Using Genetic Algorithms with Asexual Transposition

Anabela Simões[1,2], Ernesto Costa[2]
[1] Centre for Informatics and Systems of the University of Coimbra, , Polo II, 3030 Coimbra, Portugal
[2] Leiria College of Education, Leiria Polytechnic, Rua Dr. João Soares, 2411-901 Leiria, Portugal
E-mail: {abs, ernesto}@dei.uc.pt

Abstract

Traditional Genetic Algorithms (GA) use crossover and mutation as the main genetic operators to achieve population diversity. Previous work using a biologically inspired genetic operator called transposition, allowed the GA to reach better solutions by replacing the traditional crossover operators. In this paper we extend that work to the case of asexual reproduction. The GA efficiency was compared when using asexual transposition and the traditional crossover operators. The results obtained show that asexual transposition still allowed the modified GA to achieve higher performances.

1 Introduction

Genetic diversity is essential for the evolutionary process. When using genetic algorithms, a population evolves through the application of two main genetic operators: mutation and crossover. These operators allow changes in the individuals, creating evolutionary advantages in some of them. The fittest individuals are more likely to be selected by the used selection method allowing the evolution of the population to the best solution [5].

In nature genetic diversity is caused and maintained by several mechanisms besides crossover and mutation. Some of those mechanisms are: inversion, transduction, transformation, conjugation, transposition and translocation [6].

Some researchers highlighted the importance of these latest discoveries of molecular biology. Mitchell et al. and Banzhaf et al. stress that it would be important to analyze if some of the mechanisms of rearranging genetic material present in the biological systems, when implemented and used with a GA, improve its performance [10], [11], [1].

Several authors have already used some biologically inspired mechanisms besides crossover and mutation in genetic approaches. For instance, inversion [8], conjugation [7], [21],[22], transduction [4], [12], [13], [14], translocation [15], [23] and transposition [17], [18], [19], [20] were already used as the main genetic operators in the GA.

In this paper we will introduce an extension to our previous work with the transposition mechanism. This new proposal will be referred to as asexual transposition. Transposition consists in the presence of genetic mobile units called transposons, capable of relocating themselves, or transposing, onto the chromosome and subsequently jumping into new zones of the same or a different chromosome. In asexual transposition the movement of the transposon will occur in the same chromosome.

We will compare the performance of the GA in the domain of function optimization using a standard test suite with either crossover or asexual transposition.

This paper is organized in the following manner. First, in section 2, we summarize our previous work related to the transposition mechanism. In section 3, we describe how transposition works in nature and how the previous versions were implemented. In section 4, we present the computational form of asexual transposition. Section 5 describes the characteristics of the experimental environment. In section 6, we make an exhaustive comparison of the results obtained with asexual transposition, 1-point, 2-point and uniform crossover. Finally we present the relevant conclusions of the work.

2 Previous Related Work

Simões et al. presented a new biologically inspired genetic mechanism, called transposition, as an alternative to crossover [17]. In a preliminary work, using a GA as a function optimizer, with a single test function, very promising results were obtained. Such work compared the GA performance when using 1-point, 2-point, uniform crossover or a simple form of transposition. Transposition allowed the GA to reach better results than crossover, even with smaller populations. Later, this preliminary work was enlarged to a test bed containing eighteen test functions and an extensive 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 [18]. For a detailed description of this work see [20].
Both mechanisms used sexual reproduction, i.e., two individuals were selected for mating and the transposition mechanism occurs between these individuals. In simple transposition the exchange of the genetic mobile unit is made in a random manner between the two strings. In tournament-based transposition the best individual will give the genetic material to the worst one. In the next section we will explain how these two mechanisms were implemented.

3 Transposition

In this section we will explain how transposition works in nature and how it was implemented in the proposals based in sexual reproduction.

3.1 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 [6], [16].
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 chromosome or to a different one.
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 certain genetic elements in corn 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 cancer in humans and for the resistance to antibiotics in bacteria [6], [16]. In order to a transposable element to transpose as a discrete entity it is necessary for its ends to be recognized. Therefore, transposons within a chromosome are flanked by identical or inverse repeated sequences, some of which are actually part of the transposon. See Figure 1 bellow.
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.

INVERSE FLANKING SEQUENCES
NNNNN<u>ATTGA</u> (Transposon) <u>AGTTA</u>NNNNNN

IDENTICAL FLANKING SEQUENCES
NNNNN<u>ATTGA</u> (Transposon) <u>ATTGA</u>NNNNNN

Fig. 1. Inverse and Equal Flanking Sequences

The point into which the transposon is inserted requires no homology with the point where the transposon was excised. This is in evident contrast to classical recombination, where relatively long sequences of DNA must share homology to permit a recombination event to occur (same cut point(s)). Consequently, transposition is sometimes referred to as illegitimate recombination.

