Fuzzy connectives based crossover operators to model genetic algorithms population diversity[1]

F. Herrera*, M. Lozano, J.L. Verdegay

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

Abstract

Genetic algorithms are adaptive methods which may be used to solve search and optimization problems. Genetic algorithms process a population of search space solutions with three operations: selection, crossover and mutation. An important problem in the use of genetic algorithms is the premature convergence in a local optimum. Their main causes are the lack of diversity in the population and the disproportionate relationship between exploitation and exploration. The crossover operator is considered one of the most determinant elements for solving this problem.

In this paper, we present new crossover operators based on fuzzy connectives for real-coded genetic algorithms. These operators are designed to avoid the premature convergence problem. To do so, they should keep the right exploitation/exploration balance to suitably model the diversity of the population. © 1997 Elsevier Science B.V.

Keywords: Genetic algorithms; Premature convergence; Fuzzy connectives

1. Introduction

Genetic algorithms (GA) are search algorithms that use operations found in natural genetics to guide the trek through a search space. GA are theoretically and empirically proven to provide robust search capabilities in complex spaces, offering a valid approach to problems requiring efficient and effective search [11, 7, 14].

GA process a population of individuals, which represent search space solutions, employing three operators: selection, crossover and mutation. The selection operator is formulated following Darwin's principle of survival of the fittest, whereas the crossover

and mutation operators have been inspired by the mechanisms of gene mutation and chromosome recombination found in biological genetics. Their computational role is to introduce diversity into the population probing into new regions unexplored by the selection operator. Fixed-length and binary encoded strings (chromosomes) for the representation of solutions have dominated GA research, since there are theoretical results that show them to be the most appropriate, and they are amenable to simple implementation. But the GA's good properties do not stem from the use of bit strings [2, 19]. For this reason, the path has been lain towards the use of alphabets with a higher cardinal, followed by the development of new genetic operators (crossover and mutation) on these alphabets. Non-binary codings include real number representation, which would seem

---

* Corresponding author. E-mail: herrera@robinson.ugr.es.

[35] L. Wei-Yi, Extending the relational model to deal with fuzzy values, *Fuzzy Sets and Systems* **60** (1993) 207–212.
[36] A. Yahya and L. Henschen, Deduction in non-Horn databases, *J. Automat. Reason.* **1** (1985) 141–160.
[37] L.Y. Yuan and D.A. Chiang, A sound and complete query evaluation algorithm for relational databases with null values, in: *Proc. ACM-SIGMOD Internat. Conf. on Management of Data*, Chicago (1988) 74–81.
[38] L.A. Zadeh, Fuzzy sets, *Inform. and Control* **8** (1965) 338–353.
[39] L.A. Zadeh, Fuzzy sets as a basis for theory of possibility, *Fuzzy Sets and Systems* **1**(1) (1978) 3–28.
[40] L.A. Zadeh, PRUF – a meaning representation language for natural language, *Internat. J. Man Machine Studies* **10** (1978) 395–460.
[41] M. Zemankova and A. Kandel, *Fuzzy Relational Data Bases – a Key to Expert Systems* (TUV Rheinland, Koln, 1984).
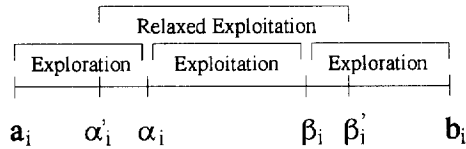[42] M. Zemankova and A. Kandel, Implementing imprecise in information systems, *Inform. Sci.* **37** (1985) 107–141.

Fig. 1. Action interval for a gene.

Table 1
Intervals properties

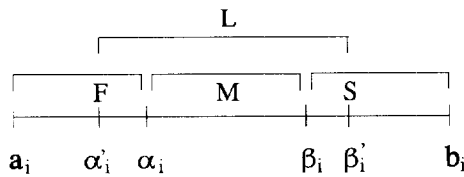| Interval	Property
$[a_i, \alpha_i]$	Exploration
$[\alpha_i, \beta_i]$	Exploitation
$[\beta_i, b_i]$	Exploration
$[\alpha_i', \beta_i']$	Relaxed exploitation

Fig. 2. Genes generated using F, S, M and L.

