

MULTIDIMENSIONAL CROSSOVER IN GENETIC ALGORITHMS

MÁRTON-ERNŐ BALÁZS*

1. INTRODUCTION

Most of the genetic algorithms (GA's) used in practice work on linear chromosomes (e.g. binary strings or sequences of some other types of symbols). However, some results have been published revealing that for certain problems multidimensional encoding and crossover may give better results than the one dimensional (linear) ones [1]–[3]. While some theoretical results have been obtained, no clear criteria are known for deciding the suitable dimensionality of the encoding to be used for a given problem.

In this paper we consider a class of problems for which we define a multidimensional encoding and a corresponding genetic operator. We show that for a GA using this encoding and operator we can obtain theoretical results similar to (under certain conditions even better than) those known for linear encoding.

Finally we demonstrate these theoretical results using a set of test examples.

2. MULTIDIMENSIONAL ENCODING AND CROSSOVER

In this section we first define what we mean by a multidimensional (search) problem. We then briefly present the classical (one dimensional) encoding used in solving such problems using GA's, followed by our proposal for a multidimensional encoding. Next we define a crossover operator on multidimensional codes and prove a Schema Theorem-like result for a GA using it. Finally we show that for certain types of problems multidimensional encoding and the defined crossover operator may give better results than the linear ones.

*Presently at Worcester Polytechnic Institute, Department of Computer Science, 100 Institute Rd., Worcester, MA 01609, USA.

2.1. MULTIDIMENSIONAL PROBLEMS

Our definition of multidimensional problems is based on the following definition of n -ary relations:

DEFINITION 2.1.1. Let A_1, A_2, \dots, A_n be sets. An n -ary relation ρ over these sets is a set

$$\rho \subseteq A_1 \times A_2 \times \dots \times A_n.$$

In the followings we shall only consider relations defined over finite sets. The set of all relations over the sets A_1, A_2, \dots, A_n will be denoted by $R(A_1, A_2, \dots, A_n)$.

Let us note here that an n -ary relation ρ over A_1, A_2, \dots, A_n can be represented by an n -dimensional matrix $M(\rho) = (m_{i_1, i_2, \dots, i_n})$ over $\{0, 1\}$ as follows:

Let $A_i := \{a_i^1, a_i^2, \dots, a_i^{k_i}\}$ for $i = \overline{1, n}$. Then

$$m_{i_1, i_2, \dots, i_n} = \begin{cases} 1 & \text{if } (a_{i_1}^{j_1}, a_{i_2}^{j_2}, \dots, a_{i_n}^{j_n}) \in \rho \\ 0 & \text{otherwise} \end{cases}$$

For the two-dimensional case this corresponds to the adjacency matrix associated with the graph corresponding to the relation.

DEFINITION 2.1.2. Given the (finite) sets A_1, A_2, \dots, A_n by an n -dimensional (search) problem attached to a predicate P defined over $R = R(A_1, A_2, \dots, A_n)$ we mean the problem of finding an n -ary relation $\rho^* \in R(A_1, A_2, \dots, A_n)$ such that $P(\rho^*)$ is true.

Our goal in the following is to study some aspects of using genetic algorithms for solving n -dimensional problems. To be able to do this we assume that for every n -dimensional search problem considered there is a fitness function $f: R \rightarrow \mathcal{R}^+$, which measures the "quality" of any given relation. For our purposes the nature of this function is irrelevant.

2.2. THE COMMON APPROACH

In using GA's to solve problems of the above mentioned type the current practice is to choose a (binary) coding for each of the dimensions of a relation and concatenate them. One-point crossover of two relations then means choosing a crossover point, cutting both of the codes in that point and exchanging their "tails"; producing two offsprings. Other, multi-point crossover operations, have been also defined [4], [5], however none of them is able to provide the type of improvement we are looking for.

Intuitively the drawback of these approaches is that they don't take into consideration the dimensionality of the problem, i.e. they don't try to preserve sub-relations of same arity of the original relations [1], [3].

2.3. n -DIMENSIONAL ENCODING AND CROSSOVER

In this section we propose another approach to using GA's for solving multidimensional search problems. Similar approaches have been proposed in [1], [2], [3] and [7]. Our approach is based on the use of multidimensional matrices for encoding and a crossover operator specific to this encoding. This was first defined in [1] and is similar to the encoding used by algorithm R1 in [3].

