5	3.18
8PNX3	4.00
8SBX0.01	3.91
2BLX0.5-2FR0.5-2PNX3-2SBX0.01	1.27

Table XXIII. Friedman and Iman–Davenport’s test.

Friedman value	X_F^2 value	Iman–Davenport value	F_F value
21.92	11.070	6.62	2.61

Table XXIV. Holm’s test with control algorithm 2BLX0.5-2FR0.5-2PNX3-2SBX0.01.

<i>i</i>		$Z = (R_0 - R_i)/SE$	<i>P</i>	α/i
4	8PNX3	$(1,27 - 4)/0.6708 = 4,06$	<0.0002	0.0125
3	8SBX0.01	$(1,27 - 3,91)/0.6708 = 3.93$	<0.0002	0.17
2	8FR0.5	$(1,27 - 3,18)/0.6708 = 2.84$	0.0046	0.025
1	8BLX0.5	$(1,27 - 2,64)/0.6708 = 2.04$	0.0414	0.05

4.5. Analysis with Other HCX-MDs

In this section, we extend our study to other HCX-MDs operators. We have chosen the most relevant operators of each group of the taxonomy presented in Section 2.1 to group them to form different HCXs, generating six and eight descendants. We apply operators that propitiate diversity at the gene level (NBCOs) and operators that introduce diversity at the chromosome level (DCOs) at the same time, and this diversity is complemented with the powerful exploitation of ABCOs (HCX-MDS-ABCOs-NBCOs-DCOs) (see Section 2.1).

We try to decide whether differences on performance exist between the best HCX-MDs-NBCOs operator and HCX-MDs-ABCOs-NBCOs-DCOs operators. For that purpose, we compare the best operator achieved in the previous sections with the new operators.

Table XXV shows the ranking of the best HCX-MDs-NBCOs operator and HCX-MDs-ABCOs-NBCOs-DCOs operators for the 11 test suites (see Table A.4).

In Table XXVI, we can see the results of Friedman and Iman–Davenport with $\alpha = 0.05$. Clearly, there are significant differences between the 13 operators.

Table XXV. Ranking of the algorithms with different HCX-MDs.

	Average rank
2DINH-2(2PTOS)-2FR0.5	10.95
2DINH-2 (2PTOS) 2FR 0.5-2BLX05	9.64
2FR 0.5-2BLX 0.5-A0.5-G0.5	9.27
2(2PTOS)-2FR0.5-2BLX05-A0.5-G0.5	8.18
2DINH-2FR 0.5-2BLX05	7.55
2DINH-2FR-2BLX05-2SBX0.01	7.55
2(2PTOS)-2BLX05-2SBX0.01-A0.5-G0.5	6.68
2(2PTOS)-2FR 0.5-2BLX05	6.55
2(2PTOS)-2FR0.5-2PNX3	6.50
2(2PTOS)-2FR0.5-2BLX0.5-2PNX3	5.55
2(2PTOS)-2BLX0.5-2SBX0.01-2PNX3	5.36
2(2PTOS)-2FR0.5-2BLX05-2SBX0.01	5.09
2BLX0.5-2FR0.5-2PNX3-2SBX0.01	2.14

Table XXVI. Friedman and Iman–Davenport’s test.

Friedman value	X_F^2 value	Iman–Davenport value	F_F value
45.30	21.02	5.22	1.83

Table XXVII. Holm’s test with control algorithm 2BLX0.5-2FR0.5-2PNX3-2SBX0.01.

