In the Proceedings of the 1999 Congress on Evolutionary Computation (CEC 99), pp. 1434-1441, Washington, USA, 6-9 July, 1999. NJ:IEEE Press.

Enhancing Transposition Performance

Anabela Borges Simões
Centre for Informatics and Systems
of the University of Coimbra
Polo II - Pinhal de Marrocos
3030 Coimbra - Portugal
abs@eden.dei.uc.pt

Ernesto Costa
Centre for Informatics and Systems
of the University of Coimbra
Polo II - Pinhal de Marrocos
3030 Coimbra - Portugal
ernesto@dei.uc.pt

Abstract - Transposition is a new genetic operator alternative to crossover and allows a classical GA to achieve better results. This mechanism characterized by the presence of mobile genetic units must be used with the right parameters to enable maximum performance to the GA. This paper presents the results of an empirical study which offers the main guidelines to choose the proper setting of parameters to use with transposition, which will lead the GA to the best solutions.

1 Introduction

A Genetic Algorithm (GA) is an iterative search process that allows the search for solutions to a given problem in an intractable space. They are inspired in the biological processes of genetics and evolution, based on the Darwinian principle of the survival of the fittest. A randomly created initial population of candidates solutions to a given problem evolves through several generations. The evolution process is guaranteed by the genetic operators of selection and reproduction. In the classical GA, reproduction consists in the exchange of genetic material between two selected individuals. This process is known by crossover. After crossover, the generated offspring can be affected by mutations, with a low occurrence probability, that change the value of a gene (Goldberg 1989).

In nature, the genetic diversity of populations is obtained and preserved by several processes, in addition to crossover and mutation. These processes involve either gene insertion, duplication, deletion or movement (Gould et al. 1996; Russell 1998).

Holland's original work (Holland 1992) used a biologically inspired genetic operator apart from crossover and mutation: the inversion. Since then this genetic operator has been largely abandoned and no new generic genetic operators have been proposed.

In the last years some authors highlighted the importance of the study of new genetic operators. Mitchell et al. (1994) emphasize the last discoveries of molecular biology as a good source of inspiration for new mechanisms of genetic material rearrangement. Banzhaf et al. (1998) share this opinion. The authors state that mechanisms such as conjugation, transposition or transduction should be studied and implemented in evolutionary computation approaches.

Following these guidelines, in the last years, some work has been done: Furuhashi et al. (1994) introduced an application using a bacterial mechanism called transduction. This work presented a new approach for finding fuzzy rules for an obstacle avoidance problem involving a mobile robot. The authors showed that using transduction to locally improve the chromosomes, the GA would be more efficient finding the solution. Later, Harvey (1996) suggested a GA where crossover was replaced by a new genetic operator - conjugation; Smith (1996a, 1996b) used the same mechanism to solve complex satisfiability problems and proposed a different version of conjugation; Odutayo (1996) empirically studied the conjugation and crossover operators; Simões et al. (1999a) introduced a new mechanism based on the presence of genetic mobile units called transposons or jumping genes. This mechanism, known as transposition proved to be a good alternative to crossover. An extensive empirical study (Simões et al. 1999b) demonstrated that, when substituting crossover (1 point, 2 point or uniform) by transposition, if the correct set of parameters is chosen the GA can achieve better results, even with smaller populations. The choice of the right setting of parameters is the most important task when using transposition.

This paper presents the main rules that should be used to set those parameters. Once established, it is assured that transposition is a better operator than crossover, allowing the GA, with smaller populations, to achieve better results with faster convergence.

The rest of the paper is organized as follows. In section 2, we will introduce the reader to the classical GA. In section 3, we will present relevant work by several authors in biologically inspired genetic operators, besides crossover and mutation. Section 4, describes previous work with transposition and focus the functioning of transposition in biology and its adaptation to the GA. Section 5 presents the empirical results obtained either with transposition and crossover and demonstrates the importance for transposition of the parameter choice. Section 6 analyzes the factors that disturb transposition performance. Finally, we present the relevant conclusions about the work.

2 The Classical Genetic Algorithm

A GA starts with a randomly initialized population of candidate solutions and implements probabilistic and parallel exploration in the search space using the domain-independent genetic operators of selection, crossover and mutation (Goldberg 1989). A GA associates each individual

candidate in the population with a fitness which measures the quality of a solution. Selection chooses individuals probabilistically, according to their fitness. The higher the fitness, the more likely it is for an individual to be selected. Crossover and mutation produce new individuals: the first operator exchanges genetic information between two selected parents; mutation randomly changes one gene value to the generated offspring. The GA searches through an iterative process: the process of one generation involving selection, crossover and mutation is called one cycle of iteration and is repeated until convergence is reached or the number of generations achieves the established limit.

