Chromosome Differentiation for the Application of Parent-Centric Real-Parameter Crossover Operators

C. García-Martínez, M. Lozano, D. Molina, and A.M. Sánchez

Abstract—**Parent-centric real-parameter crossover operators create the offspring in the neighborhood of one of the parents, the female parent, using a probability distribution. The other parent, the male parent, is considered to define the range of the probability distribution. In this paper, we present a process that determines the individuals in the population that may become female or/and male parents. An important advantage of this proposal is that it makes possible the design of two kind of real-coded genetic algorithms: ones specialized for global search and ones that are effective local searchers. We introduce an hybrid technique that combines these real-coded genetic algorithms with the aim of achieving robust performance on a wide range of problems. The experiments show that this technique may reach high robustness levels, as compared with other real-coded genetic algorithms and hybrid real-coded genetic algorithms proposed in the literature.**

Index Terms—**real-coded genetic algorithms, parent-centric crossover operators, chromosome differentiation, hybrid real-coded genetic algorithms**

I. Introduction

In the initial formulation of *genetic algorithms* (GAs), the search space solutions are coded using the binary alphabet ([Gol89]); however, other coding types, such as real coding, have also been taken into account to deal with the representation of the problem. The real coding approach seems adequate when tackling optimisation problems of parameters with variables in continuous domains. The chromosome is a vector of floating point numbers, representing a solution of the problem. Obviously, both have the same length. GAs based on real-number representation are called *real-coded* GAs (RCGAs) ([Deb01a, Her98, Yan00]). Over the past few years, an increasing interest has arisen in solving real-world optimization problems using RCGAs. Some of these applications are shown in Table I.

TABLE I
APPLICATIONS OF REAL CODED GENETIC ALGORITHMS

Application	*Reference*
Neural Networks	[Bla01]
Aerospace Design	[Haj02]
Biotechnology	[Rou99]
Control	[Arf01]
Economic	[Duf01]
Signal Processing	[Har98]
Microware	[Cao00]
Industrial Electronics	[Shi02]
Industrial Engineering	[Aza02]
Tomography	[Tur02]
Water Resources Management	[Cha98]
Electric Power Systems	[Lin04]
Multispectral Image Classification	[Liu04]
Hydrology	[Jai04]
Electronic Commerce	[Kuo04]
Data Fitting	[Yos03]
Electrical Engineering	[Bas03]
Magnetic	[Lei04]
Power Systems	[Dam03]

The crossover operator has always been regarded as the main search operator in GAs ([DeJ92, Kit01]) 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 ([Deb01a, Her98, Her03]). *Parent-centric crossover operators* (PCCOs) are a family of real-parameter crossover operators that has currently received special attention. They include fuzzy recombination ([Voi95]), SBX ([Deb95]), PCX ([Deb02]), XLM ([Tak01]), vSBX ([Bal03]), PNX ([Bal04]), and PBX ([Loz04a]). In general, these operators use a probability distribution for creating offspring in a restricted search space around the region marked by one of the parent, the *female parent*. The range of this probability distribution depends on the distance among the female parent and the other parent involved in the crossover, the *male parent*.

Experiments carried out in [Deb02] have shown that PCCOs arise as a meaningful and efficient way of solving real-parameter optimization problems. Thus, the study of these operators becomes a topic of major interest for RCGA

C. García-Martínez, M. Lozano, and D. Molina are with the Department of Computer Science and Artificial Intelligence, University of Granada, 18071, Granada, Spain (e-mail: {cgarcia,lozano,dmolina}@decsai.ugr.es).

A.M. Sánchez is with the Department of Software Engineering, University of Granada, 18071, Granada, Spain (e-mail: amlopez@uvigo.es).

research. Nowadays, researchers are focusing their investigations on the following factors that affect the performance of PCCOs:

- *Evolution model*. Most authors chose the steady-state GA model (SSGA) ([Sys89, Whi89]) as the most adequate evolution model to take out profit from PCCOs. This election was mainly motivated by the fact that SSGAs may supply high selection pressure, which becomes well-suited for the meaningful operation of these operators.
- *Selection of the female parent and choose of the male parent*. Deb et al. ([Deb02]) and Ballester et al. ([Bal04]) recommend the use of a random selection policy for these two tasks. However, Lozano et al. ([Loz04b]) proposed a method for the selection of the female parent, *the uniform fertility selection*, and a different technique for the selection of the male parent, the *negative assortative mating* ([Fer01, Mat99]). They were designed with the aim of promoting high diversity.

So far, PCCO practitioners have assumed that every chromosome in the population may become either a female parent or a male parent. However, it is very important to emphasize that female and male parents have two differentiated roles:

1) Female parents *point* at the search areas that will receive sampling points, whereas
2) Male parents are used to determine the *extent* of these areas.

At this point, it is reasonable to think that some chromosomes may be well-suited to act either as female parents or as male parents. Thus, a promising way to improve the behaviour of PCCOs involves the introduction of a chromosome differentiation in the population by considering two groups: the female group (G_F) and the male group (G_M). In this case, the mechanisms for the selection of the female parents and for the choice of the male parents would operate uniquely on G_F and G_M, respectively.

In this paper, we present a simple process for female and male differentiation that creates G_F with the best N_F individuals in the population and G_M with the best N_M individuals (N_F and N_M are tunable parameters). An important feature of this process is that two different types of specialized RCGAs may be obtained by adjusting the N_F and N_F parameters: *global* RCGAs, which offer *reliability*, and *local* RCGAs, which provide *accuracy*. In addition, with the aim of producing a robust operation, we propose an *hybrid* RCGA *method* that combines these algorithms. First, it applies a global RCGA, and then, a local RCGA. The best individuals in population of the former become the individuals in the initial population of the last.

We set up the paper as follows. In Section II, we introduce relevant issues related to PCCOs. In Section III, we present the process for female and male differentiation, and carry out experiments to analyze its behaviour when it is incorporated in two RCGA models based on PCCOs. In Section IV, we detail the hybrid RCGA model and compare its performance with a baseline RCGA based on PCCOs, the G3-PCX ([Deb02]), and other hybrid RCGAs proposed in the literature. Finally, we draw some conclusions in Section V. In Appendix A, we include the features of the test suite used for the experiments. This appendix explains, as well, the way the executed algorithms were started with a *skewed population* not bracketing the global optimum.

II. Parent-Centric Crossover Operators

PCCOs assign more probability for creating offspring near the female parent than anywhere in the search space. In particular, they determine the genes of the offspring extracting values from intervals defined in neighbourhoods associated with the genes of the female parent, throughout probability distributions. The ranges of these probability distributions depend on the distance among the genes of the female parent and the genes of the male parent. Examples are fuzzy recombination ([Voi95]), SBX ([Deb95]), XLM ([Tak01]), PCX ([Deb02]), PNX ([Bal04]), and PBX ([Loz04a]).

In this section, we deal with the main aspects of PCCOs. In Section II.A, we describe three recent PCCO instances. This is useful to understand the particular features of the crossover operators analyzed in this paper. In Section II.B, we discuss different advantages of PCCOs and explain why they are like self-adaptive real-parameter mutation operators. Finally, in Section II.C, D, and E, we sketch in the main steps of three RCGA models based on PCCOs.

A. Three PCCO Instances

In this paper, we use three recent PCCO instances, which are described below.

1) PBX ([Loz04a]). Let us assume that

$$X = (x_1 \cdots x_n)\ Y = (y_1 \cdots y_n),\ x_i, y_i \in [a_i, b_i] \subset \Re, i = 1 \ldots n$$

are two chromosomes that have been selected to apply the crossover operator to them. PBX generates the offspring

$$Z = (z_1 \cdots z_n),$$

where z_i is a randomly (uniformly) chosen number from the interval $[l_i, u_i]$, with

$$l_i = \max\{a_i, x_i - I \cdot \alpha\}, u_i = \min\{b_i, x_i + I \cdot \alpha\}, I = |x_i - y_i|,$$

and α is a parameter associated with this operator (we have used α=0.8 as suggested in [Loz04b]).

2) PNX ([Bal04]). z_i is calculated as follows:

$$z_i = N(x_i, \frac{|x_i - y_i|}{\eta}),$$

where $N(\mu, \sigma)$ is a random number drawn from a Gaussian distribution with mean μ and standard deviation σ, and η is a tuneable parameter (the authors recommend $\eta = 2$).

3) PCX ([Deb02]). It is a *multiparent* crossover operator because uses $\mu > 2$ chromosomes for generating the offspring. Initially, it computes the centroid G of the chosen μ parents. Then, one parent, X_F, is chosen with equal probability as *female parent*. The direction vector $D = X_F - G$ is calculated. Thereafter, from each of the other $(\mu - 1)$ parents, the *male ones*, perpendicular distances d_i to the line D are computed and their average $\overline{d}$ is found. Finally, the offspring is created as follows:

$$Z = X_F + w_\varsigma |D| + \sum_{\substack{i=1 \\ i \neq F}}^{\mu} w_\eta \overline{d} E_i$$

where E_i are the $(\mu - 1)$ orthonormal bases that span the subspace perpendicular to D. The parameters w_ς and w_η are zero-mean normally distributed variables with variance σ_ς^2 and σ_η^2, respectively.