To make our discussion clear we first present our results for two dimensional problems, the generalization to n -dimensional ones being discussed in a separate subsection.

Two-dimensional Encoding and One-Point Crossover

Let $A = \{a_1, a_2, \dots, a_m\}$ and $B = \{b_1, b_2, \dots, b_n\}$ be finite sets, and let us consider the two dimensional search problem attached to a given predicate P over $R(A, B)$. To solve this problem using a genetic algorithm we have to choose a representation for the relations in P . We propose that the representation of a relation $\rho \in R(A, B)$ be the corresponding $M(\rho)$ matrix as defined in Subsection 2.1. In this case a two dimensional matrix representing a relation will be a chromosome processed by the genetic algorithm. Let us now define a crossover operator for a GA using the above considered encoding.

DEFINITION 2.3.1. Let $\rho_1, \rho_2 \in R(A, B)$ be two relation and $M(\rho_1)$ and $M(\rho_2)$ the corresponding chromosomes. Let us further consider two positive integers k_1 and k_2 such that $1 \leq k_1 \leq m$ and $1 \leq k_2 \leq n$. The two-dimensional crossover of the two chromosomes using crossover point (k_1, k_2) produces two new chromosomes $M_1 = (m_{i,j}^1)$ and $M_2 = (m_{i,j}^2)$ defined by

$$m_{i,j}^1 = \begin{cases} r_{i,j} & \text{for } i < k_1 \wedge j < k_2 \text{ or } i \geq k_1 \wedge j \geq k_2 \\ q_{i,j} & \text{for } i < k_1 \wedge j \geq k_2 \text{ or } i \geq k_1 \wedge j < k_2 \end{cases}$$

$$m_{i,j}^2 = \begin{cases} q_{i,j} & \text{for } i < k_1 \wedge j < k_2 \text{ or } i \geq k_1 \wedge j \geq k_2 \\ r_{i,j} & \text{for } i < k_1 \wedge j \geq k_2 \text{ or } i \geq k_1 \wedge j < k_2 \end{cases}$$

respectively.

Example. Let us consider the following two-dimensional chromosomes:

1	0	1	1	0	0	1	1	1	0
0	1	1	0	1	1	1	0	0	0
0	0	0	1	1	1	0	0	0	0
1	1	0	1	0	1	1	1	0	0
0	1	0	1	1	0	0	1	1	1

and crossover point (4, 3). Applying the two dimensional crossover to the above chromosomes the newly created chromosomes will be

1	0	1	1	0	0	1	1	1	0
0	1	0	0	0	1	1	1	0	1
0	0	0	0	0	1	0	0	1	1
1	1	0	1	0	1	1	1	0	0
0	0	1	0	1	0	1	0	1	1

The Two-Dimensional Schema Theorem

In order to give our first result we need some more definitions similar to those given for the classical Schema Theorem.[4]-[5].

DEFINITION 2.3.2. A two-dimensional schema of size $m \times n$ is a matrix of dimension $m \times n$ over $\{0, 1, *\}$.

As in the Schema Theorem "*" means "either 0 or 1". We call the positions having values 0 or 1 *specific positions*.

A two-dimensional schema represents a set of matrices. More precisely: a two-dimensional schema with specific positions $S(\sigma) = \{(i_1, j_1), (i_2, j_2), \dots, (i_k, j_k)\}$ defines the following matrix set:

$$\sigma = \left\{ (a_{i,j})_{i=1..m, j=1..n} \mid a_{i,j} = \sigma_{i,j}, \forall (i,j) \in S(\sigma) \right\}$$

By analogy with the one-dimensional Schema Theorem we also need the following definitions:

DEFINITION 2.3.3. The defining matrix of a schema σ is the smallest sub-matrix of σ that contains all its specific positions $(\Delta(\sigma))$.

DEFINITION 2.3.4. If the dimensions of $\Delta(\sigma)$ are d_1 and d_2 respectively, then (δ_1, δ_2) is the defining size of σ , where $\delta_1 = d_1 - 1$ and $\delta_2 = d_2 - 1$.

Example. Let us consider the 5 by 5 quadratic chromosome:

*	*	*	*	*
*	*	*	*	*
*	0	*	*	1
*	*	1	0	*
*	*	*	*	*

The defining matrix of this chromosome is the sub-matrix defined by the positions (3, 3) and (4, 5), while its defining size is (1, 2).