The typical GA is described in Figure 1.

1. Randomly initialize population
2. Do
   2.1. Evaluate population
   2.2. Select parents
   2.3. Crossover
   2.4. Mutation
   2.5. Substitute old population

Until (DONE)

Figure 1: The Classical Genetic Algorithm

3 Inspirations from Biology: Previous Work

Biology is the main inspiration for the development of the GA field. Several researches have taken from biology new ideas for the search of new genetic operators, later integrated in the GA. These studies, referred briefly in the introduction, will next be further explained.

Holland's pioneering work (Holland 1992) used an operator called inversion. Inversion, a reordering operator inspired by a similar operator in real genetics, was used by Holland to retain the "semantics" of a chromosome when it is altered by crossover. To apply this operator each gene is indexed to its real position, according to which the chromosome is evaluated. Inversion works by choosing two points in the string and reversing the order of the bits between them. This does not change the quality of the chromosome, since to calculate the fitness the string is ordered by the indexes. However, it does change the linkages: the idea behind inversion is to produce orderings in which beneficial schemas are more likely to survive (Mitchell 1996).

Since Holland's work, the research of genetic operators was oriented to the problem domain (Davidor 1989; D'Haeseleer 1993; Mathias et al. 1992; Parsons et al. 1995) with no innovating studies on general purpose genetic operators. Rather, Holland's inversion operator was largely abandoned.

Only in 1994 Furuhashi et al. (1994) introduced an application using a bacterial mechanism called transduction. Transduction is a process involving bacteriophages which carry a copy of a gene from a host cell and insert it into the chromosome of an infected cell. By transduction it is possible to spread the characteristics of a single bacterium to the rest of population. Furuhashi et al (1994) presented a new approach for finding fuzzy rules for an obstacle avoidance problem involving a mobile robot, showing that, using transduction for locally improve the chromosomes, the GA became much more efficient for finding the solution. Transduction was also used by Yoshikawa et al. (1997) and Nawa et al. (1997, 1998, 1999).

Harvey (1996) introduced a simplified GA – the Microbial GA – were the crossover was replaced by a new biologically inspired genetic operator: conjugation. This mechanism, present in bacteria consists in the unidirectional exchange of genetic material between two cell, one acting as ‘male’, the donor, the other as ‘female’, the recipient. Harvey developed a tournament-based conjugation, where the tournament was used to define the donor (winner) and the recipient (loser) cell. After defining two points, just like in crossover, the genes enclosed were injected in the recipient individual. The winner was not changed.

Smith (1996a, 1996b) used the same operator to solve complex satisfiability problems and proposed a simple conjugation operator without tournament, where the individuals were placed in a 15x15 matrix and the ones positioned in adjacent positions could conjugate their genetic material. Both authors employed conjugation with success, but made no extensive empirical study.

Odutayo (1996), using both tournament and simple conjugation operator, made an comparative study with 1-point, 2-point and uniform crossover using the five De Jong's test bed functions.

Simões et al. (1999a) presented a new biologically inspired genetic mechanism, transposition, as an alternative to crossover. In a preliminary work, using a GA as a function optimizer, with a single test function, very promising results were obtained. This work compared the GA performance first with 1-point, 2-point and uniform crossover and then with a simple form of transposition. In certain circumstances, the transposition allowed the GA to reach higher results than crossover, even with smaller populations. Later, this preliminary work was extended to a test bed containing eighteen test functions and the comparative study showed that, if the right parameters were chosen, transposition always performed better than crossover. Moreover, the authors introduced a new form of transposition, inspired in Harvey’s work, called tournament-based transposition, which also proved to be a good substitute to crossover (Simões et al. 1999b).

4 Transposition

In nature the genetic diversity of the individuals is preserved by several mechanisms that involve operations like gene insertion, duplication or movement (Russell 1998).

One of these mechanisms is called transposition, and will be described in next section.

4.1 Biological Transposition

Transposition is characterized by the presence of mobile genetic units inside the genome, moving themselves to new locations or duplicating and inserting themselves elsewhere. These mobile units are called transposons (Gould et al. 1996).