i		$Z = (R_0 - R_i / SE)$	P	α/i
13	2DINH-2(2PTOS)-2FR0.5	$(2.14 - 10.95)/1.6605 = 5.30$	<0.0002	0.0041
11	2DINH-2 (2PTOS) 2FR 0.5-2BLX05	$(2.14 - 9.64)/1.6605 = 4.51$	<0.0002	0.0045
10	2FR 0.5-2BLX 0.5-A0.5-G0.5	$(2.14 - 9.27)/1.6605 = 4.29$	<0.0002	0.005
9	2(2PTOS)-2FR0.5-2BLX05-A0.5-G0.5	$(2.14 - 8.18)/1.6605 = 3.63$	0.0002	0.0055
8	2DINH-2FR 0.5-2BLX05	$(2.14 - 7.55)/1.6605 = 3.25$	0.0012	0.00625
7	2DINH-2FR-2BLX05-2SBX0.01	$(2.14 - 7.55)/1.6605 = 3.25$	0.0012	0.07
6	2(2PTOS)-2BLX05-2SBX0.01-A0.5-G0.5	$(2.14 - 6.68)/1.6605 = 2.73$	0.0064	0.08
5	2(2PTOS)-2FR 0.5-2BLX05	$(2.14 - 6.55)/1.6605 = 2.65$	0.0080	0.01
4	2(2PTOS)-2FR0.5-2PNX3	$(2.14 - 6.50)/1.6605 = 2.62$	0.0088	0.0125
3	2(2PTOS)-2FR0.5-2BLX0.5-2PNX3	$(2.14 - 5.55)/1.6605 = 2.05$	0.04024	0.17
2	2(2PTOS)-2BLX0.5-2SBX0.01-2PNX3	$(2.14 - 5.36)/1.6605 = 1.93$	0.0536	0.025
1	2(2PTOS)-2FR0.5-2BLX05-2SBX0.01	$(2.14 - 5.09)/1.6605 = 1.77$	0.0768	0.05

Holm’s test, with a standard error $SE = 1.6605$, compares the algorithm with the best rank, 2BLX0.5-2FR0.5-2PNX3-2SBX0.01, with each one of the HCX-MDs-ABCOS-NBCOS-DCOS operators. Table XXVII shows the results of Holm test with the algorithms ordered in rank from better (1) to worse (13). As we can observe, the HCX-MDs-NBCO operators are similar to a pair of HCX-MDs-ABCOS-NBCOS-DCOS operators with $\alpha = 0.05$ (but it is the best with $\alpha = 0.1$).

We have also applied Wilcoxon’s test to determine which of them presents the best behavior. Tables XXVIII and XXIX show the results. The values that we

Table XXVIII. Wilcoxon’s test comparing 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 and 2(2PTOS)-2BLX0.5-2SBX0.01-2PNX3.

	R^+	R^-	Hyp.
2(2PTOS)-2BLX0.5-2SBX0.01-2PNX3	10.5	55.5	Reject

Table XXIX. Wilcoxon’s test comparing 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 and 2(2PTOS)-2FR0.5-2BLX05-2SBX0.01.

	R^+	R^-	Hyp.
2(2PTOS)-2FR0.5-2BLX05-2SBX0.01	9.5	56.5	Reject

have obtained, 10.5 and 9.5, are smaller than the critical value for $\alpha = 0.05$, 11. Then, we can conclude that 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 is better than the combination of ABCO-NBCO-DCO operators.

5. CONCLUSIONS

This paper proposes a model of HCX-MDs-NBCOs operator. This kind of crossover operator generates eight offspring for every pair of parents by applying a different NBCO for every pair of descendants. The two most promising offspring substitute their parents in the population. The main conclusions achieved are

- The generation of multiple descendants provides a bigger exploration of the search space. This fact affects the increase of the diversity in balance with the selective pressure of the selection of the two best descendants.
- The combination of the different NBCOs with different probability distributions to create the genes of the offspring, provides a bigger diversity because the different operator components of the HCX contribute to complementary features.
- The operator with best behavior is the operator that generates eight descendants using the four NBCOs : 2BLX0.5-2FR0.5-2PNX3-2SBX0.01. This operator has been compared with each one of the four classic operators that compose it (BLX0.5, FR0.5, PNX3, SBX0.01) when they generate two offspring, and with all the combinations of two crossover (of the four classic operators) generating four descendants, two for each operator. In all cases, we show that a positive synergy is caused between the components of the best HCX-MDs-NBCO operator. We have also compared the best HCX-MDs-NBCO operator with other HCX-MDs operators of other groups of the taxonomy, and clearly, the NBCOs combination is better than the ABCO-NBCO-DCO combination.

Then, we can conclude that the operator 2BLX0.5-2FR0.5-2PNX3-2SBX0.01 improves the behavior of the RCGA with other crossover operators. The combination of different NBCOs with positive synergy, and the generation of multiple descendants, helps to the diversity.