With regard to these intervals, we propose four functions: F, S, M and L defined from $[a,b] \times [a,b]$ into $[a,b]$, a, $b \in \mathbb{R}$, and which fulfil:

(P1) $\forall c, c' \in [a,b]$, $F(c,c') \leqslant \min\{c,c'\}$,

(P2) $\forall c, c' \in [a,b]$, $S(c,c') \geqslant \max\{c,c'\}$,

(P3) $\forall c, c' \in [a,b]$, $\min\{c,c'\} \leqslant M(c,c')$
$\leqslant \max\{c,c'\}$,

(P4) $\forall c, c' \in [a,b]$, $F(c,c') \leqslant L(c,c') \leqslant S(c,c')$,

(P5) F, S, M, and L are monotone and non-decreasing.

Each one of these functions allows us to combine two genes giving results belonging to each one of the aforementioned intervals. This is shown in Fig. 2, where $\alpha_i' = F(\alpha_i, \beta_i)$ and $\beta_i' = S(\alpha_i, \beta_i)$.

These functions will have different exploration or exploitation properties depending on their range. Although the only zone that seems natural to be considered for obtaining offspring is the exploitation zone, the other two zones may be shown to be appropriate for introducing diversity in the population. The four functions presented allow all these zones to be covered.

2.1.1. F, S, M and L functions using fuzzy connectives

Using t-norms, t-conorms, averaging functions and generalized compensation operators [15, 16], we shall associate to F a t-norm, to S a t-conorm, to M an averaging operator and to L a generalized compensation operator. In order to do so, we need to transform the genes, that will be combined, there from the interval $[a,b]$ into $[0,1]$ and, later, the results into $[a,b]$.

Complying with a set of fuzzy connectives, $\{T, G, P, \hat{C}\}$, a set of functions $\{F, S, M, L\}$, associated with it, is built as described below: If $c, c' \in [a,b]$ then

$$F(c,c') = a + (b-a) \cdot T(s,s'),$$
$$S(c,c') = a + (b-a) \cdot G(s,s'),$$
$$M(c,c') = a + (b-a) \cdot P(s,s'),$$
$$L(c,c') = a + (b-a) \cdot \hat{C}(s,s'),$$

where $s = (c-a)/(b-a)$ and $s' = (c'-a)/(b-a)$.

These operators have the properties of being continuous and non-decreasing, and satisfy the respective properties (P1)–(P5).

In this paper we shall use the families of fuzzy connectives shown in Table 2.

These fuzzy connectives fulfill the following property:

(P6) $T_4 \leqslant T_3 \leqslant T_2 \leqslant T_1 \leqslant P_j$ $(j = 1, \ldots, 4)$
$\leqslant G_1 \leqslant G_2 \leqslant G_3 \leqslant G_4$.

2.2. FCB-crossovers: F-crossover, S-crossover, M-crossover and L-crossover

As follows, we shall present the use of the F, S, M and L functions as crossover operators for RCGA.

Let us assume that $Q \in \{F, S, M, L\}$ and $C_1 = (c_1^1 \ldots c_n^1)$ and $C_2 = (c_1^2 \ldots c_n^2)$ are two chromosomes that have been selected to apply the crossover operator to them. We can generate the chromosome $H = (h_1 \ldots h_n)$ as

$$h_i = Q(c_i^1, c_i^2), \quad i = 1, \ldots, n.$$

This operator applies the same F, S, M or L function for all the genes in the chromosomes to crossover. For this reason, they will be called F-crossover, S-crossover, M-crossover and L-crossover

particularly natural when tackling optimization problems of parameters with variables in continuous domains. So 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. GA with this type of coding shall be called real-coded GA (RCGA).

An important problem in the use of GA is premature convergence; the search becomes trapped in a local optimum before the global optimum is found. Premature convergence can be blamed on [18]: the loss of critical alleles due to selection, the schemata disruption due to crossover, and the parameter setting such as mutation rate, crossover rate and population size. Mainly, all these issues produce two effects which cause the problem [12]: the lack of diversity in the population and a disproportionate exploitation/exploration relationship; e.g., adequate balance between a broad search and a sufficient refinement is not established.

In order to solve the problem, some tools are needed to monitor the genetic process. In fact, there are several strategies for maintaining population diversity and a good exploitation/exploration relationship, such as modified selection and crossover operators and optimization of control parameters studies. A review of these solutions can be found in [18].

The crossover operator plays a central rule in the GA. It exploits the available information from the population about the search space. It could be considered to be one of the algorithm's defining characteristics, and it is one of the components to be borne in mind to improve the GA behaviour [13]. In [9] it was pointed out that the crossover operator is a key point for solving the premature convergence problem. Numerous investigations have been attempted to find optimal crossover rates and alternate more powerful crossover (see [18]). Thus, solutions to this problem can be found by designing new alternatives to this operator, which should allow suitable levels of exploration and exploitation to be established.