B. Advantages of PCCOs

Experiments carried out in [Deb02] have shown that PCCOs arise as a meaningful and efficient way of solving real-parameter optimization problems. We think these results are due to PCCOs combine two advantageous features:

- *PCCOs behave like a mutation operator*. PCCOs generate solutions that are close to the female parent. In this way, they may be seen as a special type of mutation. In fact, it is interesting to highlight that most RCGA models based on PCCOs do not use additional mutation operators ([Bal04, Deb02, Loz04b]).

  This view is very important, as we will explain. We should highlight that two of the most important avenues of research in evolutionary algorithms that use real-coding pay attention on mutation as the primal operation for generating novel search points. They are the *evolution strategies* (ESs) ([Sch95]) and the *evolutionary programming* (EP) ([Fog95]). They simulate evolution as a phenotypic process, that is, a process emphasizing the behavioural link between parents and offspring, rather than their genetic link. In this way, the emphasis is placed on the use of mutation operators that generate a continuous range of behavioural diversity yet maintain a strong correlation between the behaviour of the parent and its offspring ([Fog94]).

  Deb adopts a similar idea to justify the work of PCCOs ([Deb05]): since each parent is carefully picked by the selection mechanism, for most real-parameter optimization problems it can be assumed that solutions *close* to these parents are also likely to be potential good candidates. From this claim, we may remark an additional outstanding comment: the operation of PCCOs may become particularly promising when they are applied to highly fit individuals. This explains that most RCGAs based on PCCOs appeared in the literature are steady-state GAs, because they may attain higher selection pressure levels than generational GAs ([DeJ93]).

- *PCCOs are self-adaptive crossover operators*. PCCOs define a probability distribution of offspring solutions based on some measure of distance among the parent solutions. If the parents are located closely to each other, the offspring generated by the crossover might distribute densely around the female parent. On the other hand, if the parents are located far away from each other, then the offspring will be sparsely distributed around it. Therefore, PCCOs may fit their action range depending on the diversity of the population using specific information held by the parents. In this way, depending on the current level of diversity in the population, they may favour the production of additional diversity (divergence) or the refinement of the solutions (convergence). This behaviour is achieved without requiring an external adaptive mechanism.

  In fact, in the recent past, RCGAs with crossover operators with this feature have been demonstrated to exhibit self-adaptive behaviour similar to that observed in ESs and EP approaches ([Deb01b, Kit01]). Moreover, Beyer et al. ([Bey01]) argue that a variation operator that harnesses the difference of the parents in the search space is essential for the resulting evolutionary algorithm to exhibit self-adaptive behaviour on the population level.

To sum up, we may conclude that PCCOs may be seen as *self-adaptive real-parameter mutation operators*. Several self-adaptive mutation techniques have been proposed for ESs and EP as well (see [Bäc96]). However, there exists a clear difference:

- ESs and EP make parameters of this operator, such as *standard deviations*, evolve simultaneously with the decision variables.
- PCCOs calculate implicitly the standard deviations using information about the distribution of the individuals in the population.

Finally, we should point out that since PCCOs work like self-adaptive mutation operators, they are amenable for the design of effective *local search procedures*. In fact, in [Loz04a], a *crossover hill-climbing* based on PBX is proposed as local searcher of a real-coded memetic algorithm.

C. The U&N-PBX Algorithm

This algorithm is a steady-state RCGA that combines two diversification selection schemes, *the uniform fertility selection* (UFS) and the *negative assortative mating* (NAM), with a replacement method that introduces high selection pressure, the *replace worst* strategy. Its basic algorithm step is the following ([Loz04b]):

1) Select a female parent using *UFS*.
2) Select a male parent using *NAM*.
3) Create an offspring applying *PBX* to the parents.

4) Evaluate the offspring with the fitness function.

5) Introduce the offspring in the population using the *replace worst* strategy.

The female selection determines the search zones that are sampled by PBX. UFS encourages an exhaustive coverage of the regions represented in the population. It records the number of times every chromosome in the population was selected as female parent during its lifetime, and selects as next female parent the current individual with the lowest number of times.

The male selection defines the extent of the sampling areas. NAM spreads the distribution used for generating the offspring, with the objective of creating chromosomes very dissimilar from their parents. From a set of n_{nass} candidate individuals (selected at random), this procedure chooses, as male parent, the chromosome with the highest Euclidean distance from the female parent (in our experiments n_{nass}=5).

UFS and NAM are designed to favour diversity only; they do not cause a search bias towards the fittest individuals. Thus, they should be combined with a replacement strategy yielding some degree of selection pressure. In particular, Lozano et al. [Loz04b] proposed to apply UFS and NAM along with the replace worst strategy, which replaces the worst individual in the population only if the offspring is better. This strategy introduces high selection pressure, because it maintains the best individuals appearing so far.

D. The SPC-PNX Algorithm

SPC-PNX ([Bal04]) is a steady-state RCGA that uses *uniform random selection*, without replacement, to select two parents from the current population (the roles of female and male parent are assigned randomly). Then, it applies the *PNX* operator to generate an offspring, C_{off}. As replacement policy, it considers the *scaled probabilistic crowding* (SPC) scheme:

1) SPC scans n_{REP} individuals of the population (authors considered n_{REP}=2) and picks the individuals that most closely resemble the offspring, C_{cst}.

2) C_{off} and C_{cst} enter a probabilistic tournament with culling likelihood (survival, if we were in a maximization problem) given by:

$$p(C_{off}) = \frac{f(C_{off}) - f_{best}}{f(C_{off}) + f(C_{cst}) - 2 f_{best}} \text{ and } p(C_{cst}) = 1 - p(C_{off}),$$

where $f(\cdot)$ is the fitness function and f_{best} is the fitness function value of the best individual in the offspring and selected group of n_{REP} individuals.

This replacement scheme has several beneficial features; first, it is a crowding method that may promote the creation of subpopulations to explore different regions of the search space, and second, it implements elitism in an implicit way: if the best individual in either offspring or current parent population enters this replacement competition will have probability zero of being culled.

E. The G3-PCX Algorithm

G3-PCX ([Deb02]) is a steady-state RCGA model in which the recombination and selection operators are intertwined in the following manner:

1) From the population select the best parent and μ-1 other parents randomly.

2) Generate λ offspring from the chosen parents using *PCX*.

3) Choose two individuals at random from the population.

4) Form a combined sub-population of chosen two individuals and offspring, choose the best two solutions and replace the chosen two elements with these solutions.

The performance of the G3-PCX was investigated on three commonly-used test problems and was compared with a number of evolutionary and classical optimization algorithms. It was found to be consistently and reliably performing better than all other methods. Thus, G3-PCX arises as reference point in the PCCO research. In this paper, it is considered as baseline algorithm.

III. A Process for Female and Male Differentiation

Any individual in a population processed by the algorithms described in Sections II-C, II-D, and II-E may be selected as female or male parent. However, female and male parents have two different missions: female parents are the centers of the sampling zones whereas male parents determine the extent of these zones. Thus, we think that the differentiation of the chromosomes into two groups (G_F and G_M) may have drastic effects on PCCO performance.

In this section, we present a *process for female and male differentiation* (FMD) that determines the individuals in the current population that may become female parents only, or male parents only, or, even, both female and male parents. This process should be carried out before the application of the female and male selection mechanisms (which will be applied on the corresponding groups). In this way, it may be considered as a *preselection mechanism*.

The FMD process proposed needs two parameters, N_F and N_M, with $N_F \leq N$ and $N_M \leq N$ (N is the population size) and obtains G_F and G_M as follows (see Figure 1):

- G_F consists of the N_F best individuals in the population, and
- G_M is made of the N_M best individuals in the population.

In addition, it should be ensured that either $N_F = N$ or $N_M = N$ is fulfilled. Next, we provide two remarks derived from this definition:

1) In the case of N_F=N_M, there is not female and male differentiation, reaching the standard way of applying PCCOs.