Transposons (also known as jumping genes) can be formed by one or several genes or just a control unit. The movement can take place in the same or in a different chromosome.

Transposition was first discovered by Barbara McClintock in the 50's (when the DNA structure was not yet completely understood). She proved that certain phenomena present in living beings exposed to UV radiation could not be the result of the normal recombination and mutation processes. She found that in corn certain genetic elements occasionally move producing kernels with unusual colors that could not have resulted from crossover or mutation. Transposons were for a long time considered as some sort of abnormality, but in 1983 when she was awarded the Nobel Prize, many such transposons had been discovered and their possible role in evolution was beginning to be recognized. For instance, the genetic alterations caused by transposons are responsible for the growth of cancers in human or the resistance to antibiotics in bacteria (Gould et al. 1996; Russell 1998).

In order for a transposable element to transpose as a discrete entity it is necessary for its ends to be recognized. So, transposons within a chromosome are flanked by identical or inverse repeated sequences, some of which are actually part of the transposon. See Figure 2.

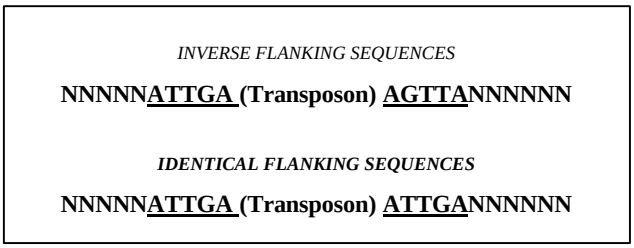

Figure 2: Inverse and Equal Flanking Sequences

When the transposon moves to another zone of the genome one of the flanking sequences goes with it.

The insertion point can be chosen at random, but there are transposons that show a regional preference when inserting into the same gene. Other method can be a correspondence in the new position with the flanking sequence. The point into which the transposon is inserted requires no homology with the point where the transposon was excised. This is in marked contrast to classical recombination, where relatively long sequences of DNA must share homology to permit a recombination event to occur (same cut point(s)). Subsequently, transposition is sometimes referred to as illegitimate recombination.

4.2 Computational Transposition

The first form of computational transposition proposed by Simões et al. (1999a) was directly inspired in biology. After the selection of two parents for mating, the transposon is formed in one of them. The insertion point is found in the second parent. According to this point, the same amount of genetic material is exchanged between the two chromosomes. The transposon is recognized by the presence of equal or inverse flanking sequences with a fixed length. The insertion point is searched in the second chromosome and is chosen when a sequence of bits equal or inverse to the flanking sequence is found. The insertion point will be the first gene after that sequence. After that, the movement of the transposon occurs. Since it was used fixed size chromosomes, the same amount of genetic material is exchanged between the two selected parents. The detailed functioning of transposition is described in Simões et al. (1999a). In this paper, this mechanism will be referred as simple transposition.

The first observations of the results immediately showed that, in spite of the good results using simple transposition, the population average became very unstable. In order to minimize this effect a new form of transposition was implemented: tournament-based transposition.

The two selected parents become competitors in a tournament. The transposon will be searched in the winner chromosome and the insertion point will be found in the loser parent. Only this individual will be altered by inserting the transposon, which replaces the same number of bits after the insertion point. Figure 3 shows these two mechanisms.

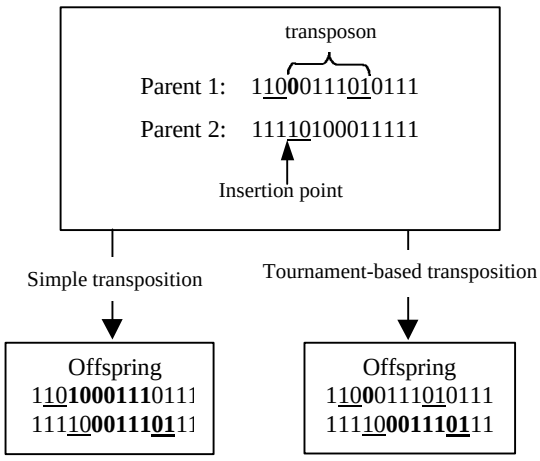

Figure 3: Simple and Tournament-based Transposition

In the next section we will briefly refer to the results obtained using the GA with transposition and crossover.