3.2 Simple Transposition

The first form of computational transposition proposed by Simões et al was directly inspired in biology [17]. 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. Figure 2 shows how simple transposition works. The detailed functioning of transposition is described in [17]

3.3 Tournament-based Transposition

In order to come closer to the biological mechanism, the authors proposed a new form of transposition: *tournament-based transposition*.
The two selected parents become competitors in a tournament of size two. The transposon will be

searched in the winner chromosome and the insertion point will be located 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 this mechanism

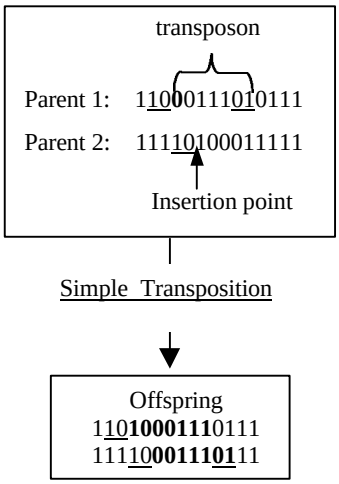

Fig. 2. Simple Transposition

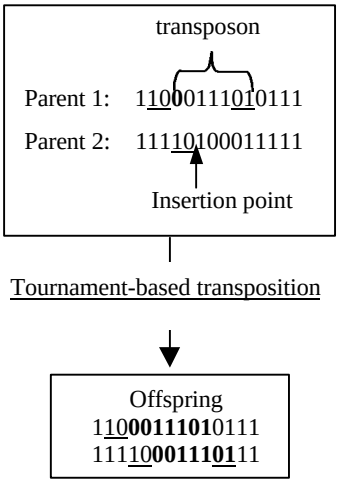

Fig. 3. Tournament-based Transposition

4 Asexual Transposition

As stated before, in nature the transposition mechanism can occur also in the same chromosome. Previous work, described above, explored two forms of transposition always involving two different chromosomes (sexual reproduction). Our work will focus in a new proposal based in asexual reproduction. The basic functioning of the mechanism will be maintained: the way of building the transposon and finding the insertion point is kept. The main difference will be that all the process will operate in the same individual. After selecting one individual for reproduction, the asexual transposition will be applied. Figure 4 synthesizes the complete process of asexual transposition.

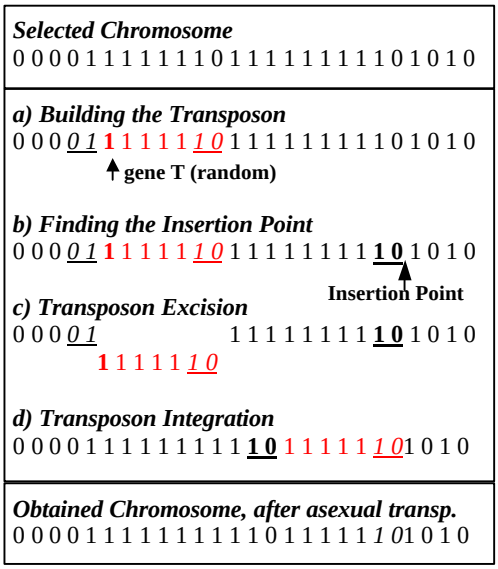

Fig. 4. Asexual Transposition

The flanking sequence length (**FSL**) is previously determined and maintained in all experiments. After selecting the first parent, the beginning of the transposon will be randomly selected (gene **T**). According to the flanking sequence length, the **FSL** bits before the gene **T** make the first flanking sequence. The search of the second flanking sequence begins after gene **T** and stops when an equal or inverse sequence is found. The transposon is constituted by the genes enclosed by gene **T** and the last gene of the second flanking sequence.

The insertion point is searched in the same chromosome and this process starts in the first bit after the second flanking sequence. The insertion point is defined when a equal or inverse sequence of bits is found in the chromosome. Notice that the chromosome is viewed as having a circular form. Therefore, after reaching the end of the chromosome the search continues in the first bit of the chromosome. When the insertion point is found, the transposon excises from its original position and will integrate in the insertion point. This process is repeated a number of times equal to the new individuals we want to find (depending on the population size and the elite size).

5 The Experimental Environment

The performance of asexual transposition was studied using a test suite containing seven functions (see appendix), characterized as continuous/ discontinuous, unimodal/multimodal, high/low/scalable, dimensional, stochastic/deterministic, quadratic/non-quadratic and convex/non-convex. These functions are a well known benchmark for genetic approaches [2], [9], [24] and were selected in order to cover such 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 [3].

The GA was first implemented with crossover (1-point. 2-point and uniform) and then with asexual transposition. The population size varied between 50, 100 and 200 individuals, either for asexual transposition or crossover. The elite size was 20% of the complete population. The mutation and crossover/asexual 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.