References

1. Goldberg DE. Genetic algorithms in search, optimization, and machine learning. Reading, MA: Addison-Wesley; 1989.
2. Holland JH. Adaptation in natural and artificial systems. Ann Arbor: University of Michigan Press; 1975. London: MIT Press; 1992.
3. Herrera F, Lozano M, Verdegay JL. Tackling real-coded genetic algorithms: operators and tools for behavioural analysis. *Artif Intell Rev* 1998;12(4):265–319.
4. Deb K. Multi-objective optimization using evolutionary algorithms. Chichester, UK: Wiley; 2001.
5. De Jong KA, Spears WM. A formal analysis of the role of multi-point crossover in genetic algorithms. *Ann Math Artif Intell* 1992;5(1):1–26.
6. Kita H. A comparison study of self-adaptation in evolution strategies and real-coded genetic algorithms. *Evol Comput* 2001;9(2):223–241.
7. Herrera F, Lozano M, Sánchez AM. A taxonomy for the crossover operator for real-coded genetic algorithms. An experimental study. *Int J Intell Syst* 2003;18(3):309–338.

8. Davis L. Adapting operator probabilities in genetic algorithms. In: Schaffer JD, editor. Proc Third Int Conf Genetic Algorithms. San Mateo, CA: Morgan-Kaufmann; 1989. pp 61–69.
9. Herrera F, Lozano M. Gradual distributed real-coded genetic algorithms. IEEE Trans Evol Comput 2000;4(1):43–63.
10. Hong I, Kahng AB, Moon BR. Exploiting synergies of multiple crossovers: initial studies. In: Proc Second IEEE Conf Evolutionary Computation. Piscataway, NJ: IEEE Press; 1995. pp 245–250.
11. Hong T-P, Wang H-S. Automatically adjusting crossover ratios of multiple crossover operators. J Inf Sci Eng 1998;14(2):369–390.
12. Hong T-P, Wang H-S, Lin W-Y, Lee W-Y. Evolution of appropriate crossover and mutation operators in a genetic process. App Intell 2002;16:7–17.
13. Spears WM. Adapting crossover in a genetic algorithm. In: McDonnell JR, Reynolds RG, Fogel D, editors. Proc Fourth Annual Conf Evolutionary Programming. MIT Press; 1995. pp 367–384.
14. Yoon H-S, Moon B-R. An empirical study on the synergy of multiple crossover operators. IEEE Trans Evol Comput 2002;6(2):212–223.
15. Herrera, F, Lozano, M, Sánchez AM. Hybrid crossover operators for real-coded genetic algorithms: an experimental study. Soft Comput 2005;9(4):280–298.
16. Altenberg L. The evolution of evolvability in genetic programming. In: Kinnear KE, editor. Advances in genetic programming. Cambridge, MA: MIT Press; 1994. pp 47–74.
17. Deb K, Anand A, Joshi D. A computationally efficient evolutionary algorithm for real-parameter evolution. Evol Comput J 2002;10(4):371–395.
18. Esquivel S, Leiva A, Gallard R. Multiple crossover per couple in genetic algorithms. In: Proc 4th IEEE Int Conf Evolutionary Computation (ICEC'97). Piscataway, NJ: IEEE Press; 1997. pp 103–106.
19. Herrera F, Lozano M, Verdegay JL. Fuzzy connectives based crossover operators to model genetic algorithms population diversity. Fuzzy Sets Syst 1997;92(1):21–30.
20. Herrera F, Lozano M, Pérez E, Sánchez AM, Villar P. Multiple crossover per couple with selection of the two best offspring: an experimental study with the blx- α crossover operator for real-coded genetic algorithms. In: Garijo FJ, Riquelme JC, Toro M, editors. IBERAMIA 2002. Lecture Notes in Artificial Intelligence, Vol 2527, Berlin: Springer-Verlag; 2002. pp 392–401.
21. Satoh H, Yamamura M, Kobayashi S. Minimal generation gap model for gas considering both exploration and exploitation. In: Proc Methodologies for the Conception, Design and Application of Intelligent Systems (IIZUKA'96), 1996. pp 494–497.
22. Tackett WA. Greedy recombination and genetic search on the space of computer programs. In: Foundations of genetic algorithms 3. San Francisco, CA: Morgan Kaufmann; 1994. pp 271–297.
23. Wright A. Genetic algorithms for real parameter optimization. In: Rawlin GJE, editor. Foundations of genetic algorithms 1. San Mateo, CA: Morgan Kaufmann; 1991. pp 205–218.
24. Sánchez AM, Lozano M, García-Martínez C, Molina D, Herrera F. Real-parameter crossover operators with multiple descendents: an experimental study. Int J Intell Syst 2008;23(2):246–268.
25. Bremermann HJ, Rogson M, Salaff S. Global properties of evolution processes. In: Pattee HH, Edlsack EA, Frin L, Callahan AB, editors. Natural automata and useful simulations. Washington, DC: Spartan; 1966. pp 3–41.
26. Eshelman LJ, Schaffer JD. Real-coded genetic algorithms and interval-schemata. In: Whitley LD, editor. Foundations of genetic algorithms 2, San Mateo, CA: Morgan Kaufmann; 1993. pp 187–202.
27. Deb K, Agrawal RB. Simulated binary crossover for continuous search space. Complex Syst 1995;9:115–148.