2) $G_F \cap G_M \neq \emptyset$, which means that some individuals may be both female and male parents (see Figure 1). In particular, the N_{min} best individuals in the population have this characteristic, where $N_{min} = min\{N_F, N_M\}$. We have assumed that these individuals may be well-suited to act as both female and male

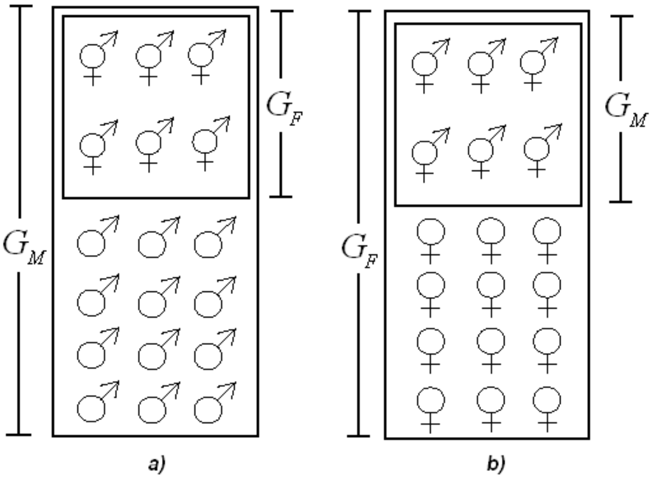

Fig. 1. Female and male differentiation imposed by the FMD process, considering: a) $N_F<N_M$ and b) $N_F>N_M$. (We assume that the two populations are ordered based on the fitness values of the chromosomes)

parents.

Another important feature of this FMD process is that it introduces *selective pressure* in the processes of selection of the female and male parents (which are applied later). In addition, we may point out that:

- The impact of the N_F parameter on this selective pressure is simple and predictable.
- The range of selective pressure that can be made by varying the N_F parameter is very large.

These are two desirable features for a selection process ([Bäc94]). On the one hand, when N_F is low, high selection pressure degrees are achieved, which forces the search process to be very focused in the best regions. On the other hand, if N_F is high, the selection pressure is softened, providing an extensive sampling on the search areas represented in the current population.

The inclusion of the FMD process in the algorithms explained in Sections II-C, II-D, and II-E is straightforward. For example, for the case of the most sophisticate algorithm, the U&N-PBX (Section II-C), the FMD process is performed before the application of UFS and NAM (Steps 1 and 2), and returns G_F and G_M. Then, UFS selects a female parent from G_F and NAM picks a male parent from G_M. The remaining steps are accomplished following the usual way.

Finally, we should recognize out that the idea of incorporating chromosome differentiation in GAs is not new:

- Bandyopadhyay et al. ([Ban98]) claim that nature generally differentiates the individuals of a species into more than one class. Sexual differentiation is a typical example, where the individuals of a species generally belong to either male or female class. The prevalence of this form of differentiation indicates an associated advantage which appears to be in terms of cooperation between two dissimilar individuals, "*who can at the same time specialize in their own fields*". These authors developed a GA with chromosome differentiation that distinguishes chromosomes into two categories (M and F), labeled by two class bits, and allows crossover only between different categories.
- Goh et al. ([Goh03]) propose a selection mechanism for generational GAs inspired by sexual selection principles through female choice selection. First, the mechanism uses random separation of males and females. Then, it forces all females get to reproduce regardless of their fitness level, and selects the males using tournament selection (which is fitness biased). The authors proposed this selection scheme with the intent of having a clear and balanced separation of functions between exploration and exploitation.

Next, in Sections III.A and B, we analyze empirically the effects of incorporating the FMD process in the U&N-PBX and SPC-PNX algorithms.

A. *Study of U&N-PBX with the FMD Process*

In this section, we investigate the way the application of the FMD process may affect the behaviour of the U&N-PBX algorithm (Section II-C). We have carried out minimization experiments on the test suite described in Appendix A considering different values for N_F and N_M (N_F = 1, 5, 25, 50, 100, 200, 300, and 400 individuals, and N_M=25, 50, 100, 200, 300, and 400 individuals). All the possible combinations of these values were investigated. The algorithms were executed 50 times, each one with a maximum of 100,000 evaluations. Table B.I (Appendix B) shows the results obtained. The performance measure used is the average of the best fitness function found at the end of each run.

Next, we examine the characteristics of the N_F and N_M combinations that reach the best solutions. Table II displays the fifteen combinations that returned the best results for each test problem in a sorted way. We have underlined, in boldface, the best combinations with N_F=N_M, which represent the standard way of implementing the U&N-PBX algorithm.

TABLE II
BEST N_F AND N_M VALUES USED BY U&N-PBX WITH FMD

f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}
5, 100	5, 100	25, 50	400, 100	400, 300	200, 400
5, 50	5, 200	**50, 50**	300, 100	200, 400	300, 400
1, 200	5, 300	5, 100	**200, 200**	**300, 300**	**400, 400**
25, 50	25 100	100, 25	400, 50	300, 400	50, 400
5, 200	25 200	25, 100	100, 400	400, 200	400, 300
1, 300	25 300	5, 200	200, 100	100, 300	300, 300
25, 100	5 400	200, 25	100, 300	300, 200	400, 200
5, 300	25 400	100, 50	100, 200	200, 300	100, 400
1, 400	25 50	5, 300	300, 200	100, 400	200, 300
50, 50	50 100	50, 100	200, 300	300, 100	300, 200
100, 25	50 200	300, 25	300, 50	50, 400	100, 300
5, 400	50 300	25, 200	50, 400	200, 200	50, 300
200, 25	**50, 50**	5, 400	200, 50	100, 200	25, 400
50, 25	50, 400	400, 25	400, 25	400, 400	200, 200
25, 200	100, 50	200, 50	50, 300	25, 400	100, 200

For every test functions, the combination that achieves the best results fulfils that $N_F \neq N_M$, that is, it implies female and male differentiation. We have applied a two-sided t-test (H_o: means of the two groups are equal, H_a: means of the two group are not equal) at 0.05 level of significance in order to ascertain if differences in the performance of the best combinations with $N_F \neq N_M$ are significant when compared against the one for the best combinations with $N_F = N_M$ (we have introduced a '*' sign when this occurs). Table III has the results. For most test functions, we may see that the female and male differentiation allows the performance of U&N-PBX to be improved. Only for f_{Gri}, a comparable behaviour is achieved.

TABLE III
BEST COMBINATIONS WITH $N_F \neq N_M$ VERSUS BEST COMBINATIONS WITH $N_F = N_M$ (U&N-PBX WITH FMD)

Test Problem	N_F, N_M	Average Best Fitness
f_{sph}	5, 100 *	9.98e-187
	50, 50	6.19e-095
f_{Ros}	5, 100 *	1.56e+000
	50, 50	1.46e+001
f_{Sch}	25, 50 *	1.01e-012
	50, 50	3.54e-010
f_{Ras}	400, 100 *	2.60e+000
	200, 200	3.58e+000
f_{Gri}	400, 300	3.48e-004
	300, 300	5.42e-004
P_{sle}	200, 400 *	5.45e+000
	400, 400	7.84e+000

Another important observation, from Table II, is that the best solution for each problem is reached by using different values for N_F and N_M. In particular, we may remark the big differences between the best N_F values for the multimodal and complex problems (f_{Ras}, f_{Gri}, and P_{sle}) and the ones for the unimodal problems (f_{sph}, f_{Ros}, and f_{Sch}):

- Low N_F values (e.g. 5 or 25 individuals) induce very high selection pressures, which may be suitable to obtain *accuracy* on unimodal problems. However, this is not the unique determinant aspect to achieve success for this type of problems (and in particular in our case, since we follow a skewed initialisation (Section A.3 in Appendix A)). The use of high N_M values (e.g. 50 or 100 individuals) enlarges the sampling zones. This circumstance has produced synergetic effects with the selection pressure, ensuring that the algorithm may progress towards better zones.

- The use of high N_F values induces a scattered search, because different female parents become the centre of attention of the PCCO. This high exploration of the search space is essential to provide *reliability* for multimodal and complex problems. In addition, the application of very high N_M values (e.g. 400 individuals) reinforces this exploration ability, inducing a promising performance on the most complex problems (e.g. P_{sle}).

TABLE IV
BEST N_F AND N_M VALUES USED BY SPC-PNX WITH FMD

f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}
5, 100	5, 200	5, 400	400, 100	200, 400	400, 300
1, 200	5, 300	5, 200	100, 400	400, 300	**300, 300**
1, 100	5, 100	5, 300	300, 100	400, 200	200, 200
1, 300	5, 400	25, 100	**200, 200**	300, 200	200, 400
5, 200	1, 300	100, 25	100, 300	**300, 300**	400, 400
1, 400	1, 400	25, 50	100, 200	300, 100	300, 200
5, 50	200, 25	50, 25	200, 100	400, 100	400, 200
5, 300	1, 200	200, 25	400, 50	300, 400	400, 100
25, 50	25, 100	25, 200	300, 200	100, 300	200, 300
50, 25	25, 50	**50, 50**	50, 400	100, 400	300, 400
25, 100	100, 25	100, 50	100, 100	200, 300	100, 400
100, 25	400, 25	50, 100	200, 300	200, 100	200, 100
5, 400	25, 400	300, 25	50, 300	400, 50	300, 100
25, 25	25, 300	25, 300	300, 50	100, 200	100, 300
50, 50	25, 200	400, 25	200, 400	200, 200	100, 200

TABLE V
BEST COMBINATIONS WITH $N_F \neq N_M$ VERSUS BEST COMBINATIONS WITH $N_F = N_M$ (SPC-PNX WITH FMD)

Test Problem	N_F, N_M	Average Best Fitness
f_{sph}	5, 100 *	4.07e-072
	50, 50	8.10e-035
f_{Ros}	5, 200 *	9.88e+000
	50, 50	1.94e+001
f_{Sch}	5, 400 *	6.10e-003
	50, 50	3.40e-001
f_{Ras}	400, 100 *	2.15e+001
	200, 200	2.45e+001
f_{Gri}	200, 400	3.15e-003
	300, 300	4.72e-003
P_{sle}	400, 300	4.15e+001
	300, 300	4.25e+001

B. Study of SPC-PNX with the FMD Process

In this section, we undertake the study of the application of the FMD process to the SPC-PNX algorithm (Section II.D). Our aim is to confirm that the use of this mechanism may improve as well the performance of this algorithm, and conclude, that it is a useful tool to empower RCGAs based on PCCOs.