Similar to the Schema Theorem presented in Goldberg's book and using the same notations we have the following results:

THEOREM 2.3.1. (two-dimensional Schema-Theorem) *If σ is a two-dimensional schema of size $n_1 \times n_2$, then*

$$n(\sigma, t+1) \geq n(\sigma, t) \cdot \frac{f(\sigma, t)}{f(t)} \cdot \left[1 - p_c \cdot \frac{\delta_1 \cdot n_2 + \delta_2 \cdot n_1 - \delta_1 \cdot \delta_2}{(n_1 \cdot n_2 - 1)} - w(\sigma) p_m \right]$$

Proof. In most of its parts the proof of this theorem is identical to that of the one-dimensional Schema Theorem. Thus we will only insist on the single significant difference which is the estimation of the survival probability of a schema from one generation to the next one.

It is easy to observe that there are $\delta_1 \cdot n_2 + \delta_2 \cdot n_1 - \delta_1 \cdot \delta_2$ crossover points in an $n \times m$ chromosome that disrupt a schema of dimension $\delta(\sigma) = (\delta_1, \delta_2)$.

This allows us to state that the survival probability of a schema σ of defining size $\delta(\sigma) = (\delta_1, \delta_2)$ is given by

$$p_s(\sigma) = \frac{\delta_1 \cdot n_2 + \delta_2 \cdot n_1 - \delta_1 \cdot \delta_2}{n^2 - 1}$$

The remaining part of the proof is identical to the proof of the one-dimensional Schema Theorem. \square

This theorem only shows that a GA using the chosen coding and the above defined crossover operator behaves in a similar way to the one using linear coding and crossover.

Crossover: Two dimensional vs. One dimensional

In the previous subsection we showed that a genetic algorithm using the two-dimensional encoding and crossover operator defined earlier behaves the same way as one using a linear encoding and single-point crossover.

In the followings we shall study in what situations the use of this encoding and crossover can be advantageous. To do this let us consider a schema of dimensions (n_1, n_2) with the defining matrix given by (i_1, j_1) and (i_2, j_2) , where

$i_1 < i_2$ and $j_1 < j_2$. We shall estimate the probability with which this schema will be disrupted on one hand if the schema is rearranged by rows and a one-point linear crossover is applied, on the other hand if the two-dimensional crossover is used.

We illustrate our reasoning on the following example. Let us consider again the schema used earlier:

```

* * * * *
* * * * *
* * 0 * 1
* * 1 0 *
* * * * *
    
```

If we rearrange this schema by rows we obtain the following linear schema:

```

* * * * * * * * * * 0 * 1 * * 1 0 * * * * *
    
```

We can note that the defining length of the linear schema is $\delta(\sigma) = \delta_i n_2 + \delta_j$.

Generally we can state that the probability $p_d(\sigma)$ that a two-dimensional schema σ of dimension (n_1, n_2) and defining size (δ_i, δ_j) is disrupted by a single point crossover applied to its linear encoding satisfies the following double inequality:

$$\frac{\delta_i n_2 - \delta_i}{n_1 \cdot n_2 - 1} \leq p_d(\sigma) \leq \frac{\delta_i n_2 + \delta_i}{n_1 \cdot n_2 - 1}$$

On the other hand we have already seen that the probability that the same schema is disrupted by a two-dimensional crossover is given by

$$\frac{\delta_i \cdot n_2 + \delta_j \cdot n_1 - \delta_i \cdot \delta_j}{n_1 \cdot n_2 - 1}$$

In [2] we showed that for certain schemata (e.g. column ones) the disruption probability is smaller when using two-dimensional encoding and crossover than when using the one-dimensional ones. This leads us to the conclusion that for arbitrary two-dimensional problems the use of two-dimensional encoding and crossover should be preferred to the one-dimensional ones.

2.4. MULTIDIMENSIONAL ENCODING AND CROSSOVER

In this section we only present the results obtained for the n -dimensional case. The reasoning behind these results is similar to the two-dimensional case, the difference consisting only in the complexity of the computations needed.

To give an intuitive idea of how crossover is defined in the n -dimensional case we illustrate it graphically in figure for the three-dimensional case (see Fig. 1).