Asexual transposition was tested with flanking sequences from 1 to 10, 15 or 20, depending on the chromosome length. All the tests were run over 500 or 1000 generations, depending on the test function.

We used the De Jong's off-line measure to compare GA efficiency when applied crossover or transposition [2]. This measure is defined by:

$$X^*_e(g) = \frac{1}{T} * \sum_{t=1}^{T} f_e^*(t)$$

Were $f_e^* = best\ \{f_e(1), f_e(2), ..., f_e(n)\}$ and **T** is the number of runs. This means that off-line measure is the average of the best individuals in each generation. Due to the total of ten trials, the average of the tens runs was evaluated.

6 The Results

The results obtained with the seven studied functions were very similar. In all the cases asexual transposition allowed the GA to achieve better solutions. To illustrate these results we chose Schwefel's test function, which is representative of all the test suite.

We will show the comparative analysis of the results obtained with asexual transposition and one point crossover, two point and uniform crossover. The solutions obtained when using asexual transposition refer to populations size of 50 individuals, since that, with 100 and 200 strings, the results were always much better.

6.1 The Role of the Flanking Sequence Length

The GA performance using the asexual transposition is dependent on the flanking sequence size. Only certain sequence lengths allow good results. Table 1 shows the sequence length which allowed the GA to find the best solutions. The heuristics given in the last column show that the flanking sequence size is directly dependent on the chromosome length used to codify each test function.

Table 1. Relation Between the Best Flanking Sequence Length and the Chromosome Size

Test Func.	*Chrom. Length (CL)*	*Best Seq Length*	*Heuristic*
F1	24	2, 3	10 % CL + 1
F4	33	3, 4	10 % CL + 1
F2	50	5, 6	10 % CL + 1
F7	200	10, 11	5% CL + 1
F5	210	11, 12	5% CL + 1
F3	240	12, 13	5% CL + 1
F6	280	14, 15	5% CL + 1

When larger strings are used to codify the problem (functions F3, F5, F6 and F7) the sequence lengths that allowed the GA to reach the best solutions are 5% of the chromosome length plus one. In the remainder cases, when smaller binary strings are employed, the best flanking sequences lengths are 10% of the chromosome length plus one.

6.2 Comparing Asexual Transposition with One-Point Crossover

In this section we will show the results obtained with the GA using asexual transposition (using the appropriate flanking sequence length) and 1-point crossover.

Figure 5 shows that asexual transposition with only 50 individuals allowed the GA to reach higher solutions than one point crossover with 50, 100 or 200 individuals. Using larger populations (100 and 200 strings), the obtained results were even better.

6.3 Comparing Asexual Transposition with Two-Point Crossover

The results concerning to the GA using two point crossover are very similar to the previous case. In fact, asexual transposition with only 50 individual outperformed the solutions reached by the GA using 2-point crossover with 50, 100 and 200 individuals.

Asexual transposition used with larger populations lead the GA to better solutions and faster convergence. Figure 6 shows the obtained results.

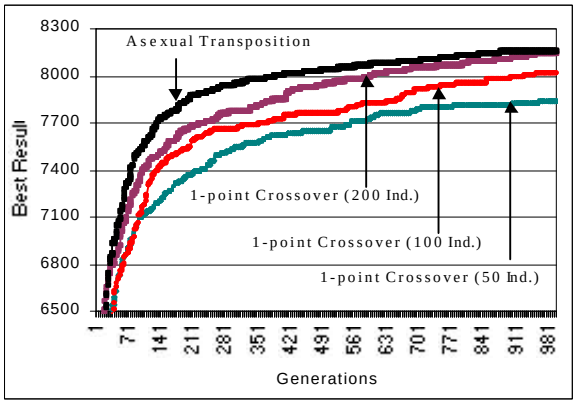

Fig. 5. Comparing Asexual Transposition (50 Individuals) with 1-point Crossover (50, 100, 200 Individuals)

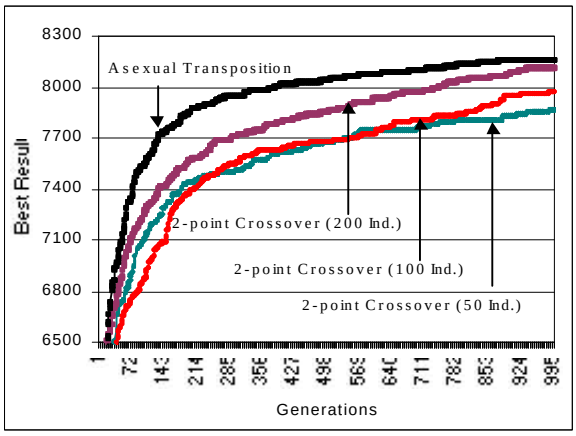

Fig. 6. Comparing Asexual Transposition (50 Individuals) with 2-point Crossover (50, 100, 200 Individuals)

6.4 Comparing Asexual Transposition with Uniform Crossover

Just like before, asexual transposition lead the GA to better solutions than uniform crossover. With only 50 individual the results obtained by the proposed mechanism outperformed the results achieved by uniform crossover with 50, 100 and 200 binary strings in the population.