5 Empirical Results

The empirical study consisted in comparing the GA performance, as a function optimizer, first using transposition and then using 1-point, 2-point and uniform crossover. To measure its production, when using different genetic operators, was applied a test suit containing eighteen test functions. These functions, used by several authors to evaluate evolutionary approaches (De Jong 1975; Fogel 1995; Foster 1995; Michalewicz 1994; Whitley et al. 1995; Koon et al. 1995), were selected to include a large set of characteristics.

Since the GA was used as a function optimizer, we chose roulette wheel with elitism as the selection method, in order to keep track of the best solution found (De Jong 1993).

The GA was first implemented with crossover (1-point. 2-poit and uniform) and then with transposition (simple and tournament-based). The population size varied between 50, 100 and 200 individuals, either for transposition and crossover. The elite size was 20% of the complete population. The mutation and crossover/transposition rate used was 0.01 and 0.7, respectively. Ten runs of each experiment involving 1-point, 2-point and uniform crossover were executed. All the tests were run over 500, 1000 or 2000 generations - depending on the test function.

The main conclusion was that the GA with transposition, choosing the right size for the flanking sequences, even with smaller populations (50 individuals), can achieve higher results than crossover (Simões et al. 1999b).

Figure 4, taken from (Simões et al. 1999b), is a typical example that illustrates these conclusions. With a population of 50 individuals, the GA with transposition obtained much better results than crossover with 50, 100 and 200 individuals.

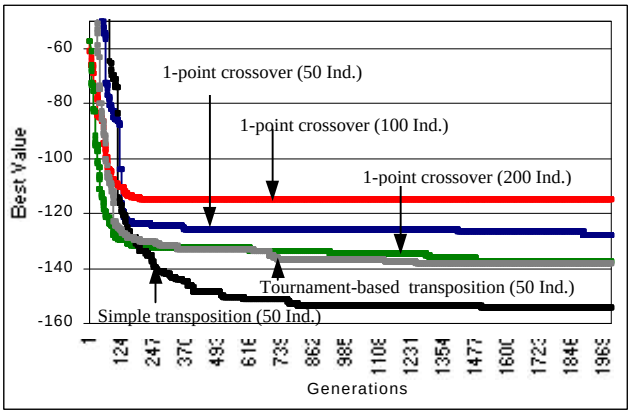

Figure 4: Transposition (seq. = 4; pop = 50) versus 1-point Crossover (Pop. =50, 100, 200)

Detailed information about this study and its results can be found in Simões et al. (1999b) and Simões (1999).

In next section we will try to understand which factors influence transposition and how we can control such parameters in order to enhance transposition performance.

6 Transposition Performance

Transposition performance depends essentially on two factors: the flanking sequences length and the population size.

The used flanking sequences length was varied in an interval from 1 to a maximum value, depending on the chromosome size. This maximum value was 8, 10, 15 and 20 for chromosomes of 'small', 'medium' and 'large' size, respectively. If we intend to use transposition as a good alternative to crossover, we have to find a way to avoid such variation and only then, according to the results, choose the sequence size that lead to the best solution. Therefore, the selection of the right length for the flanking sequences is the most important task when using transposition in the GA.

In the next sections we will present the role of these two parameters in transposition performance and we will introduce the heuristics that will allow the selection of the appropriate sequences size.

To illustrate the importance of the sequences size in the obtained results, we chose one of the used test functions - the *Rastrigin's bidimensional function* - which is representative of the global results. In section 6.2 we will present a synthesis of the solutions achieved in the global test suit in order to introduce the heuristics.

Rastrigin's bidimensional function is a continuous, convex, multimodal (50 local minima), with one global minimum, quadratic and low-dimensional (two variables) function, defined by:

$$f(x_1, x_2) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2)$$

Some of the referred characteristics can be seen in the graphical representation of the function (Figure 5).

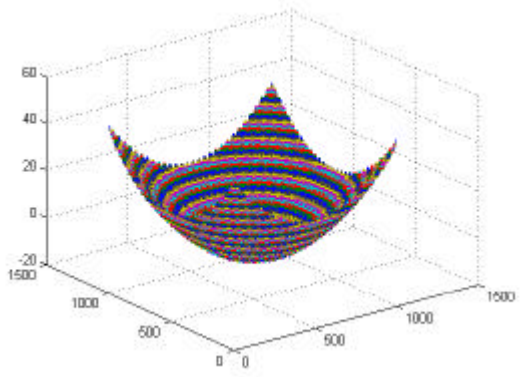

Figure 5: The Rastrigin's Bidimensional Test Function

The search domain of the variables was restricted to x_1, $x_2 \in$ [-5.0, 5.0], using a 0.0001 axis precision. Each variable was codified with seventeen bits being the chromosome size equal to 34.