In this paper, in order to deal with this problem, we present the fuzzy connectives based crossovers (FCB-crossovers), new crossover operators for RCGA based on the use of fuzzy connectives: t-norms, t-conorms, average functions and generalized compensation operators [15, 16]. Furthermore, we propose a set of offspring selection mechanisms (OSM) which choose the

chromosomes (produced by the crossover) that will be members of the population. These crossover operators allow us to introduce different exploration and exploitation degrees and the OSM establish a relationship between these properties so that they induce different diversity levels in the population and therefore the premature convergence problem may be eradicated. Experiments have been carried out in order to study the efficiency of the proposed crossovers in comparison with other crossover operators proposed in the literature.

2. Fuzzy connectives based crossover operators for RCGA

Next, we present new crossover operators for RCGA which allow us to establish different exploration and exploitation degrees and OSM inducing different diversity levels in the population. In order to do so, we need to develop the following steps.

1. Define genes combination functions.

2. Use these functions to define crossover operators between two chromosomes.

3. Apply the crossover operators to the individuals in the population, establishing the number and type of operators along with the OSM to be used. Then, we can design different RCGA families.

2.1. Genes combination functions

Let us consider c_i^1, $c_i^2 \in [a_i, b_i]$ two genes to be combined and $\alpha_i = \min\{c_i^1, c_i^2\}$ and $\beta_i = \max\{c_i^1, c_i^2\}$. The action interval $[a_i, b_i]$ of these genes can be divided into three intervals $[a_i, \alpha_i]$, $[\alpha_i, \beta_i]$ and $[\beta_i, b_i]$. These intervals bound three regions to which the resultant genes of some combination of the former may belong. Moreover, considering a region $[\alpha_i', \beta_i']$ with $\alpha_i' \leqslant \alpha_i$ e $\beta_i' \geqslant \beta_i$ would seem reasonable. Graphically it is demonstrated in Fig 1.

The intervals described above could be classified as exploration or exploitation zones. The interval with both genes being the extremes is an exploitation zone, the two intervals that remain on both sides are exploration zones and the region with extremes α_i' and β_i' could be considered as a relaxed exploitation zone. Table 1 shows these considerations which are related to Fig. 1.

Table 4
Exploration and relaxed exploitation levels

Operators		Exploration level	Operators	Relaxed exploitation level
F_1	S_1	$\alpha_i' = \alpha_i \quad \beta_i' = \beta_i$	L_1	$\alpha_i' = \alpha_i \quad \beta_i' = \beta_i$
F_2	S_2	$-$	L_2	$-$
F_3	S_3		L_3	
F_4	S_4	$+$	L_4	$+$
		$\alpha_i' = a_i \quad \beta_i' = b_i$		$\alpha_i' = a_i \quad \beta_i' = b_i$

The two most promising offspring of the four substitute their parents in the population.

ES4: For each pair of chromosomes from a total of $\frac{1}{2} \cdot p_c \cdot N$, four offspring are generated, the result of applying an F-crossover, an S-crossover, an M-crossover, and an L-crossover to them.

All four offspring will form part of the population, in such a way that two of them substitute their parents and the other two substitute two chromosomes belonging to the remaining $\frac{1}{2}$ of the population that should undergo crossover.

These four strategies represent four ways to introduce diversity as we describe below.

- ES1 introduces two offspring which come from crossover operators with exploration properties and another one generated by crossover operators with exploitation property.
- With regard to ES2, the properties or the form of the search space will have an influence on the selection. With this strategy, the exploitation level of the best regions is very high since it chooses the best individuals from a set of four: two with exploitation properties and two with exploration properties.
- ES3, the same as ES2, introduces a high exploitation level, but with more underlying exploration, caused by the use of an L-crossover instead of an M-crossover.
- With ES4, four offspring are produced, each one having been generated by crossover operators with different exploitation and exploration properties.

Table 5 shows all these observations. With respect to the aforementioned features, we considered three types of diversity levels: strong, high and weak.

During the execution of a GA, the diversity levels of the population can be observed by means of the comparison of the Online measure [5], which is

Table 5
Diversity levels associated with the presented OSM

Strategy	Diversity
ES1	Strong
ES2	Weak
ES3	Weak
ES4	High

defined as the average of all adaptation values up to a determined trial. The further it is from the optimum the more diversity would exist. It should be pointed out that the features of the diversity levels will depend on the crossover operators used.