28. Voigt HM, Mühlenbein H, Cvetkovic D. Fuzzy recombination for the breeder genetic algorithm. In: Eshelman L, editor. Proc Sixth Int Conf Genetic Algorithms. San Mateo, CA: Morgan Kaufmann; 1995. pp 104–111.
29. Ballester PJ, Carter JN. An effective real-parameter genetic algorithm with parent centric normal crossover for multimodal optimisation. In: Proc of the Genetic and Evolutionary Computation Conference 2004. LNCS, Vol 3102. Berlin: Springer-Verlag; 2004. pp 901–913.
30. Eshelman LJ, Caruana A, Schaffer JD. Biases in the crossover landscape. In: Schaffer JD, editor. Proc Third Int Conf Genetic Algorithms. San Mateo, CA: Morgan Kaufmann; 1989. pp 86–91.
31. Michalewicz Z, Nazhiyath G, Michalewicz M. A note on usefulness of geometrical crossover for numerical optimization problems. In: Proc 5th Annual Conf Evolutionary Programming. Cambridge, MA: MIT Press; 1996. pp 305–312.
32. Herrera F, Lozano M, Verdegay JL. Dynamic and heuristic fuzzy connectives based crossover operators for controlling the diversity and convergence of real-coded genetic algorithms. *Int J Intell Syst* 1996;11:1013–1041.
33. Lozano M, Herrera F, Krasnogor N, Molina D. Real-coded memetic algorithms with crossover hill-climbing. *Evol Comput* 2004;12(3):273–302.
34. O'Reilly UM, Oppacher F. Hybridized crossover-based search techniques for program discovery. In: IEEE Int Conf Evolutionary Computation, 1995. Piscataway, NJ: IEEE Press; 1995. pp 573–578.
35. Tackett WA. Recombination, selection, and the genetic construction of computer programs. PhD thesis, University of Southern California, Department of Electrical Engineering Systems, 1994.
36. Esquivel S, Gallard R, Michalewicz Z. MPC: Another approach to crossover in genetic algorithms. In: Actas del Primer Congreso de Ciencias de la Computación, 1995; pp 141–150.
37. Yang J-M, Kao C-Y. Integrating adaptive mutations and family competition into genetic algorithms as function optimizer. *Soft Comput* 2000;4:89–102.
38. Yang J-M, Kao C-Y. Flexible ligand docking using a robust evolutionary algorithm. *J Comput Chem* 2000;21(11):988–998.
39. Eshelman LJ, Schaffer JD. Productive recombination and propagating and preserving schemata. In: Whitley LD, Vose MD, editors. Foundations of genetic algorithms 3. San Mateo, CA: Morgan Kaufmann; 1995. pp 299–313.
40. Wolpert DH, Macready WG. No free lunch theorems for optimization. *IEEE Trans Evol Comput* 1997;1(1):67–82.
41. Schlierkamp-Voosen D, Mühlenbein H. Strategy adaptation by competing subpopulations. In: Davidor Y, Schwefel H-P, Männer R, editors. Parallel problem solving from nature 3. Berlin: Springer-Verlag; 1994. pp 199–208.
42. Eiben AE, Sprinkhuizen-Kuyper IG, Thijssen BA. Competing crossovers in an adaptive GA framework. In: Proc IEEE Conf. Evolutionary Computation. Piscataway, NJ: IEEE Press; 1998. pp 787–792.
43. Julstrom BA. What have you done for me lately? Adapting operator probabilities in a steady-state genetic algorithm. In: Eshelman L, editor. Proc Sixth Int Conf Genetic Algorithms. San Francisco, CA: Morgan Kaufmann; 1995. pp 81–87.
44. Tuson AL, Ross P. Adapting operator settings in genetic algorithms. *Evol Comput* 1998;6(2):161–184.
45. Li F, Liu Q-H, Min F, Yang G-W. A new adaptive crossover operator for the preservation of useful schemata. In: ICMLC 2005. LNAI, Vol 3930. Berlin: Springer-Verlag; 2006. pp 507–516.