6.1 The Role of the Flanking Sequences Length

The empirical study didn't enable us to conclude a precise generic rule to predict the GA behavior resulting from different flanking sequences. Nevertheless, our study revealed some regularities that culminated in heuristic guidance rules. As an example, for simple transposition, smaller sequences (3, 4, 5) allow the GA to achieve better results than larger ones (8, 9, 10, ...). For tournament-based transposition, the higher quality solutions were obtained with sequences of medium size between 6 and 7. In Figure 6, we can see the effects of the flanking sequences variation (between 1 and 10) in simple transposition.

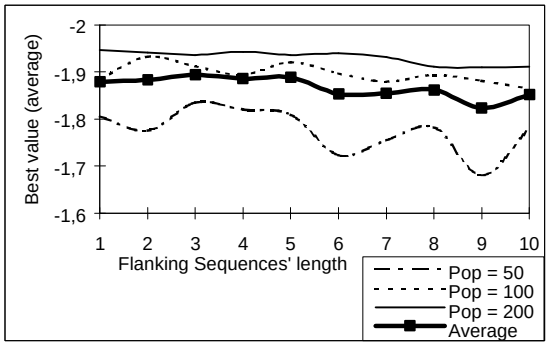

Figure 6: Effects of Changing the Flanking Sequences Length in Simple Transposition.

Trying to understand the reasons for those results, we analyzed the changing of the transposon length along with the variation of the flanking sequences size. Table 1 resumes the results.

Sequences Length	Transposon Length (average)	Transposon length = 34 (%)
1	2,79	0,00%
2	4,53	0,33%
3	6,70	1,46%
4	11,62	6,83%
5	15,26	14,24%
6	17,87	20,33%
7	24,26	37,26%
8	27,65	44,21%
9	25,80	46,87%
10	28,45	52,98%

Table 1: Variation of the Transposon Length in Simple Transposition

As the table shows, by increasing the flanking sequences size, we obtain a higher transposon length average. Moreover, the number of times that the transposon size is equal to the chromosome size (i.e. no transposition occurs), also increases. Therefore, the worst results achieved with larger sequences are due to the reduction of the population diversity, as a consequence of less exchange of genetic material by transposition.

The choice of the flanking sequences size is, then, a highly relevant task, since it conditions the final solution.

In the next section we will explain how this task can be simplified.

6.2 Choosing the Flanking Sequences Length

As mentioned, the analysis of the achieved results in all the test functions couldn't culminate in any general rule, capable to compute the appropriate sequences length. Nevertheless, the direct observation of the results, allowed to conclude about the way to confine the interval of variation for the flanking sequences.

Simple and tournament-based transposition behave differently. In each case, the best flanking sequences size had different characteristics. Each case will next be separately described.

6.2.1 Simple Transposition

The chromosome size of the different test functions varies between 19 and 280 bits. The analysis of the results clearly shows that, if we divide the chromosomes lengths in three categories - small, medium and large dimension - it is possible to obtain partial rules for compute the suitable sequences length.

Those categories are determined by the following intervals:

- Small dimension: from 19 to 38;
- Medium dimension: from 39 to 74;
- Large dimension: from 75 to max (280 in our case).

According to these three categories the variation interval for the flanking sequences size can be confined using one of the following heuristics:

- Sequence size = (10% * chromosome size) ± 1, if chromosome size ∈ [19, 38]
- Sequence size = (5% * chromosome size) ± 1, if chromosome size ∈ [39, 74]
- Sequence size = (0.3% * chromosome size) ± 1, if chromosome size ∈ [75, max]

These heuristics were inferred from the empirical results and offer an interval of three values in which best solution for transposition can be found. In few exceptional cases, the interval computed by the respective rule doesn't include the size which assures the best result. Nevertheless, the heuristic always estimate an interval that includes a size allowing a result close to the optimal. For instance, for the

bidimensional test function II (N=3) (Koon et al. 1996) the respective heuristic estimates the interval [3,5]. The best result was obtained with sequences size equal to 6. However, sequences of 3, 4 and 5 bits (included in the computed interval) allowed the GA to reach results very close to the best one.

Table 2 resumes all the intervals obtained by the heuristics and the appropriate flanking sequences size which lead to the best score. In the last column we indicate the sequence size responsible for results near the optimal.