3. Algorithms

Four families of algorithms using fuzzy connectives based crossovers were built: RCGA-ES1, RCGA-ES2, RCGA-ES3 and RCGA-ES4. All these families are made up by four algorithms, each one of them uses an FCB-crossover class from Table 3 (Logical, Hamacher, Algebraic and Einstein). For an RCGA-ESi family they will be denoted as RCGA-ESi-Logical, RCGA-ESi-Hamacher, RCGA-ESi-Algebraic and RCGA-ESi-Einstein respectively, every algorithm uses its respective OSM STi.

The value of the λ parameter, used by the fuzzy connective $P_j(\cdot, \cdot)$ associated with each M_j-crossover, is 0.35 for the RCGA-ES2 family and 0.5 for the remaining families. We propose the use of non-uniform mutation [14] which is considered the most suitable mutation operator for RCGA [14, 10]. Its definition is the following:

Table 2
Families of fuzzy connectives

Family	t-norm	t-conorm	Averaging fun. $(0 \leqslant \lambda \leqslant 1)$	Gen. comp. op.
Logical	$T_1(x, y) = \min(x, y)$	$G_1(x, y) = \max(x, y)$	$P_1(x, y) = (1 - \lambda)x + \lambda y$	$\hat{C}_1 = T_1^{1-\lambda} \cdot G_1^{\lambda}$
Hamacher	$T_2(x, y) = \dfrac{xy}{x + y - xy}$	$G_2(x, y) = \dfrac{x + y - 2xy}{1 - xy}$	$P_2(x, y) = \dfrac{1}{\frac{y - y\lambda - xy + x\lambda}{xy} + 1}$	$\hat{C}_2 = P_2(T_2, G_2)$
Algebraic	$T_3(x, y) = xy$	$G_3(x, y) = x + y - xy$	$P_3(x, y) = x^{1-\lambda} y^{\lambda}$	$\hat{C}_3 = P_3(T_3, G_3)$
Einstein	$T_4(x, y) = \dfrac{xy}{1 + (1 - x)(1 - y)}$	$G_4(x, y) = \dfrac{x + y}{1 + xy}$	$P_4(x, y) = \dfrac{2}{1 + \left(\frac{2-x}{x}\right)^{1-\lambda}\left(\frac{2-y}{y}\right)^{\lambda}}$	$\hat{C}_4 = P_4(T_4, G_4)$

Table 3
Set of crossover operators

Gene combination functions	Crossover operators
F_1, S_1, M_1, L_1	Logical
F_2, S_2, M_2, L_2	Hamacher
F_3, S_3, M_3, L_3	Algebraic
F_4, S_4, M_4, L_4	Einstein

when the F, S, M and L functions are applied respectively. It should be emphasized how these crossover operators have different properties: the F-crossover and S-crossover operators show exploration, the M-crossover operators show exploitation and the L-crossover operator shows relaxed exploitation.

Using the families of fuzzy connectives in Table 2 we can build four families of crossover operators. Each one of them shall be called the same as the related fuzzy connective family. Table 3 shows these.

According to the property (P6) of the families of fuzzy connectives in Table 2, we can see that the degree, in which each crossover operator shows its related property, will depend on the fuzzy connective on which it is based. Thus, we dispose of F_j-crossover and S_j-crossover operators with different exploration levels; the F_4-crossover and S_4-crossover show the maximum exploration, whereas the F_1-crossover and the S_1-crossover represent the minimum exploration. These operators give results between the extremes of the exploration domain. With respect to the L_j-crossover, the level of relaxed exploitation directly depends on the T_j, G_j and P_j used for its definition. Table 4 shows the behaviour of the interval definition for the crossover operators.

2.3. Application of the FCB-crossovers

We considered four strategies for applying the presented crossover operators to the population, these shall be called ES1–ES4. These strategies establish an exploration/exploitation relationship by selecting the most suitable offspring; therefore they induce required diversity levels in the population and the premature convergence can be eradicated.

ES1: For each pair of chromosomes from a total of $\frac{2}{3} \cdot p_c \cdot N$ (obtained from a sampling scheme), where p_c is the crossover probability, N is the population size and $p_c \cdot N$ is the number of chromosomes which should undergo crossover, three offspring are generated, the result of applying an F-crossover, an S-crossover and an M-crossover to them.