46. Ono I, Kita H, Kobayashi S. A robust real-coded genetic algorithm using unimodal normal distribution crossover augmented by uniform crossover: effects of self-adaptation of crossover probabilities. In: Proc Genetic and Evolutionary Computation Conference (GECCO-1999). San Mateo, CA: Morgan Kaufmann; 1999. pp 496–503.
47. Herrera F, Lozano M. Adaptation of genetic algorithm parameters based on fuzzy logic controllers. In: Herrera F, Verdegay JL, editors. Genetic algorithms and soft computing. Heidelberg, Germany: Physica-Verlag; 1996. pp 95–125.
48. Ono I, Kobayashi S. A real-coded genetic algorithm for function optimization using unimodal normal distribution crossover. In: Bäck T, editor. Proc Seventh Int Conf Genetic Algorithms. San Mateo, CA: Morgan Kaufmann; 1997. pp 246–253.
49. Syswerda G. Uniform crossover in genetic algorithms. In: Schaffer JD, editor. Proc Third Int Conf Genetic Algorithm. San Mateo, CA: Morgan Kaufmann; 1989. pp 2–9.
50. Michalewicz Z. Genetic algorithms + data structures = evolutionary programs. Berlin: Springer-Verlag; 1992.
51. Baker JE. Adaptive selection methods for genetic algorithms. In: Grefenstette JJ, editor. Proceedings of the first international conference on genetic algorithms and their applications. Hillsdale, MA: Lawrence Erlbaum; 1985. pp 101–111.
52. Baker JE. Reducing bias and inefficiency in the selection algorithm. In: Grefenstette JJ, editor. Proceedings of the second international conference on genetic algorithms and their applications. Hillsdale, NJ: Lawrence Erlbaum; 1987. pp 14–21.
53. De Jong KA. An analysis of the behavior of a class of genetic adaptive systems. Doctoral dissertation, University of Michigan, Ann Arbor, 1975.
54. Demsar J. Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 2006;7:1–30.
55. Sheskin DJ. Handbook of parametric and nonparametric statistical procedures. Boca Raton, FL: CRC Press; 2000.
56. Zar JH. Biostatistical analysis. Upper Saddle River, NJ: Prentice Hall; 1999.
57. Schwefel H-P. Numerical optimization of computer models. Chichester, UK: Wiley; 1981.
58. Törn A, Antanas Z. Global optimization. In: Lecture notes in computer science, Vol 350. Berlin: Springer-Verlag; 1989. pp 217–252.
59. Griewangk AO. Generalized descent of global optimization. *J Optim Theory Appl* 1981;34:11–39.
60. Whitley D, Beveridge R, Graves C, Mathias K. Test driving three 1995 genetic algorithms: new test functions and geometric matching. *J Heuristics* 1995;1:77–104.
61. Eshelman LJ, Mathias KE, Schaffer JD. Convergence controlled variation. In: Belew R, Vose M, editors. Foundations of genetic algorithms 4. San Mateo, CA: Morgan Kaufmann; 1997. pp 203–224.
62. Tsutsui S, Fujimoto Y. Forking genetic algorithm with blocking and shrinking modes. In: Forrest S, editor. Proc Fifth Int Conf Genetic Algorithms. San Mateo, CA: Morgan Kaufmann; 1993. pp 206–213.
63. Storn R, Price K. Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012. International Computer Science Institute, Berkeley, CA, 1995.
64. Ackley DH. A connectionist machine for genetic hill climbing. Boston, MA: Kluwer; 1987.
65. Reynolds RC, Chung C. Knowledge-based self-adaptation in evolutionary programming using cultural algorithms. In: Proc 1997 Int Conf Evolutionary Computation. Piscataway, NJ: IEEE Press; 1997. pp 71–76.
66. Friedman M. A comparison of alternative tests of significance for the problem of m ranking. *Ann Math Stat* 1940;11:86–92.