Test Function	Chrom. length (CL)	Transposition				
		(10% CL) ± 1			*Best*	***» Best***
N-Dim. (N = 1)	19	1	2	3	2	1, 3
Uni Dim.	19	1	2	3	3	4, 1, 2
De Jong F2	25	2	3	4	3	5, 2
De Jong F1	30	2	3	4	4	6, 3
Michalewicz	33	2	3	4	4	2, 5
De Jong F5	34	2	3	4	all	all
Bohachevsky I	34	2	3	4	3	4, 9
Bohachevsky II	34	2	3	4	3	4, 2
6-Hump CamelBack	34	2	3	4	4	3, 5
Shubert	34	2	3	4	2	4, 3
Bi-Dim. I	34	2	3	4	2	3, 4
Bi-Dim. II (N = 1)	34	2	3	4	3	4, 7
Bi-Dim. II (N = 2)	34	2	3	4	2	7, 3
Rastrigin's 2-Dim	34	2	3	4	3	5, 4, 2
N-Dim. (N = 2)	38	3	4	5	3	4, 5
Bi-Dim. II (N = 3)	38	3	4	5	***10***	**3, 4, 5**
Bi-Dim. II (N = 4)	38	3	4	5	***6***	***2, 4***
		(5% CL) ± 1			***Best***	***» Best***
De Jong F3	50	2	3	4	2	3, 4, 5
N- Dim. (N = 3)	57	2	3	4	4	3, 6
N - Dim. (N= 4)	74	3	4	5	4	6, 3, 5
		(0.3% CL) ± 1			***Best***	***» Best***
Schwefel (N=10)	200	5	6	7	***4***	5, 3, 6
Griewangk (N=10)	210	5	6	7	***8***	7, 6, 4
De Jong F4	240	6	7	8	***2***	7, 8, 9
Rastrigin (N=20)	280	7	8	9	9	11, 3
Sequences length average					***3.9***	

Table 2: Sequences Size Computed by the Heuristics; Sequences Length which Allowed to Achieve the Best Results.

The table clearly shows that the achieved heuristics compute the intervals that enable the GA to reach the best scores. In fact, only one case is observed where the sequence length which lead to the best result is very distant from the given interval (bidimensional test function II, N=3).

6.2.2 Tournament-based Transposition

For tournament-based transposition, the analysis of the results revealed that, separating the chromosomes lengths in two categories - small and large dimension - it is possible to obtain partial heuristics to compute the best sequence length to a given problem.

Those categories are determined by the following intervals:

- Small dimension: from 19 to 38;
- Large dimension: from 39 to max;

In this case, best results were obtained with larger flanking sequences than the ones used in simple transposition. The average size for flanking sequences which allows best results is 7.13 and in simple transposition is 3. 9.

According to those two categories the interval for the flanking sequences size can be limited using one of the following heuristics:

- Sequence size = (18% * chromosome size) ±1, if chromosome size ∈ [19, 38]
- Sequence size = (5% * chromosome size) ± 1, if chromosome size ∈ [39, max]

The conclusions are similar to simple transposition. In general, the correct rule offers an interval of three values where the best (or close to the best) solution can be found. Nevertheless, in some cases, the interval computed by the respective heuristic doesn't include the size which leads to the best score. Just like before, the estimated interval for flanking sequence includes a size which allows results close to the optimal. As an example, for the first function listed in the table heuristics estimates the interval [2, 4], being the best score obtained with sequence size equal to 6. However, sequences of 3 and 4 bits (included in the computed interval) allowed the GA to get results very close to the best.

Table 3 resumes all the intervals obtained by the heuristics and the proper flanking sequences size.

6.3 The Role of the Population Size

The effects of the population size in transposition performance have less impact than the flanking sequences size. As Figure 6 reveals, larger populations allow more stability when changing the flanking sequences size. When using populations of 200 individuals, in most cases, the selection of the flanking sequences size became less problematic because the GA easily reaches the best score.

Thus, there is a trade-off between choosing a larger population, and paying the inherent computational cost, or to use a smaller population and apply the respective heuristic to get the appropriated flanking sequences length.