Tables B.II (Appendix B), IV, and V show the same information as Tables B.I, II, and III, but for the case of SPC-PNX with the FMD process. Again, the best combinations adopt $N_F \neq N_M$, and in general, they are significantly better than the best combinations with $N_F = N_M$ (see Table V). In addition, we observe again the greater differences between the well-suited N_F values for the unimodal problems (e.g. 5

individuals) and the ones for the complex and multimodal problems (e.g. 200 and 400 individuals).

These results along with the ones in Section III.A suggest that the FMD process arises as a key mechanism for the design of effective RCGAs based on PCCOs.

C. U&N-PBX with FMD versus SPC-PNX with FMD

In this section, we compare the best results of the U&N-PBX algorithm with the FMD process with the ones of the SPC-PNX algorithm with the FMD process. Table VI has these results (for each problem, the first result is obtained from U&N-PBX). We indicate, with the '*' sign, the result that becomes significantly the best, according to a t-test.

TABLE VI
U&N-PBX WITH FMD VERSUS SPC-PNX WITH FMD

Test Problem	N_F ,N_M	Average Best Fitness
f_{sph}	5, 100 *	9.98e-187
	5, 100	4.07e-072
f_{Ros}	5, 100 *	1.56e+000
	5, 200	9.88e+000
f_{Sch}	25, 50 *	1.01e-012
	5, 400	6.10e-003
f_{Ras}	400, 100 *	2.60e+000
	400, 100	2.15e+001
f_{Gri}	400, 300	3.48e-004
	200, 400	3.15e-003
P_{sle}	200, 400 *	5.45e+000
	400, 300	4.15e+001

We may remark that U&N-PBX with FMD outperforms the other algorithm on all the problems. We think that the application of the FMD process allows the operation of UFS and NAN to be more effective, providing very good solutions for all the test problems.

IV. Global RCGAs and Local RCGAs

An important conclusion obtained from Sections III.A and III.B is that the FDM process allows the application of PCCOs to be more effective. This is possible due to:

- The large range of selective pressure obtained by varying N_F, and
- The possibility of supplying adequate exploration levels by controlling N_M.

These two characteristics allow us to design two different kind of specialized search algorithms:

- RCGAs that reach *accurate* solutions when they deal with unimodal problems. An example may be U&N-PBX with the FDM process using N_F =5 and N_M=100 (see Table II).
- RCGAs that offer *reliable* solutions when they attempt to solve multimodal and complex problems. An example may be U&N-PBX with the FDM process adopting N_F =200 and N_M=400 (see Table II).

In this paper, these algorithms will be termed *local RCGAs* and *global RCGAs*, respectively.

In order to achieve a *robust* operation for problems with different characteristics, local and global RCGAs should be hybridized in such a way that their beneficial advantages might be offered simultaneously, allowing the most promising search space regions to be reached and refined. Thus, the objective of this section is the design of an hybrid RCGA model that might be suited to most practical problems.

A. The Conflict between Accuracy and Reliability

There exists a fundamental conflict between *accuracy* and *reliability* when searching for the global optimum in most practical problems ([Ren96]). Traditionally, this conflict was tackled by means of advanced genetic operators (e.g. the non-uniform mutation operator proposed in [Mic92]), adaptation of GA control parameters ([Eib99]), heterogeneous distributed populations ([Her00]), etc. Nowadays, an alternative that receives special attention is the *hybridization* of GAs with other search techniques. Three important examples are:

- *Memetic algorithms* ([Mos99]). They are GAs that apply a separate local search process (which searches efficiently only for a local optimum) to members of the population after recombination and mutation. In this case, the local search procedure works within the GA.

- *Continuous hybrid algorithm* ([Che03]). It comprises two main stages. The first stage involves the run of an RCGA. The second stage involves the application of a local search process to the best individual found by the RCGA.

- *Hybridization of GAs with different purposes.* Kazarlis et al. ([Kaz01]) propose the use of a *microgenetic algorithm* (MGA) (GA with small population that evolves for a few generations) as a generalized hill-climbing operator. They combine a standard GA with the MGA to produce an hybrid genetic scheme. In contrast to conventional hill climbers that attempt independent steps along each axis, a MGA operator performs genetic local search. The authors claimed that the MGA operator is capable of evolving paths of arbitrary direction leading to better solutions and following potential ridges in the search space regardless of their direction, width, or even discontinuities.

  Lozano et al. ([Loz04a]) present a real-coded memetic algorithm that applies a *real-parameter crossover hill-climbing*. This hill-climbing maintains a pair of parents and performs repeatedly crossover (the PBX operator (Section II.A)) on this pair until some number of offspring is reached. Then, the best offspring is selected and replaces the worst parent only if it is better. The authors claimed that this process may be conceived as a *micro selecto-recombinative* RCGA.

B. Combining Global RCGAs and Local RCGAs

In this section, we propose an hybrid RCGA model that combines a global RCGA (with N_F^G and N_M^G as N_F and N_M values, respectively) and a local RCGA (with N_F^L and N_M^L as

TABLE VII
RESULTS FOR THE HYBRID RCGA MODEL

Algorithms	f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}
Local RCGA	9.98e-187	1.56e+000	1.74e-009	2.90e+002	1.27e-002	1.64e+002
GL-25	3.17e-147	7.61e-001	1.61e-007	1.33e+001	2.22e-017	4.69e+000
GL-50	1.29e-104	6.03e+000	1.22e-005	8.26e+000	1.33e-017	3.25e+000
GL-75	3.94e-061	1.22e+001	3.26e-002	3.74e+000	0.00e+000	2.72e+000
Global RCGA	2.95e-018	1.91e+001	3.12e+001	1.92e+001	4.93e-004	5.45e+000

N_F and N_M values, respectively). The hybridization is accomplished following the idea of the continuous hybrid algorithm ([Che03]) because is very simple: first, it runs the global RCGA during the P_G% of the available evaluations, and then, it performs the local RCGA. The initial population for the local RCGA consists of the N_{max} best individuals in the final population of the global RCGA, where $N_{max}=max\{N_F^L, N_M^L\}$.

This hybridization follows a classical heuristic: "*to protect the exploration in the initial stages and the exploitation later*". This heuristic has been considered for designing other search techniques, such as *simulated annealing*. With the initial exploration, the diversity became greater, increasing the probability of finding zones which are close to optimal solutions. Then, supposing that the population has information about these zones, the convergence towards the optimum is produced through exploitation.

We have implemented three instances of this hybrid model. The global RCGA is the U&N-PBX algorithm with FMD adopting N_F^G=200 and N_M^G=400. The local RCGA is the same algorithm but with N_F^L=5 and N_M^L=100. The instances are distinguished with regards to the values considered for P_G (P_G=25%, 50%, and 75%). They will be called GL-25, GL-50, and GL-75, respectively. Their results are outlined in Table VII. We have included the results for the local and global RCGAs when they are independently executed.

We should point out that, for four problems, there exist instances of our hybrid RCGA model that perform better than the sole usage of the corresponding global and local RCGAs: f_{Ros} (GL-25), f_{Ras} (GL-50 and GL-75), f_{Gri} (GL-25, GL-50, and GL-75) and P_{sle} (GL-25, GL-50, and GL-75). This means that the hybrid technique proposed is a suitable way to achieve *synergy* between global and local RCGAs.

Another important observation is that GL-25 and GL-75 have returned solutions for f_{Ros} and f_{Gri}, respectively, which are the best ones as compared with the solutions achieved by all the algorithms considered in this paper (See Table III and V).