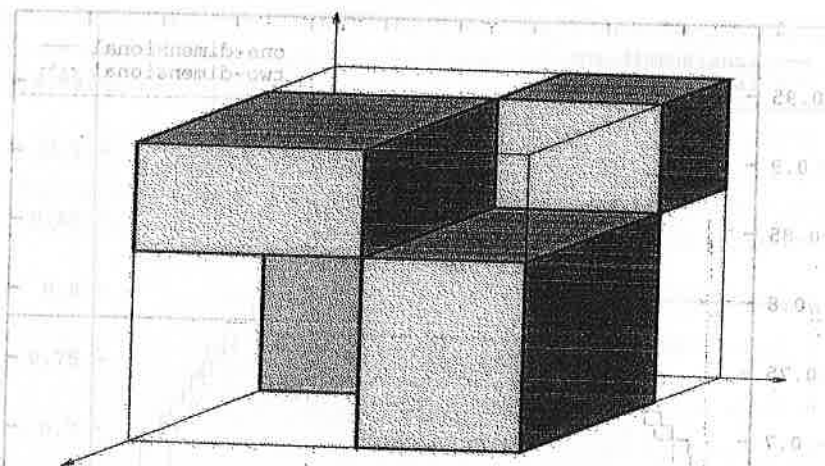


Fig. 1.

It is easy to show that the disruption probability for n -dimensional case is given by

$$p_d = \frac{\sum_{i=1}^n \delta_1 \dots \delta_{i-1} d_i \delta_{i+1} \dots \delta_n - (n-1) \prod_{i=1}^n \delta_i}{\prod_{i=1}^n d_i}$$

Based on this the n -dimensional Schema Theorem (using similar definitions and notations as above) can be formulated as follows:

THEOREM 2.4.1. *If σ is an n -dimensional schema of size $d_1 \times d_2 \dots \times d_n$, then*

$$n(\sigma, t+1) \geq n(\sigma, t) \cdot \frac{f(\sigma, t)}{f(t)} [1 - p_c \cdot p_d - \omega(\sigma) p_m]$$

The comparison to one-dimensional encoding and crossover presented in the previous section can be easily extended to the n -dimensional case.

3. EXPERIMENTAL RESULTS

To demonstrate our theoretical results we performed a set of experiments on selected two-dimensional problems. All of these problems required to find homogeneous binary relations, that is relations $\rho \subseteq A \times A$, where A is a finite set ($A = \{a_1, a_2, \dots, a_n\}$). The relations to be found were chosen such that reordering by rows or columns of the adjacency matrix doesn't provide an advantage to linear crossover.

For the experiments we used a steady-state GA with a population size of 100, crossover rate of 0.6 and mutation rate of 0.001. The fitness function was simply calculated by counting the positions in a chromosome that matched the positions in the target chromosome.

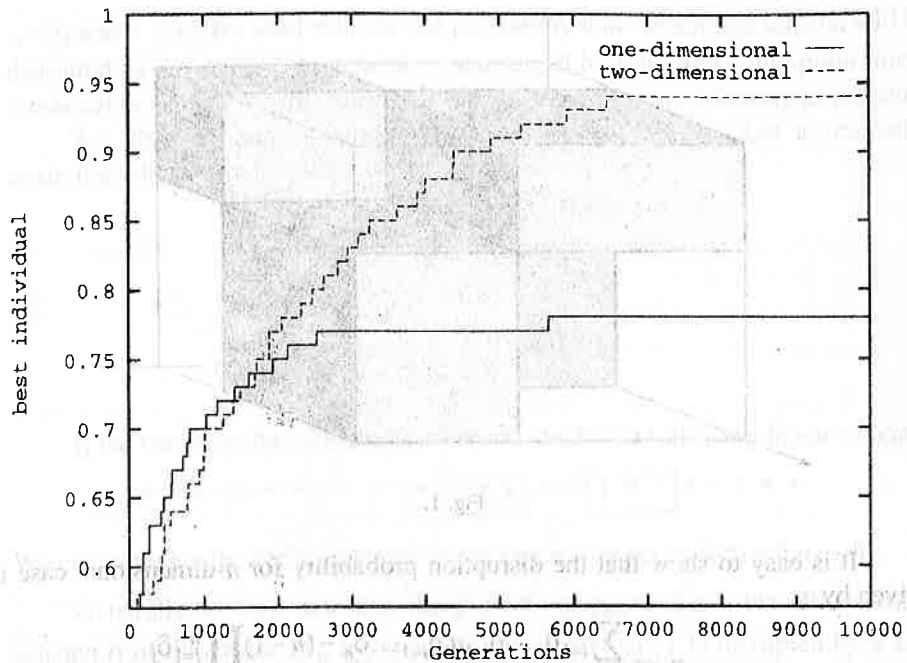


Fig. 2(a). -- "First Diagonal" relation.