Test Function	Chrom. length (CL)	Tournament-based Transposition				
		(18% CL) ± 1			*Best*	***» Best***
N-Dim. (N = 1)	19	2	3	4	***6***	4, 3
Uni Dim.	19	2	3	4	***5***	3, 4
De Jong F2	25	4	5	6	***8***	6, 4, 5
De Jong F1	30	4	5	6	6	4, 5
Michalewicz	33	5	6	7	***8***	3, 4, 5
De Jong F5	34	5	6	7	all	all
Bohachevsky I	34	5	6	7	6	9, 7
Bohachevsky II	34	5	6	7	***9***	6, 7, 5
6-Hump CamelBack	34	5	6	7	***8***	6, 2, 7
Shubert	34	5	6	7	***4***	6, 7
Bi-Dim. I	34	5	6	7	***10***	7, 5, 6
Bi-Dim II (N=1)	34	5	6	7	6	9, 7
Bi-Dim II (N=2)	34	5	6	7	5	3, 8
Rastrigin's 2-Dim.	34	5	6	7	5	2, 6, 4
N-Dim. (N = 2)	38	6	7	8	6	4, 7
Bi-Dim II (N=3)	38	6	7	8	6	9, 8
Bi-Dim II (N=4)	38	6	7	8	6	***8, 5***
		(5% CL) ± 1			***Best***	***» Best***
De Jong F3	50	2	3	4	3	4, 8
N- Dim. (N = 3)	57	2	3	4	***6***	4, 5
N - Dim. (N = 4)	74	3	4	5	3	4, 5, 6
Schwefel (N=10)	200	9	10	11	10	11, 9
Griewangk (N=10)	210	10	11	12	10	12
De Jong F4	240	11	12	13	***20***	13, 12
Rastrigin (N=20)	280	13	14	15	***11***	12, 13, 14
Sequences length average					***7.13***	

Table 3: Sequences Size Computed by the Heuristics; Sequences Length which Allowed to Achieve the Best Results

Once more, the heuristics demonstrate to be dependable. The computed intervals, with only one exception (bidimensional I test function), offer values matching the ones enhancing the GA performance.

7 Conclusions

In this paper we tried to understand the factors which affect the performance of the transposition mechanism.

Transposition is a new biologically inspired genetic operator which demonstrated to be a powerful alternative to the traditional crossover. The GA performance when using transposition and crossover was previously analyzed in recent works. This paper presented a study on the factors which influence the GA performance when applied transposition, in order to enhance its efficiency. The two analyzed parameters were the flanking sequences length and the populations size.

We concluded that larger populations allow the GA achieve the best scores, independently of the flanking sequences size. Nevertheless, this option carries computational costs. To avoid this limitation, we introduced some heuristics that can be used to confine the interval for the flanking sequences size. Applying the respective heuristic, the calculated interval allows transposition to get the best results, even with smaller populations.

Acknowledgments

This work was partially financed by the Portuguese Ministry of Science and Technology under Program Praxis XXI.

Bibliography

Banzhaf, W., Nordin P., Keller, R. E. and Francone, F. D. (1998), "Genetic Programming - An Introduction - On the Automatic Evolution of Computer Programs and its Applications", CA: Morgan Kaufmann.

Davidor, Y. (1989), "Analogous Crossover", In J. D. Schaffer (ed.), Proceedings of the Third International Conference on Genetic Algorithms, pp. 98-103, CA: Morgan Kaufmann.

De Jong, K. A. (1975), "Analysis of the Behavior of a Class of Genetic Adaptive Systems", Ph.D. Dissertation, Department of Computer and Communication Science, University of Michigan.

De Jong, K. A. (1993), "Genetic Algorithms Are NOT Function Optimizers", In L. Darrell Whitley, (ed.), Foundations of Genetic Algorithms 2, pp. 5-17, CA: Morgan Kaufmann.

D'Haeseleer, P. (1993), "Context Preserving Crossover in Genetic Programming", Proceedings of the 1994 IEEE World Congress on Computational Intelligence, Vol. 1, pp. 256-261, IEEE Press.

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

Foster, J. A. (1995), "On Test Suits", Lecture Notes on Genetic Algorithms.

Furuhashi, T., Nakaoka, K. and Uchikawa Y. (1994), "A New Approach to Genetic Based Machine Learning and an Efficient Finding of Fuzzy Rule", In Furuhashi, T. (ed.), Proceedings of the 1994 IEEE/Nagoya-University World Wisepersons Workshop (WWW'94), Lecture Notes in Artificial Intelligence, Vol. 1011, pp. 173-189, Springer Verlag.

Goldberg, D. E. (1989), "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley Publishing Company, Inc.

Gould, J. L. and Keeton, W. T. (1996), "Biological Science", W. W. Norton & Company.