Appendix: Results of the Experiments

Table A.1. S-NBCO hybrid real-parameter crossover operators.

n_d	<i>Sph</i>	<i>Sch</i>	<i>Ras</i>	<i>Gri</i>	<i>EF10</i>	<i>SLE</i>	<i>Ros</i>	<i>PPF</i>	<i>FMS</i>	<i>Ack</i>	<i>Boh</i>	
	2BLX0.3-4BLX0.5-2BLX0.7	6.09e-16	8.22e-01	2.68e+00	2.46e-04	1.59e-02	2.51e+01	2.10e+01	2.28e+02	7.54e+00	1.31e-07	0.00e+00
	2FR0.3-4FR.5-2FR0.7	9.50e-17	1.52e+00	1.02e+00	4.86e+00	6.50e-03	6.00e+01	2.23e+01	4.29e+02	4.86e+00	4.68e-08	0.00e+00
	2PNX2-4PNX3-2PNX4	6.79e-12	1.39e+00	2.13e+01	3.10e+00	4.40e-01	3.81e+01	2.60e+01	2.47e+02	3.10e+00	1.23e-05	1.73e-14
	2BLX0.1-2BLX0.3-2BLX0.5-2BLX0.7	1.02e-15	5.26e+00	9.94e-01	3.77e-03	1.26e-02	4.25e+01	2.14e+01	3.22e+02	8.69e+00	1.61e-07	0.00e+00
	2FR0.1-2FR0.3-2FR0.5-2FR0.7	1.16e-17	4.79e+00	1.12e+00	4.93e-04	2.22e-03	7.58e+01	2.95e+01	6.24e+02	4.54e+00	1.64e-08	0.00e+00
	2PNX1-2PNX2-2PNX3-2PNX4	7.48e-11	4.94e+00	3.06e+01	1.36e-02	8.66e-01	2.77e+01	2.21e+01	4.02e+02	1.23e+00	3.88e-05	3.35e-14
	2SBX0.01-2SBX1-2SBX2-2SBX5	4.89e-07	1.50e+02	7.06e+00	1.11e-02	9.05e+00	8.97e+01	3.06e+01	2.08e+02	1.23e+01	2.24e-03	0.00e+00
	2BLX0.3-2BLX0.5-2BLX0.7	7.21e-16	1.29e+00	3.74e+00	5.44e-13	1.68e-02	2.61e+01	2.09e+01	1.64e+02	8.18e+00	1.32e-07	0.00e+00
	2FR0.3-2FR.5-2FR0.7	1.29e-16	1.77e+00	1.55e+00	1.15e-03	8.18e-03	5.36e+01	2.37e+01	4.88e+02	3.97e+00	5.93e-08	0.00e+00
	2PNX2-2PNX3-2PNX4	8.01e-12	1.35e+00	2.45e+01	1.66e-02	4.87e-01	3.19e+01	2.62e+01	4.86e+02	5.41e+00	1.51e-05	9.87e-15
	2SBX0.01-2SBX1-2SBX2	1.79e-09	7.04e+01	5.60e+00	1.12e-02	4.91e+00	5.99e+01	3.07e+01	3.13e+02	1.14e+01	2.57e-04	9.84e-19

Table A.2. Hybrid NBCOs with four descendants.