The OSM introduce the three offspring into the population in such a way that two of them substitute their parents and the remaining one substitutes a chromosome belonging to the remaining $\frac{1}{3}$ of the population that should undergo crossover.

ES2: For each pair of chromosomes from the total population that undergoes crossover, four offspring are generated, the result of applying an F-crossover, an S-crossover, and two M-crossovers to them. Both M-crossovers are based on the same averaging operator; however the parameter λ of one of them is $1 - \lambda$ of the other.

The OSM choose the two most promising offspring of the four to substitute their parents in the population.

ES3: For each pair of chromosomes from the total population that undergoes crossover, four offspring are generated, the result of applying an F-crossover, an S-crossover, an M-crossover and an L-crossover to them.

chromosomes are built: $H_1 = (c_1^1, c_2^1, \ldots, c_i^1, c_{i+1}^2, \ldots, c_n^2)$ and $H_2 = (c_1^2, c_2^2, \ldots, c_i^2, c_{i+1}^1, \ldots, c_n^1)$.

Discrete crossover [17]: The chromosome $H_1 = (h_1, \ldots, h_i, \ldots, h_n)$ is generated, where h_i is a randomly (uniformly) chosen value of the set $\{c_i^1, c_i^2\}$.

BLX-α *crossover* [6]: The chromosome $H_1 = (h_1, \ldots, h_i, \ldots, h_n)$ is generated, where h_i is a randomly (uniformly) chosen number from the interval $[c_i^{min} - I \cdot \alpha, c_i^{max} + I \cdot \alpha]$, $c_i^{max} = \max(c_i^1, c_i^2)$, $c_i^{min} = \min(c_i^1, c_i^2)$, and $I = c_i^{max} - c_i^{min}$.

In [10], experiments with several values of α are tried. The value which allowed the best results to be obtained was $\alpha = 0.3$.

Linear crossover [21]: Treating the parents as two points p_1 and p_2, three offspring are generated: $0.5p_1 + 0.5p_2$, $1.5p_1 - 0.5p_2$ and $-0.5p_1 + 1.5p_2$. The two most promising points are selected to substitute the parents in the population.

The algorithms that use each one of these crossovers shall be called RCGA-Simple, RCGA-Discrete, RCGA-BLX and RCGA-Linear respectively. The mutation operator used in RCGA-Discrete is Mühlenbein's mutation [17]. The non-uniform mutation is used for the remaining algorithms.

Also, a binary-coded GA (BCGA) has been included, which is based on a two-point crossover. For this purpose, we used the GENESIS program [8]. The number of binary genes assigned to each variable is 26 for f_4 and 11 for the remaining functions. Therefore, the precision is approximately 10^{-6}.

We carried out our experiments using the following parameters: the population size is 61 individuals, the crossover probability $p_c = 0.6$, the probability of chromosome update $p_u = p_m \cdot 25 = 0.6$, the parameter b used by the non-uniform mutation is 5, the selection procedure was linear ranking [3] and elitist selection [5] and the sampling model used was stochastic universal sampling [4]. We executed all the algorithms 5 times, each one with 5000 generations. For f_2, we repeated the experiments with 10 000 generations, since it is highly complex.

4.1. Results

Table 6 shows the average values of the results obtained. For each function the best in 5000 (for f_2

10 000 as well) generations is shown and the final On-line measure.

For the RCGA-ES2 and RCGA-ES3 families, Table 7 shows the percentages in which each crossover operator generated a chromosome that was chosen by the respective OSM to form part of the population. With these values, we shall discover the levels in which each operator produced promising offspring.

4.2. Results analysis

In general, the best algorithms belong to the RCGA-ES2 and RCGA-ES3 families. These families use an OSM that provides a *weak* diversity in the population; we can see that the values of the Online measure returned by these families are similar to the ones returned by the RCGA with simple, discrete, BLX-α and linear crossovers. The results of RCGA-ES3 in f_1, f_3 and f_4 are slightly lower than the ones in RCGA-ES2, since the use of the relaxed exploitation zone produces greater diversity levels (observe the Online measure). In the difficult f_2, this allowed the best solutions to be found.

We can point out the suitability of the OSM that chooses two individuals from a set of four: two with exploration properties and two with exploitation properties. However, this mechanism is very expensive since it needs too many evaluations.

In most functions, the RCGA-ES1 and RCGA-ES4 families returned low results; the replacement in the population of two parents by a set of three or four offspring produces too much diversity and thus slow convergence. To sum up, the exploration property related to these offspring selection mechanisms leads to a high diversity level during the GA execution; the Online measure in these families is greater than in the remaining executed algorithms. However, in f_2 this property was useful (RCGA-ES1-Hamacher has returned the second best result for this function).