Harvey, I. (1996), "The Microbial Genetic Algorithm", Submitted as a Letter to Evolutionary Computation.

Holland, J. H. (1992), "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence", 1st MIT Press Edition, MIT Press.

Koon, G. H. and Sebald, A. V. (1996), "Some Interesting Test Functions for Evaluating Evolutionary Programming Strategies", In McDonnell, J. R., Reynolds, R. G. and Fogel, D. B. (eds.), Proceedings of the Fourth Annual Conference on Evolutionary Programming, pp. 479-499, San Diego, CA, MIT Press.

Mathias, K. and Whitley D. (1992), "Genetic Operators, the Fitness Landscape and the Traveling Salesman Problem", In Männer, R. and Manderick, B., (eds.), Parallel Problem Solving from Nature (PPSN 2), pp. 219-228, North Holland-Elsevier.

Michalewicz, Z. (1994), "Genetic Algorithms + Data Structures = Evolution Programs", Springer-Verlag.

Mitchell, M. and Forrest, S. (1994), "Genetic Algorithms and Artificial Life", Artificial Life 1 (3), pp. 267-289.

Mitchell, M. (1996), "An Introduction to Genetic Algorithms", MIT Press.

Nawa, N., Hashiyama, T., Furuhashi, T. and Uchikawa, Y. (1997), "A Study on Fuzzy Rules Using Pseudo-Bacterial Genetic Algorithm with Adaptive Operator", Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC'97), pp. 589-593, IEEE Press.

Nawa, N. and Furuhashi, T. (1998), "Bacterial Evolutionary for Fuzzy System Design", Proceedings of the 1998 IEEE International Conference on Systems, Man and Cybernetics. IEEE Press.

Nawa, N., Furuhashi, T., Hashiyama, T. and Uchikawa, Y. (1999), "A Study of the Discovery of Relevant Fuzzy Rules Using Pseudo-Bacterial Genetic Algorithm", IEEE Transactions on Industrial Electronics.

Odutayo, M. A. (1996), "An Empirical Study of Crossover and Conjugation using the De Jong's Test Bed of Optimization Functions", M. Eng. Degree in Software Engineering Dissertation. (unpublished work), City University.

Parsons, R., Forrest, S. and Burks, C. (1995), "Genetic Algorithms, Operators, and DNA Fragment Assembly", Machine Learning, pp. 1-24.

Russell, P. J. (1998), "Genetics", 5th edition, Addison-Wesley.

Simões, A. and Costa, E. (1999a), "Transposition: A Biologically Inspired Mechanism to Use with Genetic Algorithms", In Dobnikar, A., Steele, N., Pearson D. and Albrecht, R. (eds.), Proceedings of the Fourth International Conference on Neural Networks and Genetic Algorithms (ICANNGA'99), Slovenia, Springer-Verlag.

Simões, A. and Costa, E. (1999b), "Transposition versus Crossover: An Empirical Study". In Banzhaf, W., Daida, J., Eiben, A. E., Garzon, M. H., Honavar, V., Jakiela, M., and Smith, R. E. (eds.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99), Orlando, Florida USA, CA: Morgan Kaufmann.

Simões A. (1999), "Transposition: A Biologically Inspired Mechanism and its Integration in the Classical Genetic Algorithm", Unpublished M.Sc. Dissertation Thesis (in Portuguese), University of Coimbra.

Smith, P. (1996a), "Finding Hard Satisfiability Problems Using Bacterial Conjugation", Presented at the AISB96 Workshop on Evolutionary Computing, University of Sussex.

Smith, P. (1996b), "Conjugation - A Bacterially Inspired Form of Genetic Recombination", Late Breaking Papers of the First International Conference on Genetic Programming, Stanford University, California.

Whitley, D., Mathias, K., Rana, S. and Dzubera, J. (1995), "Building Better Test Functions", In L. J. Eshelman (ed.), Proceedings of the Sixth International Conference on Genetic Algorithms, pp. 239-246, CA: Morgan Kaufmann.

Yoshikawa, T., Furuhashi T. and Uchikawa, Y. (1997), "Knowledge Acquisition of Fuzzy Control Rules for Mobile Robots Using DNA Coding Method and Pseudo-Bacterial Genetic Algorithm", Proceedings of the First Asia-Pacific Conference on Simulated Evolution and Learning, Lecture Notes in Artificial Intelligence, Vol. 1285, pp. 126-135. Springer-Verlag.