n_d	<i>Sph</i>	<i>Sch</i>	<i>Ras</i>	<i>Gri</i>	<i>EF10</i>	<i>SLE</i>	<i>Ros</i>	<i>PPF</i>	<i>FMS</i>	<i>Ack</i>	<i>Boh</i>	
	2FR0.5-2PNX3	1.22e-14	4.73e-01	9.35e+00	5.98e-03	5.07e-02	3.59e+01	2.64e+01	3.90e+02	5.95e+00	5.65e-07	0.00e+00
	2FR0.5-2SBX0.01	9.25e-14	8.05e-01	4.94e+00	8.94e-03	5.35e-02	1.94e+01	2.05e+01	9.06e+01	9.29e+00	1.38e-06	0.00e+00
	2PNX3-2SBX0.01	4.15e-12	9.09e-01	8.92e+00	9.66e-03	3.94e-01	2.06e+01	2.04e+01	9.56e+01	1.11e+01	9.88e-06	0.00e+00
	2BLX0.5-2FR0.5	3.33e-15	1.77e+00	6.99e+00	1.15e-03	2.18e-02	3.77e+01	2.08e+01	3.47e+02	9.31e+00	2.88e-07	0.00e+00
	2BLX0.5-2PNX3	7.27e-01	1.35e+01	5.41e-03	1.10e-01	1.91e+01	2.25e+01	3.76e+02	5.85e+00	1.40e-06	0.00e+00	7.06e-14
	2BLX0.5-2SBX0.01	1.11e-13	3.34e-01	6.26e+00	6.15e-03	7.33e-02	1.40e+01	2.08e+01	1.50e+02	1.22e+01	1.63e-06	0.00e+00

Table A.3. D-NBCO hybrid real-parameter crossover operators.

n_d	<i>Sph</i>	<i>Sch</i>	<i>Ras</i>	<i>Gri</i>	<i>EF10</i>	<i>SLE</i>	<i>Ros</i>	<i>PFP</i>	<i>FMS</i>	<i>Ack</i>	<i>Boh</i>
	2BLX0.1-2FR0.3-2PNX4	3.92e+01	7.95e-01	8.03e-03	1.01e-02	6.45e+01	2.34e+01	4.31e+02	9.76e+00	1.50e-07	0.00e+00
	2BLX0.5-2FR0.5-2PNX3	3.11e-01	6.59e+00	1.06e-03	2.00e-02	4.44e+01	2.06e+01	3.36e+02	5.16e+00	1.90e-07	0.00e+00
	2BLX0.5-2FR0.5-2SBX0.01	2.25e-15	9.70e-01	4.10e-03	1.69e-02	2.63e+01	2.07e+01	1.13e+02	8.01e+00	2.11e-07	0.00e+00
	2BLX0.5-2PNX3-2SBX0.01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	1.15e+00	2.10e+01	3.76e+01	1.91e+01	0.00e+00	0.00e+00
	2FR0.5-2PNX3-2SBX0.01	8.50e-15	8.80e-01	5.17e+00	6.32e-03	3.75e-02	2.47e+01	1.32e+02	7.08e+00	5.11e-07	0.00e+00
	2BLX0.5-2FR0.5-2PNX3-2SBX0.01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	1.40e+00	2.04e+01	5.29e+01	8.41e+00	0.00e+00	0.00e+00

Table A.4. Hybrid real-parameter crossover operators.