C. Comparison with other Algorithms

The main aim of this section is to compare our hybrid RCGA method with the following algorithms:

- The G3-PCX algorithm (proposed in [Deb02] and described in Section II.E), which is considered in this paper as baseline RCGA based on PCCOs. We implement several instances that use different λ values (λ=2, 3, and 4), μ=3, and the population size is 150 individuals. The parameters for the PCX operator are: σ_ς^2=0.1 and σ_η^2=0.1. This setting was suggested by the authors. These algorithms will be referred as G3-PCX-λ.
- The real-coded memetic algorithm with crossover hill-climbing ([Loz04a]). Their authors claimed that this algorithm improves the performance of other real-coded memetic algorithms appeared in the literature. It will be called RCMA-XHC.
- An hybrid algorithm that combines the CHC algorithm ([Esh91]) with the Solis and Wets' algorithm ([Sol81]) following the idea of the continuous hybrid algorithm ([Che03]). First, the CHC is performed during the P_G% of the available evaluations, and then, the local searcher refines the best individual returned by CHC. We have considered P_G=25%, 50%, and 75%. These algorithms will be termed CHC-SW-P_G. CHC is very adequate for designing hybrid RCGAs, because it incorporates different techniques to promote high population diversity (incest prohibition and restart). Our CHC implementation applies PBX with α=1 ([Loz04b]).

We have chosen GL-25 (Table VII) for the comparison, because it shows an acceptable level of robustness. The results for all these algorithms are included in Table VIII. We have included the results for the CHC algorithm as well. A t-test was applied to detect significant differences between GL-25 and the other algorithms. We denote the direction of any significant differences as follows:

- A plus sign (+): the performance of GL-25 is better than the one of the corresponding algorithm.
- A minus sign (-): the algorithm improves the performance of GL-25.
- An approximate sign (~): non significant differences.

In order to facilitate the analysis of these results, we have included in Table VIII the last three columns. They have the number of improvements, reductions, and non-differences, according to the t-test.

We may remark that, in general, GL-25 outperforms all the other algorithms, because it achieves remarkable amount of improvements and few reductions. In addition, GL-25 provides the best results for four out six test problems: f_{Ros}, f_{Ras}, f_{Gri}, and P_{sle}. Therefore, we may summarize that our proposal of combining global and local RCGAs is very competitive with state-of-the-art RCGAs.

TABLE VIII
COMPARISON OF GL-25 WITH OTHER ALGORITHMS

Algorithms	f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}	+	~	-
G3-PCX-2	6.77e-176 -	1.13e+001 +	5.62e-022 -	4.90e+002 +	1.56e-001 +	4.36e+002 +	4	0	2
G3-PCX-3	4.70e-166 -	2.10e+000 +	1.67e-028 -	4.82e+002 +	9.47e-002 +	2.32e+002 +	4	0	2
G3-PCX-4	8.56e-146 +	1.45e+000 +	4.27e-028 -	4.88e+002 +	4.45e-002 +	1.52e+002 +	5	0	1
RCMA-XHC	9.47e-100 +	2.85e+000 +	9.48e-007 +	1.24e+001 ~	4.71e-002 +	1.87e+002 +	5	1	0
CHC	6.02e-031 +	1.89e+001 +	3.57e-002 +	1.93e+001 +	5.72e-003 +	6.48e+001 +	6	0	0
CHC-SW-25	3.41e-322 -	5.59e+000 +	1.91e-031 -	2.61e+001 +	5.27e-003 +	8.13e+001 +	4	0	2
CHC-SW-50	3.80e-322 -	8.99e+000 +	8.51e-022 -	2.72e+001 +	4.93e-003 +	6.98e+001 +	4	0	2
CHC-SW-75	5.30e-205 -	1.31e+001 +	3.97e-012 -	2.15e+001 +	5.91e-003 +	8.60e+001 +	4	0	2
GL-25	3.17e-147	7.61e-001	1.61e-007	1.33e+001	2.22e-017	4.69e+000			

The CHC-SW and G3-PCX instances significantly improve the results of GL-25 on two unimodal problems: f_{sph}, and f_{Sch}. On the one hand, the Solis and Wets' algorithm is a local searcher that becomes effective for this type of problems. On the other, G3-PCX always considers the best individual in the population as parent. This favors an exhaustive exploitation of the zones located around this individual. For some unimodal functions, this may cause a successful operation.

V. CONCLUSIONS

This paper presented an FMD process that assigns to the chromosomes in the population the role of female or/and male parent, before the application of a particular PCCO instance.

An experimental study has shown that this process may empower the work of PCCOs when its associated parameters, N_F and N_M, are adequately adjusted. We have made use of this fact to design reliable global RCGAs and accurate local RCGAs.

Finally, with the aim of achieving a robust operation, we have followed a simple hybridization technique to put together these specialized search algorithms. We have confirmed empirically that this technique allows synergy to occur between global and local RCGAs, that is, their combination performs better than the sole usage of any of them. In addition, in general, it improves the performance of the G3-PCX algorithm (considered here as baseline RCGA based on PCCOs) and other recent hybrid RCGA instances proposed in the literature.

In essence, the research line initiated with the present work is indeed worth of further studies. We are currently extending our investigation to different test-suites and real-world problems. Also we intend to: 1) design *adaptive* FMD processes that select female and male parents depending on the current state of the search, 2) study the effectiveness of the FMD process when multiparent crossover operators are applied (e.g. PCX (Section II.A)), and 3) study adaptive mechanisms to adjust the P_G parameter required by the hybrid RCGA model proposed.

APPENDIX A. TEST SUITE

The test suite that we have used for the experiments consists of five test functions and one real-world problem. They are described in Subsections A.1 and A.2, respectively.

A.1. TEST FUNCTIONS

We have considered five frequently used test functions, which are described below. The dimension of the search space is 15 for f_{Gri} and 25 for the remaining test functions.

Sphere model ([DeJ75]).

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

$$-5.12 \le x_i \le 5.12,\ n = 25,\ f_{Sph}(x^*) = 0.$$

Generalised Rosenbrock's function ([DeJ75]).

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

$$-5.12 \le x_i \le 5.12,\ n = 25,\ f_{Ros}(x^*) = 0.$$

Schwefel's function 1.2 ([Sch81]).

$$f_{Sch}(x) = \sum_{i=1}^{n} \sum_{j=1}^{i} x_j^2$$

$$-65.536 \le x_i \le 65.536,\ n = 25,\ f_{Sch}(x^*) = 0.$$

Generalised Rastrigin's function ([Tör89]).

$$f_{Ras}(x) = a \cdot n + \sum_{i=1}^{n} x_i^2 - a \cdot \cos(\omega \cdot x_i)$$

$$a = 10,\ \omega = 2 \cdot \pi,\ -5.12 \le x_i \le 5.12,\ n = 25,\ f_{Ras}(x^*) = 0.$$

Griewangk's function ([Gri81]).

$$f_{Gri}(x) = \frac{1}{d} \sum_{i=1}^{n} x_i^2 - \Pi_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$d = 4000,\ -600 \le x_i \le 600,\ n = 25,\ f_{Gri}(x^*) = 0.$$

The main features of these functions are:

- f_{sph} is a continuous, strictly convex, and unimodal function.
- f_{Ros} 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 must continually change their search direction to reach the optimum. This function has been considered by some authors to be a real challenge for any continuous function optimisation program ([Sch94]). A great part of its difficulty lies in the fact that there are nonlinear interactions between the variables, i.e., it is *nonseparable* ([Whi96]).
- f_{Sch} is a continuous and unimodal function. Its difficulty concerns the fact that searching along the coordinate axes only gives a poor rate of convergence, since the gradient of f_{Sch} is not oriented along the axes. It presents similar difficulties to f_{Ros}, but its valley is much narrower.
- f_{Ras} is a scalable, continuous, and multimodal function, which is made from f_{sph} by modulating it with $a \cdot \cos(\omega \cdot x_i)$.
- f_{Gri} is a continuous and multimodal function. This function is difficult to optimise because it is nonseparable ([Müh91]) and the search algorithm has to climb a hill to reach the next valley. Nevertheless, one undesirable property exhibited is that it becomes easier as the dimensionality is increased ([Whi96]). In particular, in [Yan00], it is suggested that this function becomes easier when its dimension exceeds 15.

A.2. Real-World Problem

As real-world problem we have introduced a system of linear equations ([Esh97]). The problem to be solved is to obtain the elements of a vector *X*, given the matrix *A* and vector *B* in the expression: *A·X = B*. The evaluation function used for these experiments is:

$$P_{sle}(x_1,...,x_n) = \sum_{i=1}^{n}\sum_{j=1}^{n}(a_{ij} \cdot x_j) - b_j .$$

Clearly, the best value for this objective function is $P_{sle}(x^*) = 0$. Inter-parameter linkage (i.e., non-linearity) is controlled easily in systems of linear equations; their non-linearity does not deteriorate as increasing numbers of parameters are used, and they have proven to be quite difficult.