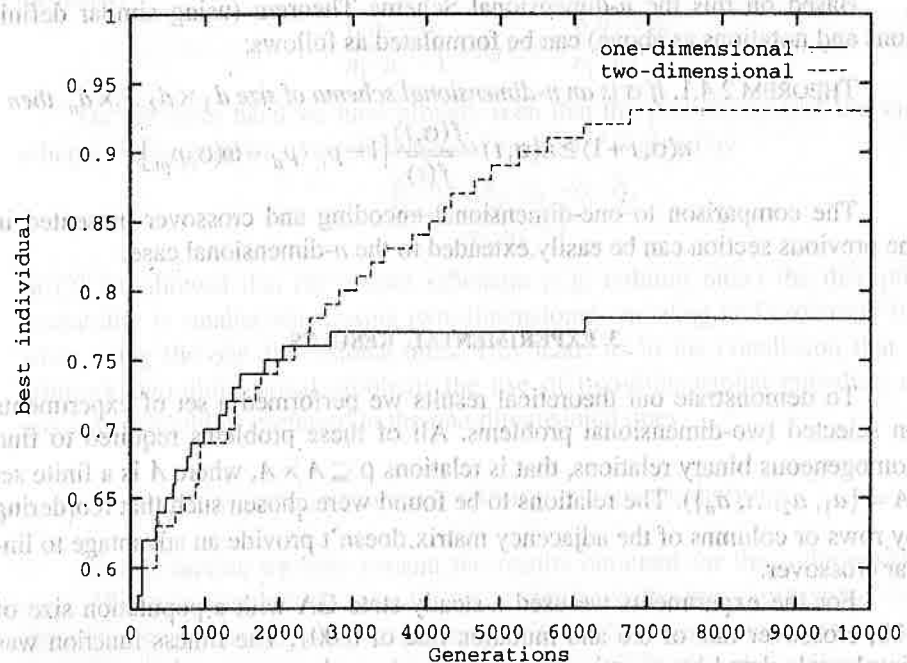


Fig. 2(b). -- "Second Diagonal" relation.

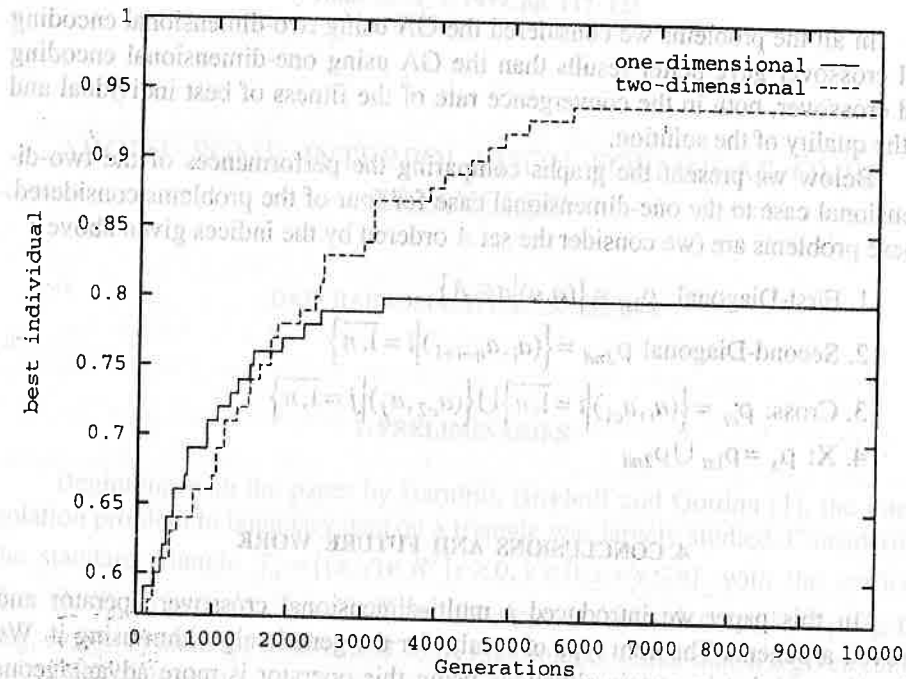


Fig. 2(c). -- "Cross" relation.