Among the FBC-crossovers the Logical crossovers show the best behaviours. If we study Table 7 we can observe that for the Logical families the use of the F-crossover and S-crossover was considerable. This has allowed a suitable exploitation/exploration balance to be reached.

Non-uniform mutation: If $C = (c_1, \ldots, c_i, \ldots, c_n)$ is a chromosome and $c_i \in [a_i, b_i]$ is the element to be mutated, the resulting vector will be $C' = (c_1, \ldots, c_i', \ldots, c_n)$, where c_i' is determined by

$$c_i' = \begin{cases} c_i + \Delta(t, b_i - c_i) & \text{if } \tau = 0, \\ c_i - \Delta(t, c_i - a_i) & \text{if } \tau = 1, \end{cases}$$

τ being a random number which can have a value of zero or one, and

$$\Delta(t, y) = y(1 - r^{(1 - t/T)^b}),$$

where r is a random number from the interval $[0, 1]$, T is the maximum number of generations and b is a parameter chosen by the user, which determines the degree of dependency on the number of iterations. This function gives a value in the range $[0, y]$ such that the probability of returning a number close to zero increases as the algorithm advances. This property causes this operator to make a uniform search in the initial space when t is small, and very locally at later stages.

4. Experiments

Minimization experiments on the following four functions were carried out in order to study the efficiency of the algorithms proposed.

Sphere model [5]

$$f_1(x) = \sum_{i=1}^{n} x_i^2$$

$$-5.12 \leqslant x_i \leqslant 5.12$$

$$f_1^* = f_1(0, \ldots, 0) = 0$$

Generalized Rosenbrock's function [5]

$$f_2(x) = \sum_{i=1}^{n-1} (100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$$

$$-5.12 \leqslant x_i \leqslant 5.12$$

$$f_2^* = f_2(1, \ldots, 1) = 0$$

Generalized Rastringin's function [20]

$$f_3(x) = An + \sum_{i=1}^{n} x_i^2 - A\cos(\omega x_i)$$

$$A = 10, \omega = 2\pi$$

$$-5.12 \leqslant x_i \leqslant 5.12$$

$$f_3^* = f_3(0, \ldots, 0) = 0$$

Ackley's function [1]

$$f_4(x) = -a \cdot \exp\left(-b\sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} x_i^2}\right)$$

$$\qquad - \exp\left(\frac{1}{n} \cdot \sum_{i=1}^{n} \cos(c \cdot x_i)\right) + a + e$$

$$a = 20, b = 0.2, c = 2\pi$$

$$-32.768 \leqslant x_i \leqslant 32.768$$

$$f_4^* = f_4(0, \ldots, 0) = 0$$

where $n = 25$.

f_1 is a continuous, strictly convex and unimodal function. f_2 is a continuous and unimodal function, with the optimum located in a steep parabolic valley with a flat bottom. This feature will probably cause slow progress in many algorithms since they should permanently change their search direction to reach the optimum. f_3 is a scalable, continuous and multimodal function which is made from f_1 by modulating it with $A\cos(\omega x_i)$. Finally, f_4 is a continuous and multimodal function. It is formed by a platform with a lot of local optimums and a large hole where the global optimum is located. This hole has some local optimums placed on small plains. This may cause a problem when the search direction goes down towards the global optimum.

A GA does not need too much diversity to reach the global optimum of f_1 since there is only one optimum which could be easily accessed. On the other side, for the multimodal functions, the diversity is fundamental for finding a way to lead towards the global optimum. Also, in the case of f_2, diversity can help to find solutions close to the parabolic valley, and so avoid slow progress.

We executed all the algorithms presented along with three RCGA based on other crossover operators proposed in the literature. These operators were widely studied in [10] and they seem very adequate.