We have studied an example of a ten-parameter problem. We have considered that $-127 \le x_i \le 127$ and the following matrices:

$$\begin{vmatrix} 5 & 4 & 5 & 2 & 9 & 5 & 4 & 2 & 3 & 1 \\ 9 & 7 & 1 & 1 & 7 & 2 & 2 & 6 & 6 & 9 \\ 3 & 1 & 8 & 6 & 9 & 7 & 4 & 2 & 1 & 6 \\ 8 & 3 & 7 & 3 & 7 & 5 & 3 & 9 & 9 & 5 \\ 9 & 5 & 1 & 6 & 3 & 4 & 2 & 3 & 3 & 9 \\ 1 & 2 & 3 & 1 & 7 & 6 & 6 & 3 & 3 & 3 \\ 1 & 5 & 7 & 8 & 1 & 4 & 7 & 8 & 4 & 8 \\ 9 & 3 & 8 & 6 & 3 & 4 & 7 & 1 & 8 & 1 \\ 8 & 2 & 8 & 5 & 3 & 8 & 7 & 2 & 7 & 5 \\ 2 & 1 & 2 & 2 & 9 & 8 & 7 & 4 & 4 & 1 \end{vmatrix} \begin{vmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{vmatrix} = \begin{vmatrix} 40 \\ 50 \\ 47 \\ 59 \\ 45 \\ 35 \\ 53 \\ 50 \\ 55 \\ 40 \end{vmatrix}$$

A.3. Skewed Initialisation

In our experiments, we have applied a *skewed initialisation* that forces the initial population to be away from the global basin. This is made due to two reasons ([Deb02]):

1. To prevent the advantage enjoyed by algorithms which have inherent tendency to create solutions near the centroid of the parents.
2. To make sure that an algorithm must overcome a number of local minima to reach the global basin when dealing with multimodal functions.

Table A.I shows the ranges associated with the parameters of each test problem to carry out the skewed initialisation of the population.

TABLE A.I
Ranges for Initialisation

f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}
[4, 5]	[-5, -4]	[60, 65]	[4, 5]	[580, 600]	[-120, -100]

Appendix B. Results of U&N-PBX and SPC-PNX with FMD

Tables B.I and B.II show the results obtained from the incorporation process to the U&F-PBX AND SPC-PNX algorithms.

Acknowledgment

This research has been supported by the project TIC2002-04036-C05-01.

TABLE B.I
RESULTS OF U&N-PBX WITH FMD

N_F	N_M	f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}
1	25	1.90e+002	3.36e+005	5.70e+006	5.04e+002	5.90e+002	7.59e+003
1	50	3.19e+001	1.85e+004	1.94e+006	4.61e+002	1.46e+002	4.11e+003
1	100	2.86e-005	4.92e+001	2.29e+005	4.41e+002	2.80e+001	2.21e+003
1	200	2.46e-144	2.13e+001	2.61e+004	4.34e+002	3.57e+000	9.85e+002
1	300	4.92e-116	1.59e+001	8.34e+003	4.16e+002	5.21e+000	7.03e+002
1	400	5.10e-095	1.95e+001	3.06e+003	4.19e+002	7.44e-002	3.90e+002
5	25	9.17e+001	1.75e+005	1.17e+005	4.00e+002	3.14e+002	2.70e+003
5	50	6.21e-146	2.89e+001	2.81e+003	3.50e+002	1.04e+001	5.94e+002
5	100	9.98e-187	1.56e+000	1.74e-009	2.90e+002	1.27e-002	1.64e+002
5	200	2.76e-129	2.33e+000	3.09e-009	2.22e+002	1.41e-002	8.86e+001
5	300	8.99e-098	7.67e+000	1.05e-006	2.06e+002	1.18e-002	5.09e+001
5	400	2.53e-079	1.16e+001	7.60e-005	1.85e+002	1.49e-002	3.82e+001
25	25	9.24e-007	3.27e+003	3.69e+003	2.02e+002	2.22e+001	5.78e+002
25	50	1.64e-130	1.33e+001	1.01e-012	1.34e+002	1.07e-002	1.47e+002
25	100	3.31e-104	9.48e+000	2.68e-009	6.62e+001	6.70e-003	6.50e+001
25	200	1.05e-074	9.80e+000	6.66e-005	4.48e+001	2.56e-003	3.85e+001
25	300	1.24e-059	1.08e+001	3.82e-003	2.37e+001	3.45e-003	2.10e+001
25	400	4.18e-050	1.23e+001	3.43e-002	1.84e+001	2.51e-003	1.46e+001
50	25	5.98e-076	2.53e+001	1.74e-003	1.21e+002	2.67e+000	4.50e+002
50	50	6.19e-095	1.46e+001	3.54e-010	4.97e+001	6.45e-003	8.96e+001
50	100	1.29e-072	1.37e+001	3.85e-006	2.39e+001	4.73e-003	5.61e+001
50	200	1.15e-053	1.38e+001	4.15e-003	1.26e+001	2.86e-003	2.77e+001
50	300	9.83e-044	1.45e+001	9.00e-002	1.04e+001	2.71e-003	1.43e+001
50	400	1.82e-037	1.51e+001	5.36e-001	7.80e+000	1.38e-003	8.59e+000
100	25	1.15e-088	2.45e+001	2.45e-009	5.37e+001	1.84e-002	2.50e+002
100	50	1.29e-071	1.55e+001	1.00e-006	1.94e+001	5.07e-003	1.19e+002
100	100	2.28e-049	1.60e+001	2.20e-003	1.10e+001	3.30e-003	3.32e+001
100	200	5.59e-037	1.65e+001	2.34e-001	6.27e+000	1.58e-003	1.79e+001
100	300	9.01e-031	1.71e+001	1.45e+000	5.32e+000	9.37e-004	1.33e+001
100	400	1.06e-026	1.75e+001	4.58e+000	4.24e+000	1.18e-003	1.05e+001
200	25	2.19e-078	2.53e+001	3.96e-007	1.86e+001	6.99e-003	1.49e+002
200	50	8.56e-052	1.65e+001	6.07e-004	7.92e+000	4.04e-003	6.90e+001
200	100	6.31e-035	1.73e+001	9.10e-002	5.15e+000	2.76e-003	2.02e+001
200	200	1.22e-024	1.83e+001	3.40e+000	3.58e+000	1.53e-003	1.61e+001
200	300	9.95e-021	1.88e+001	1.45e+001	6.64e+000	1.04e-003	1.07e+001
200	400	2.95e-018	1.91e+001	3.12e+001	1.92e+001	4.93e-004	5.45e+000
300	25	3.22e-068	1.85e+001	1.37e-005	1.13e+001	4.78e-003	1.28e+002
300	50	2.72e-044	1.70e+001	1.40e-002	6.81e+000	3.74e-003	5.04e+001
300	100	1.09e-027	1.81e+001	6.89e-001	3.33e+000	1.23e-003	1.94e+001
300	200	5.19e-020	1.89e+001	1.18e+001	6.32e+000	9.86e-004	1.16e+001
300	300	3.17e-016	1.96e+001	3.76e+001	3.80e+001	5.42e-004	9.13e+000
300	400	2.53e-014	1.98e+001	7.35e+001	6.58e+001	5.92e-004	7.45e+000
400	25	3.47e-061	1.87e+001	2.96e-004	8.08e+000	7.44e-003	1.10e+002
400	50	1.28e-039	1.74e+001	9.27e-002	4.16e+000	2.86e-003	3.73e+001
400	100	1.71e-024	1.85e+001	2.53e+000	2.60e+000	3.10e-003	2.43e+001
400	200	7.73e-017	1.94e+001	2.35e+001	3.45e+001	6.41e-004	1.01e+001
400	300	4.93e-014	1.99e+001	7.18e+001	6.72e+001	3.48e-004	8.83e+000
400	400	5.35e-012	2.03e+001	1.38e+002	9.71e+001	1.77e-003	7.84e+000