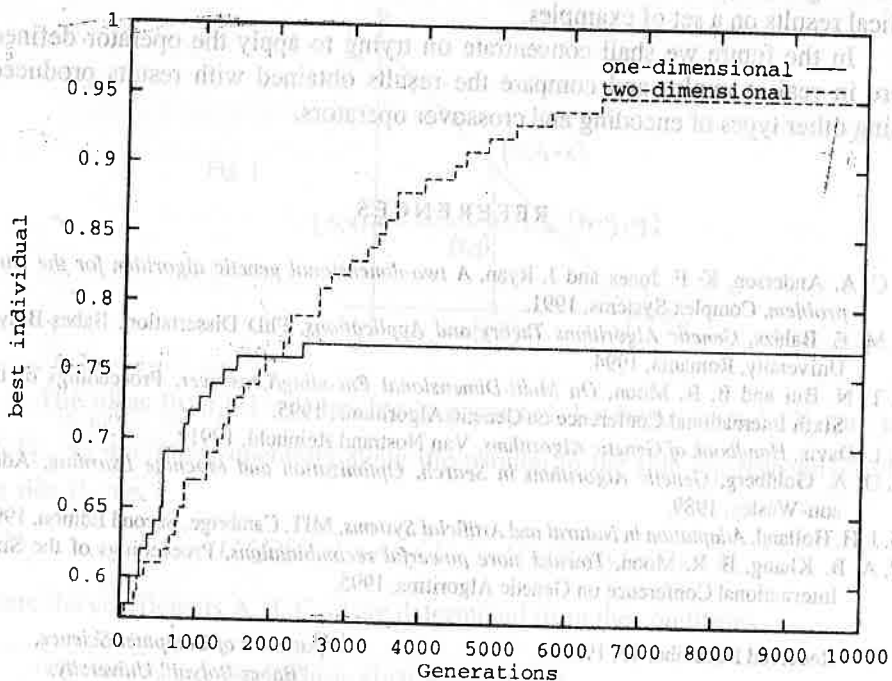


Fig. 2(d). -- "X" relation.

In all the problems we considered the GA using two-dimensional encoding and crossover gave better results than the GA using one-dimensional encoding and crossover, both in the convergence rate of the fitness of best individual and in the quality of the solution.

Below we present the graphs comparing the performances of the two-dimensional case to the one-dimensional case for four of the problems considered. These problems are (we consider the set A ordered by the indices given above):

1. First-Diagonal: $\rho_{1st} = \{(a, a) | a \in A\}$
2. Second-Diagonal $\rho_{2nd} = \{(a_i, a_{n-i+1}) | i = \overline{1, n}\}$
3. Cross: $\rho_{cr} = \{(a_i, a_{c1}) | i = \overline{1, n}\} \cup \{(a_{c2}, a_j) | j = \overline{1, n}\}$
4. X: $\rho_x = \rho_{1st} \cup \rho_{2nd}$

4. CONCLUSIONS AND FUTURE WORK

In this paper we introduced a multi-dimensional crossover operator and proved a Schema-Theorem type of results for the genetic algorithm using it. We also showed that in certain situations using this operator is more advantageous than using the classical, one-dimensional one. Finally we demonstrated this theoretical results on a set of examples.

In the future we shall concentrate on trying to apply the operator defined here in real examples and compare the results obtained with results produced using other types of encoding and crossover operators.

REFERENCES

1. C. A. Anderson, K. F. Jones and J. Ryan, *A two-dimensional genetic algorithm for the ising problem*, Complex Systems, 1991.
2. M. E. Balázs, *Genetic Algorithms Theory and Applications*, PhD Dissertation, Babeş-Bolyai University, Romania, 1994.
3. T. N. Bui and B. R. Moon, *On Multi-Dimensional Encoding/Crossover*, Proceedings of the Sixth International Conference on Genetic Algorithms, 1995.
4. L. Davis, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, 1991.
5. D. A. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.
6. J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT, Cambridge, Second Edition, 1992.
7. A. B. Khang, B. R. Moon, *Toward more powerful recombinations*, Proceedings of the Sixth International Conference on Genetic Algorithms, 1995.

Received December 15, 1996

Department of Computer Science,
"Babeş-Bolyai" University,
3400 Cluj-Napoca, Romania