Simple crossover [21, 14]: A position $i \in \{1, 2, \ldots, n - 1\}$ is randomly chosen and the two new

Table 7
Utilization percentage of the crossover operators

Functions	Algorithms	F-crossover	S-crossover	M-crossover	M- or L-crossover
f_1	RCGA-ES2-Logical	18.836889	18.778556	31.171111	31.213444
	RCGA-ES2-Hamacher	0.006889	0.315333	49.841222	49.836556
	RCGA-ES2-Algebraic	0.000000	0.004444	49.997778	49.997778
	RCGA-ES2-Einstein	0.000000	0.000111	49.999889	50.000000
	RCGA-ES3-Logical	13.728889	14.455000	36.168000	35.648111
	RCGA-ES3-Hamacher	0.386556	12.493889	49.613444	37.506111
	RCGA-ES3-Algebraic	0.237778	8.879556	49.939444	40.943222
	RCGA-ES3-Einstein	5.585111	9.787111	48.796889	35.830889
f_2	RCGA-ES2-Logical	19.552667	19.788889	30.355778	30.302667
	RCGA-ES2-Hamacher	0.073778	1.005556	49.461556	49.459111
	RCGA-ES2-Algebraic	0.000222	0.103667	49.948444	49.947667
	RCGA-ES2-Einstein	0.000000	0.011222	49.994444	49.994333
	RCGA-ES3-Logical	15.924333	15.966333	33.717889	34.391444
	RCGA-ES3-Hamacher	0.554556	14.404556	49.444667	35.596222
	RCGA-ES3-Algebraic	0.245556	8.466444	49.943778	41.344222
	RCGA-ES3-Einstein	4.264444	7.874111	49.270444	38.591000
f_3	RCGA-ES2-Logical	27.052111	27.001111	22.949444	22.997333
	RCGA-ES2-Hamacher	13.189111	15.415444	35.724000	35.671444
	RCGA-ES2-Algebraic	0.025111	0.241889	49.868222	49.864778
	RCGA-ES2-Einstein	0.000667	0.035000	49.979667	49.984667
	RCGA-ES3-Logical	25.378111	25.324333	24.014667	25.282889
	RCGA-ES3-Hamacher	15.125444	26.221667	29.544889	29.108000
	RCGA-ES3-Algebraic	17.545111	22.616556	32.736444	27.101889
	RCGA-ES3-Einstein	19.394444	22.209778	32.836222	25.559556
f_4	RCGA-ES2-Logical	19.621000	19.655889	30.360111	30.363000
	RCGA-ES2-Hamacher	0.008000	0.326111	49.834667	49.831222
	RCGA-ES2-Algebraic	0.000111	0.005556	49.996889	49.997444
	RCGA-ES2-Einstein	0.000111	0.000778	49.999444	49.999667
	RCGA-ES3-Logical	14.542000	15.307556	35.153889	34.996556
	RCGA-ES3-Hamacher	0.393667	12.557667	49.598222	37.450444
	RCGA-ES3-Algebraic	14.437667	20.777222	36.198222	28.586889
	RCGA-ES3-Einstein	20.576444	24.509000	29.183111	25.731444

5. Conclusions

In this paper, we presented tools to model the population diversity of the RCGA. These tools are made up of two components
- Crossover operators families that introduce different exploitation and exploration degrees.
- OSM that allow us to induce different diversity levels once the operator family is fixed.

The results of the experiments show the suitability of using the Logical FCB-crossovers and the OSM that choose the two best elements from a set of four where the exploitation and exploration properties are equitably assigned. However, other types of combinations stood out for some functions (for example RCGA-ES1 for f_2). Then, for a particular problem, it should be the user who can select the more suitable family of operator and OSM to induce the best diversity levels, and so avoid the premature convergence problem. Tables 4 and 5 can help him to do so.

Finally, we consider the following question about the FCB-crossovers: *What kind of operators are they really?*. For Eshelman [6] a crossover operator should take into account the iterations between the genes of