TABLE B.II
RESULTS OF SPC-PNX WITH FMD

N_F	N_M	f_{sph}	f_{Ros}	f_{Sch}	f_{Ras}	f_{Gri}	P_{sle}
1	25	2.63e+000	1.42e+003	6.96e+005	4.20e+002	2.65e+001	2.00e+003
1	50	9.96e-003	1.43e+002	3.03e+004	3.98e+002	1.42e+001	1.05e+003
1	100	1.08e-071	3.38e+001	8.51e+003	3.91e+002	1.61e-001	7.47e+002
1	200	4.64e-072	1.71e+001	1.92e+003	3.93e+002	1.14e-001	4.95e+002
1	300	7.20e-065	1.37e+001	6.25e+002	3.90e+002	5.88e-002	4.51e+002
1	400	6.95e-055	1.62e+001	2.70e+002	3.89e+002	7.45e-002	3.73e+002
5	25	7.64e-001	1.36e+003	1.85e+004	3.71e+002	3.13e+001	1.14e+003
5	50	1.16e-054	2.37e+001	2.74e+003	3.55e+002	5.33e+000	6.34e+002
5	100	4.07e-072	1.14e+001	3.74e+001	3.28e+002	4.49e-002	4.09e+002
5	200	7.80e-060	9.88e+000	7.69e-003	3.23e+002	2.99e-002	2.69e+002
5	300	7.88e-051	1.09e+001	1.35e-002	3.11e+002	2.88e-002	2.52e+002
5	400	1.67e-043	1.24e+001	6.10e-003	2.89e+002	3.22e-002	2.68e+002
25	25	2.74e-036	2.37e+001	7.63e+000	2.22e+002	7.23e-002	4.71e+002
25	50	1.89e-050	1.78e+001	5.39e-002	1.81e+002	3.10e-002	3.48e+002
25	100	5.34e-044	1.74e+001	2.27e-002	1.44e+002	1.99e-002	2.20e+002
25	200	8.36e-033	1.89e+001	1.55e-001	9.98e+001	1.39e-002	1.84e+002
25	300	7.19e-027	1.87e+001	6.70e-001	9.18e+001	1.52e-002	1.57e+002
25	400	3.55e-023	1.81e+001	1.70e+000	7.58e+001	1.69e-002	1.35e+002
50	25	2.73e-050	2.14e+001	9.79e-002	1.87e+002	2.47e-002	4.04e+002
50	50	8.10e-035	1.94e+001	3.40e-001	8.81e+001	2.04e-002	2.13e+002
50	100	6.99e-030	1.94e+001	5.62e-001	6.91e+001	1.27e-002	1.58e+002
50	200	1.81e-022	1.92e+001	3.24e+000	5.25e+001	1.18e-002	1.17e+002
50	300	1.81e-018	1.91e+001	1.20e+001	4.31e+001	9.60e-003	1.25e+002
50	400	5.39e-016	2.05e+001	2.34e+001	3.58e+001	9.65e-003	1.10e+002
100	25	5.77e-044	1.80e+001	3.14e-002	1.63e+002	2.23e-002	2.57e+002
100	50	1.85e-030	1.99e+001	5.59e-001	7.73e+001	1.98e-002	1.40e+002
100	100	7.34e-018	1.98e+001	1.76e+001	3.70e+001	1.60e-002	9.63e+001
100	200	1.16e-014	2.03e+001	4.59e+001	2.85e+001	8.91e-003	8.62e+001
100	300	3.41e-012	2.04e+001	8.57e+001	2.52e+001	6.50e-003	8.25e+001
100	400	1.24e-010	2.07e+001	1.48e+002	2.29e+001	6.75e-003	7.46e+001
200	25	2.79e-033	1.70e+001	1.26e-001	1.11e+002	1.71e-002	1.95e+002
200	50	9.02e-023	1.94e+001	3.32e+000	5.04e+001	1.20e-002	1.35e+002
200	100	8.27e-015	2.02e+001	4.49e+001	3.13e+001	7.68e-003	8.02e+001
200	200	2.87e-008	2.10e+001	3.38e+002	2.45e+001	9.45e-003	4.51e+001
200	300	1.49e-007	2.13e+001	4.42e+002	3.93e+001	7.04e-003	6.34e+001
200	400	1.09e-006	2.15e+001	5.78e+002	4.99e+001	3.15e-003	5.14e+001
300	25	3.62e-027	1.89e+001	5.98e-001	8.70e+001	1.25e-002	1.47e+002
300	50	1.39e-018	1.94e+001	1.10e+001	4.53e+001	9.60e-003	1.18e+002
300	100	3.07e-012	2.04e+001	9.15e+001	2.44e+001	4.97e-003	8.14e+001
300	200	1.45e-007	2.13e+001	4.71e+002	3.40e+001	4.43e-003	5.44e+001
300	300	4.18e-005	2.21e+001	1.02e+003	9.53e+001	4.72e-003	4.25e+001
300	400	8.03e-005	2.23e+001	1.09e+003	1.06e+002	5.79e-003	6.40e+001
400	25	2.57e-023	1.80e+001	1.46e+000	7.49e+001	1.19e-002	1.56e+002
400	50	4.17e-016	1.94e+001	2.43e+001	3.25e+001	8.66e-003	1.19e+002
400	100	1.15e-010	2.06e+001	1.43e+002	2.15e+001	5.02e-003	6.28e+001
400	200	1.11e-006	2.15e+001	6.17e+002	5.30e+001	3.84e-003	6.01e+001
400	300	8.62e-005	2.22e+001	1.12e+003	1.09e+002	3.83e-003	4.15e+001
400	400	1.76e-003	2.39e+001	1.68e+003	1.31e+002	7.60e-002	5.43e+001

REFERENCES

[Arf01] Arfiadi, Y., Hadi, M.N.S. (2001). Optimal Direct (Static) Output Feedback Controller Using Real Coded Genetic Algorithms. Computers & Structures 79(17): 1625-1634.

[Aza02] Azariadis, P.N., Nearchou, A.C., Aspragathos, N.A. (2002). An Evolutionary Algorithm for Generating Planar Developments of Arbitrarily Curved Surfaces. Computers in Industry 47(3): 357-368.

[Bäc94] Bäck, T. (1994). Selective pressure in evolutionary algorithms: a characterization of selection mechanisms. In: Proceedings of the First IEEE Conference on Evolutionary Computation, pp 57-62.

[Bäc96] Bäck, T. (1996). Evolutionary Algorithms in Theory and Practice, Oxford.

[Bal03] Ballester, P.J., Carter, J.N. (2003). Real-Parameter Genetic Algorithms for Finding Multiple Optimal Solutions in Multi-modal Optimization. In: Erick Cantú-Paz et al. (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference, Springer, LNCS 2723, pp. 706-717.

[Bal04] Ballester, P.J., Carter, J.N. (2004). An Effective Real-Parameter Genetic Algorithm with Parent Centric Normal Crossover for Multimodal Optimisation. Proceedings of the Genetic and Evolutionary Computation Conference, Springer, LNCS 3102, pp. 901-913.

[Ban98] Bandyopadhyay, S., Pal, S.K., Maulik, U. (1998). Incorporating chromosome differentiation in genetic algorithms. Information Sciences 104: 293-319.

[Bas03] Baskar, S., Subbaraj, P., Rao, M.V.C. (2003). Hybrid real coded genetic algorithm solution to economic dispatch problem. Computers & Electrical Engineering 29(3): 407-419.

[Bey01] Beyer, H.-G., Deb, K. (2001). On self-adaptive features in real-parameter evolutionary algorithms. IEEE Transactions on Evolutionary Computation 5(3): 250-270.

[Bla01] Blanco, A., Delgado, M., Pegalajar, M.C. (2001). A Real-Coded Genetic Algorithm for Training Recurrent Neural Networks. Neural Networks 14(1): 93-105.

[Cao00] Caorsi, S., Massa, A., Pastorino, M. (2000). A Computational Technique Based on a Real-Coded Genetic Algorithm for Microwave Imaging Purposes. IEEE Transactions on Geoscience and Remote Sensing 38(4): 1697-1708.

[Cha98] Chang, F.-J. (1998). Real-Coded Genetic Algorithm for Rule-Based Flood Control Reservoir Management. Water Resources Management 12(3): 185-198.

[Che03] Chelouah, R., Siarry, P. (2003). Genetic and Nelder–Mead algorithms hybridized for a more accurate global optimization of continuous multiminima functions. European Journal of Operational Research 148: 335-348.

[Dam03] Damousis, I.G., Bakirtzis, A.G., Dokopoulos, P.S. (2003). Network-constrained economic dispatch using real-coded genetic algorithm. IEEE Transactions on Power Systems 18(1): 198 – 205.

[Deb95] Deb K., Agrawal, R.B. (1995). Simulated binary crossover for continuous search space. Complex System 9: 115-148.

[Deb01a] Deb, K. (2001). Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons, New York.

[Deb01b] Deb, K., Beyer, H. (2001). Self-adaptive genetic algorithms with simulated binary crossover. Evolutionary Computation Journal 9(2): 195-219.

[Deb02] Deb, K., Anand, A., Joshi, D. (2002). A computationally efficient evolutionary algorithm for real-parameter evolution. Evolutionary Computation Journal 10(4): 371-395.

[Deb05] Deb, K. (2005). A population-based algorithm-generator for real-parameter optimization. Solf-computing, 2005. In press.

[DeJ75] De Jong, K.A. (1975). An Analysis of the Behavior of a Class of Genetic Adaptive Systems. Doctoral dissertation, University of Michigan, Ann Arbor. Dissertation Abstracts International, 36(10), 5140B (University Microfilms No 76-9381).

[DeJ92] De Jong, K.A., Spears, W.M. (1992). A formal analysis of the role of multi-point crossover in genetic algorithms. Annals of Mathematics and Artificial Intelligence 5(1): 1-26.

[DeJ93] De Jong, K.A., Sarma, J. (1993). Generation gaps revisited. In: L.D. Whitley (Ed.), Foundations of Genetic Algorithms 2, Morgan Kaufmann, San Mateo, California, pp. 19-28.

[Duf01] Duffy, J., McNelis, P.D. (2001). Approximating and Simulating the Stochastic Growth Model: Parameterized Expectations, Neural Networks, and the Genetic Algorithm. Journal of Economic Dynamics and Control 25(9): 1273-1303.