n_d	<i>Sph</i>	<i>Sch</i>	<i>Ras</i>	<i>Gri</i>	<i>EF10</i>	<i>SLE</i>	<i>Ros</i>	<i>PFP</i>	<i>FMS</i>	<i>Ack</i>	<i>Boh</i>
	2DINH-2FR 0.5-2BLX05	1.80e+01	6.96e-01	6.14e-03	3.11e-03	7.03e+01	2.16e+01	4.30e+02	1.16e+01	6.42e-09	0.00e+00
	2DINH-2(2PTOS)-2FR0.5	1.73e-12	7.31e+01	2.15e-02	3.43e-01	9.09e+01	2.19e+01	1.06e+03	1.27e+01	2.24e-06	0.00e+00
	2(2PTOS)-2FR 0.5-2BLX05	7.20e-21	3.36e+00	9.94e-01	2.05e-03	8.41e+01	2.47e+01	6.30e+02	2.13e+00	2.63e-10	0.00e+00
	2FR 0.5-2BLX 0.5-A0.5-G0.5	5.65e-13	2.74e+01	2.75e+00	3.77e-03	4.29e+01	2.18e+01	3.32e+02	1.26e+01	3.56e-06	0.00e+00
	2(2PTOS)-2FR0.5-2PNX3	6.42e-21	1.84e+00	8.29e-01	5.66e-03	8.60e+01	2.39e+01	5.93e+02	1.46e+00	2.21e-10	0.00e+00
	2DINH-2FR-2BLX05-2SBX0.01	4.40e-16	5.42e+00	1.39e+00	1.17e-02	2.94e+01	2.11e+01	1.82e+02	1.15e+01	6.06e-08	0.00e+00
	2DINH-2(2PTOS)	5.70e-14	3.87e+01	3.97e-01	2.12e-02	7.87e+01	2.16e+01	7.01e+02	1.24e+01	7.84e-07	0.00e+00
	2FR 0.5-2BLX05	1.68e-19	8.54e-01	6.30e-01	9.10e-03	8.49e-05	3.48e+01	8.85e+01	5.73e+00	2.12e-09	0.00e+00
	2(2PTOS)-2FR0.5-2BLX05-2SBX0.01	2.98e-13	2.27e+00	3.97e-01	4.92e-03	1.97e-02	3.05e+01	1.49e+02	9.03e+00	1.36e-06	0.00e+00
	2(2PTOS)-2FR0.5-2BLX0.5-2SBX0.01-A0.5-G0.5	3.34e-21	5.01e-01	1.32e+00	2.78e-03	3.49e-05	4.71e+01	4.43e+02	3.09e+00	2.10e-10	0.00e+00
	2(2PTOS)-2FR0.5-2BLX0.5-2PNX3	3.17e-18	6.85e-01	6.96e-01	1.37e-02	6.01e-04	3.43e+01	1.04e+02	3.04e+00	6.45e-09	0.00e+00
	2(2PTOS)-2FR0.5-2BLX05-A0.5-G0.5	1.90e-14	3.37e+01	3.31e-01	5.66e-03	6.22e-04	9.33e+01	5.17e+02	6.33e+00	2.34e-07	0.00e+00

Table A.5. NBCOs with two descendants.

n_d	<i>Sph</i>	<i>Sch</i>	<i>Ras</i>	<i>Gri</i>	<i>EF10</i>	<i>SLE</i>	<i>Ros</i>	<i>PPF</i>	<i>FMS</i>	<i>Ack</i>	<i>Boh</i>	
	2BLX0.5	6.31e-06	1.36e+03	8.72e+01	5.29e-01	1.47e+01	2.62e+01	2.61e+01	3.16e+02	1.50e+01	1.02e-02	7.84e-12
	2FR0.5	1.30e-11	8.97e+00	1.96e+01	7.71e-03	2.45e-01	2.66e+01	2.54e+01	4.51e+02	7.30e+00	1.81e-05	7.33e-14
	2PNX3	3.50e+01	2.52e+02	4.37e+04	6.99e-02	1.00e+01	5.63e+01	3.50e+01	5.75e+02	1.38e+01	1.59e-03	1.29e-11
	2SBX0.01	1.23e+01	5.90e+03	1.23e+02	4.44e+01	1.35e+02	8.93e+02	4.22e+03	0.00e+00	2.41e+01	1.36e+01	6.67e+00

Table A.6. Homogeneous NBCOS with eight descendants.

n_d	<i>Sph</i>	<i>Sch</i>	<i>Ras</i>	<i>Gri</i>	<i>EF10</i>	<i>SLE</i>	<i>Ros</i>	<i>PPF</i>	<i>FMS</i>	<i>Ack</i>	<i>Boh</i>	
	2BLX0.5-2BLX0.5-2BLX0.5-	2.27e-15	5.04e-01	3.64e+00	5.75e-04	2.16e-02	1.83e+01	2.09e+01	2.23e+02	9.65e+00	2.38e-07	0.00e+00
	2BLX0.5											
	2FR0.5-2FR0.5-2FR0.5-	4.70e-18	5.14e+02	1.26e+00	1.19e-02	9.97e-04	1.91e+02	5.02e+01	3.05e+03	5.55e+00	1.12e-08	0.00e+00
	2FR0.5											
	2PNX3-2PNX3-2PNX3-2PNX3	4.30e-12	1.68e+00	2.04e+01	1.59e-02	4.14e-01	3.43e+01	2.49e+01	3.54e+02	6.98e+00	1.00e-05	1.12e-14
	2SBX0.01-2SBX0.01-2SBX0.01-	2.23e-08	4.35e+01	6.99e+00	1.00e-02	6.56e+00	2.51e+01	2.33e+01	1.82e+02	2.25e+01	1.34e-03	4.97e-15
	2SBX0.01											