Table 6
Results

Algorithms	f_1		f_2			f_3		f_4	
	5000	Online	5000	10 000	Online	5000	Online	5000	Online
BCGA	6.0E−10	3.6E+00	3.1E+01	1.6E+01	2.9E+03	1.7E+01	3.2E+01	6.9E−05	2.9E+00
RCGA-Simple	1.2E−12	9.7E+00	1.3E+01	3.0E+01	8.6E+03	5.3E+00	5.6E+01	4.6E−06	6.0E+00
RCGA-Discrete	8.5E−07	1.5E+00	4.9E+01	2.3E+01	7.8E+02	3.1E−04	1.4E+01	4.8E−03	2.7E+00
RCGA-BLX	7.5E−14	5.3E+00	2.1E+01	2.0E+01	3.6E+03	1.3E+01	1.0E+02	1.3E−06	5.5E+00
RCGA-Linear	1.7E−14	8.0E+00	2.1E+01	2.0E+01	6.2E+03	1.5E+01	1.7E+02	6.9E−07	6.0E+00
RCGA-ES1-Logical	4.5E−11	3.7E+00	2.2E+01	2.1E+01	2.7E+03	1.0E+01	7.3E+01	3.7E−05	4.6E+00
RCGA-ES1-Hamacher	1.1E−03	1.0E+02	1.3E+01	1.2E+01	8.2E+04	2.0E+01	3.6E+02	8.7E−01	1.7E+01
RCGA-ES1-Algebraic	7.3E−03	3.5E+02	1.3E+02	6.6E+01	7.3E+05	2.7E+01	5.6E+02	1.3E+00	2.1E+01
RCGA-ES1-Einstein	4.7E−03	3.9E+02	1.6E+02	8.3E+01	8.3E+05	2.5E+01	5.7E+02	1.4E+00	2.1E+01
RCGA-ES2-Logical	2.2E−15	1.2E+00	2.1E+01	2.0E+01	8.6E−02	4.8E−13	2.2E+01	2.3E−07	2.9E+00
RCGA-ES2-Hamacher	1.6E−12	4.6E+00	2.2E+01	2.1E+01	1.8E+03	1.6E+01	1.9E+02	5.7E−06	4.8E+00
RCGA-ES2-Algebraic	1.3E−12	3.3E+00	2.2E+01	2.1E+01	1.5E+03	1.5E+01	8.9E+01	6.3E−06	4.4E+00
RCGA-ES2-Einstein	1.6E−12	2.9E+00	2.2E+01	2.1E+01	1.6E+03	1.9E+01	9.1E+01	6.6E−06	4.2E+00
RCGA-ES3-Logical	1.0E−14	1.4E+00	2.1E+01	2.0E+01	8.9E+02	1.5E−12	2.4E+01	4.6E−07	3.2E+00
RCGA-ES3-Hamacher	4.8E−04	1.5E+01	9.3E+00	9.3E+00	2.5E+03	1.9E+01	2.2E+02	8.1E−01	1.2E+01
RCGA-ES3-Algebraic	1.0E−02	1.7E+02	7.9E+01	5.9E+01	2.5E+05	2.3E+01	4.7E+02	1.1E+00	2.0E+01
RCGA-ES3-Einstein	2.0E−03	2.6E+02	9.5E+01	3.4E+01	5.9E+05	1.7E+01	4.8E+02	1.5E+00	2.1E+01
RCGA-ES4-Logical	6.7E−11	3.4E+00	2.2E+01	2.1E+01	2.3E+03	1.3E+01	8.7E+01	4.3E−05	4.6E+00
RCGA-ES4-Hamacher	3.9E−04	9.3E+01	2.1E+01	1.3E+01	8.0E+04	2.2E+01	3.7E+02	7.2E−01	1.7E+01
RCGA-ES4-Algebraic	2.6E−03	4.8E+02	8.4E+01	7.1E+01	1.3E+06	2.3E+01	6.4E+02	8.9E−03	2.0E+01
RCGA-ES4-Einstein	2.6E−03	5.2E+02	8.3E+01	1.0E+02	1.5E+06	2.1E+01	6.5E+02	1.3E−01	2.0E+01

The decrease in the participation percentage of the F-crossovers and S-crossovers from the Logical families to the Einstein ones, in which it is approximately zero, and the increase in the Online measure in RCGA-ES1 and RCGA-ES4 from the Logical algorithms to the Einstein ones suggests that diversity is greater using F-crossover and S-crossover operators distant from the Logical (which agrees with the properties shown in Table 4). The reason is that these operators generate individual ones which do not generally preserve the properties of their parents, and so they introduce greater diversity. Under these circumstances, in the RCGA-ES2 family the OSM only considers the offspring generated by the averaging operators producing the premature convergence. In fact, under these OSM, the best results were found using the Logical family. The effect is similar in RCGA-ES3 for f_1 and

f_2. For the multimodal functions (f_3 and f_4) the relaxed diversity of the L-crossover has competed with the diversity of the F-crossover and the S-crossover to achieve the best solutions. Since in these functions the diversity allows good individuals to be found, approximately all these operators generated the same proportions of best individuals (see the percentages of RCGA-ES3 for f_3 and f_4 in Table 7).

If we consider the RCGA-ES2 and RCGA-ES3 families, from the results in Table 7 we could deduce that the maximum diversity level should be reached with the use of the Logical crossovers since the percentage of generated offspring belonging to the exploration zones is approximately 40%. However, the join effect of the use of operators far from the Logical together with the OSM of ES2 and ES3 produces over-exploitation, and so high Online measures.