[Eib99] Eiben, A.E., Hinterding, R., Michalewicz, Z. (1999). Parameter Control in Evolutionary Algorithms. IEEE Transactions on Evolutionary Computation 3(2): 124-141.

[Esh91] Eshelman, L.J. (1991). The CHC adaptive search algorithm: how to have safe search when engaging in non-traditional genetic recombination. In: Rawlin, G.J.E., (Ed.), Foundations of Genetic Algorithms 1, Morgan Kaufmann, San Mateo, California, pp. 265-283.

[Esh97] Eshelman, L.J. Mathias, K.E., Schaffer, J.D. (1997). Convergence controlled variation. In: Belew, R. and Vose, M., (Eds.), Foundations of Genetic Algorithms 4, Morgan Kaufmann, San Mateo, California, pp 203-224.

[Fer01] Fernandes, C., Rosa, A. (2001). A Study on non-random mating and varying population size in genetic algorithms using a royal road function. Proceedings of the 2001 Congress on Evolutionary Computation, IEEE Press, Piscataway, New Jersey, pp. 60-66.

[Fog94] Fogel, D.B. (1994). Evolutionary programming: an introduction and some current directions. Statistics and Computing 4: 113-129.

[Fog95] Fogel, D.B. (1995). Evolutionary Computation: Toward a New Philosophy of Machine Intelligence. IEEE Press, Piscataway.

[Goh03] Goh, K.S., Lim, A., Rodrigues, B. (2003). Sexual selection for genetic algorithms. Artificial Intelligence Reviews 19: 123-152.

[Gol89] Goldberg, D.E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading, MA.

[Gri81] Griewangk, A.O. (1981). Generalized descent of global optimization. Journal of Optimization Theory and Applications 34: 11-39.

[Haj02] Hajela, P. (2002). Soft Computing in Multidisciplinary Aerospace Design-New Directions for Research. Progress in Aerospace Sciences 38(1): 1-21.

[Har98] Harris, S.P., Ifeachor, E.C. (1998). Automatic Design of Frequency Sampling Filters by Hybrid Genetic Algorithm Techniques. IEEE Transactions on Signal Processing 46(12): 3304-3314.

[Her98] Herrera, F., Lozano, M., Verdegay, J.L. (1998). Tackling real-coded genetic algorithms: operators and tools for the behavioral analysis. Artificial Intelligence Reviews 12(4): 265-319.

[Her00] Herrera, F., Lozano, M. (2000). Gradual distributed real-coded genetic algorithms. IEEE Transactions on Evolutionary Computation 4(1): 43-63.

[Her03] Herrera, F., Lozano, M., Sánchez, A.M. (2003). A taxonomy for the crossover operator for real-coded genetic algorithms. An experimental study. International Journal of Intelligent Systems 18(3): 309-338.

[Jai04] Jain, A., Srinivasalu, S., Bhattacharjya, R.K. (2004). Determination of an optimal unit pulse response function using real-coded genetic algorithm. Journal of Hydrology. In Press.

[Kaz01] Kazarlis, S.A., Papadakis, S.E., Theocharis, J.B., Petridis, V. (2001). Microgenetic algorithms as generalized hill-climbing operators for GA Optimization. IEEE Transaction on Evolutionary Computation 5(3): 204-217.

[Kit01] Kita, H. (2001). A comparison study of self-adaptation in evolution strategies and real-coded genetic algorithms. Evolutionary Computation Journal 9(2): 223-241.

[Kuo04] Kuo, R.J., Chen, J.A. (2004). A decision support system for order selection in electronic commerce based on fuzzy neural network supported by real-coded genetic algorithm. Expert Systems with Applications 26(2): 141-154.

[Lei04] Leite, J.V., Avila, S.L., Batistela, N.J., Carpes, W.P., Sadowski, N., Kuo-Peng, P., Bastos, J.P.A. (2004). Real coded genetic algorithm for Jiles-Atherton model parameters identification. IEEE Transactions on Magnetics 40(2): 888 – 891.

[Lin04] Lin, F-J., Shieh, H-J., Shyu, K-K., Huang, P-K. (2004). On-line gain-tuning IP controller using real-coded genetic algorithm. Electric Power Systems Research 72(2): 157-169.

[Liu04] Liu, Z., Liu, A., Wang, C., Niu, Z. (2004). Evolving neural network using real coded genetic algorithm (GA) for multispectral image classification. Future Generation Computer Systems 20(7): 1119-1129.

[Loz04a] Lozano M., Herrera F., Krasnogor N., Molina D. (2004). Real-coded memetic algorithms with crossover hill-climbing. Evolutionary Computation Journal 12(3): 273-302.

[Loz04b] Lozano M., Herrera F., (2004). Diversification techniques as a way to improve the performance of real-parameter crossover operators: the case of the parent-centric BLX. Submitted.

[Mat99] Matsui, K. (1999). New selection method to improve the population diversity in genetic algorithms. In: Proceedings IEEE International Conference on Systems, Man, and Cybernetics, pp. 625 -630.

[Mic92] Z. Michalewicz. (1992). Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag: New York.

[Mos99] Moscato, P.A. (1999). Memetic algorithms: a short introduction. In: D. Corne, M. Dorigo and R. Glower, (Eds.), New Ideas in Optimization, McGraw-Hill, London, pp. 219-234.

[Müh91] Mühlenbein, H., Schomisch, M., Born, J. (1991). The parallel genetic algorithm as function optimizer. In: R. Belew, L.B. Booker, (Eds.), Fourth International Conference on Genetic Algorithms, Morgan Kaufmann, San Mateo, California, pp. 271-278.

[Ren96] Renders, J-M., Flasse, S.P. (1996). Hybrid methods using genetic algorithms for global optimisation. IEEE Transactions on Systems, Man, and Cybernetics 26(2): 243-258.

[Rou99] Roubous, J.A., van Straten, G., van Boxtel, A.J.B. (1999). An Evolutionary Strategy for Fed-Batch Bioreactor Optimization; Concepts and Performance. Journal of Biotechnology 67(2-3): 173-187.

[Sch94] Schlierkamp-Voosen, D., Mühlenbein, H. (1994). Strategy adaptation by competing subpopulations. In: Y. Davidor, H.-P. Schwefel, R. Männer, (Eds.), Parallel Problem Solving from Nature 3, Springer-Verlag, Berlin, Germany, pp. 199-208.

[Sch81] Schwefel, H-P. (1981). Numerical Optimization of Computer Models, Chichester, Wiley.

[Sch95] Schwefel, H-P. (1995). Evolution and Optimum Seeking. Sixth-Generation Computer Technology Series, Wiley, New York.

[Shi02] Shi, K.L., Chan, T.F., Wong, Y.K., Ho, S.L. (2002). Speed Estimation of an Induction Motor Drive using an Optimized Extended Kalman Filter. IEEE Transactions on Industrial Electronics 49(1): 124-133.

[Sol81] Solis, F.J., Wets, R.J.B. (1981). Minimization by random search techniques. Mathematical Operations Research 6: 19-30.

[Sys89] Syswerda, G. (1989). Uniform crossover in genetic algorithms. In: J. David Schaffer (Ed.), Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmann, San Mateo, pp. 2-9.

[Tak01] Takahashi, O., Kobayashi, S. (2001). An adaptive neighboring search using crossover-like mutation for multi modal function optimisation. IEEE International Conference on Systems, Man, and Cybernetics, pp. 261-267.

[Tör89] Törn, A, Antanas, Z. (1989). Global Optimization. Springer, LNCS 350.

[Tur02] Turcanu, C.-O., Craciunescu, T. (2002). A Genetic Approach to Limited Data Tomographic Reconstruction of Time-Resolved Energy Spectrum of Short-Pulsed Neutron Sources. Pattern Recognition Letters 23(8): 967-976.

[Voi95] Voigt, H.M., Mühlenbein, H., Cvetkovic, D. (1995). Fuzzy recombination for the breeder genetic algorithm. In: L. Eshelman (Ed.), Proceedings of the Sixth International Conference on Genetic Algorithms, Morgan Kaufmann, San Francisco, pp. 104-111.

[Whi89] Whitley, D. (1989). The GENITOR algorithm and selection pressure: why rank-based allocation of reproductive trials is best. In: J. David Schaffer (Ed.), Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmann, San Mateo, pp. 116-121.

[Whi96] Whitley, D., Rana, S., Dzubera, J., Mathias, E. (1996). Evaluating evolutionary algorithms. Artificial Intelligence 85: 245-276.

[Yan00] Yang, J.-M., Kao, C.-Y. (2000). Integrating adaptive mutations and family competition into genetic algorithms as function optimiser. Soft Computing 4: 89-102.

[Yos03] Yoshimoto, F., Harada, T., Yoshimoto, Y. (2003). Data fitting with a spline using a real-coded genetic algorithm. Computer-Aided Design 35(8): 751-760.