Figure 7 illustrates the achieved results.

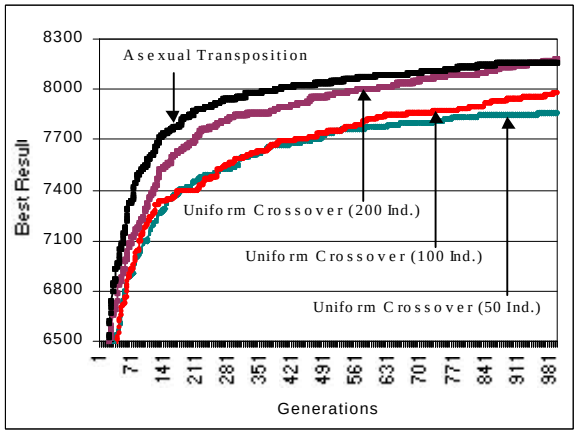

Fig. 7. Comparing Asexual Transposition (50 Individuals) with Uniform Crossover (50, 100, 200 Individuals)

7 Conclusions

In this paper we proposed a new way for using the transposition mechanism involving asexual reproduction. The GA was executed as function optimizer and its efficiency was compared when using the classical crossover operators and when applying the asexual transposition as the main recombination mechanism. For both cases we compared the GA performance with a test suite containing seven test functions.

The process employed to evaluate the GA performance was off-line measure. Some parameters, such as the population size and the flanking sequences length were changed.

Comparing the results with crossover we realized that, just like in the sexual forms of transposition studied before, asexual transposition is always better than crossover.

Furthermore, even with smaller populations the GA using asexual transposition can obtain much better results than crossover with larger populations.

8 Acknowledgements

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

9 Appendix

F1: De Jong's Test Function F2 (Rosenbrock's Saddle)

$$F_1(x_1, x_2) = 100 * \left(x_1^2 - x_2^2\right)^2 + (1 - x_1)^2$$

F2: De Jong's Test Function F3 (Step Function)

$$F_2(x) = \sum_{i=1}^{5} integer(x_i)$$

F3: De Jong's Test Function F4 (Gaussian Quartic)

$$F_3(x) = \sum_{i=1}^{30} i x_i^4 + Gauss(0,1)$$

F4: Michalewicz's Function

$$F_4(x1, x2) = 21.5 + x1.sin(4Px1) + x2.sin(20Px2)$$

F5: Griewangk's Function

$$F_5(x) = 1 + \sum_{i=1}^{n} \left[\frac{x_i^2}{4000}\right] - \prod_{i=1}^{n} \left[cos\left(\frac{x_i}{\sqrt{i}}\right)\right]$$

with $n = 10$

F6: Rastrigin's Function

$$F_6(x) = n * A + \sum_{i=1}^{n} \left[x_i^2 - A * cos(2Px_i)\right]$$

with $n = 20, A = 10$

F7: Schwefel's (Sine Root) Function

$$F_7(x) = V * n + \sum_{i=1}^{n} \left[-x_i * sin\left(\sqrt{|xi|}\right)\right]$$

with $n = 10, V = 418{,}9829$

10 References

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

[2] De Jong, K. A.: Analysis of the Behavior of a Class of Genetic Adaptive Systems. Ph.D. Dissertation, Department of Computer and Communication Science, University of Michigan 1975.

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

[4] Furuhashi, T, Nakaoka, K. and Uchikawa, Y. : A New Approach to Genetic Based Machine Learning and an Efficient Finding of Fuzzy Rule. T. Furuhashi (ed.), Proceedings of the 1994 IEEE/Nagoya-University World Wisepersons Workshop. LNAI, vol. 1011, 173-189. Springer-Verlag 1994.

[5] Goldberg, D. E.: Genetic Algorithms in search, optimization and machine learning. Addison-Wesley Publishing Company, Inc 1989.

[6] Gould, J. L., Keeton, W. T.: Biological Science. W. W. Norton & Company 1996.

[7] Harvey, I.: The Microbial Genetic Algorithm. Submitted as a Letter to Evolutionary Computation. MIT Press 1996.

[8] Holland, J. H.: Adaptation in Natural and Artificial Systems: an introductory analysis with applications to biology, control and artificial intelligence. MIT Press 1992.

[9] Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd Edition Springer-Verlag 1999.

[10] Mitchell, M.: An Introduction to Genetic Algorithms, MIT Press 1996.

[11] Mitchell, M., Forrest, S.: Genetic Algorithms and Artificial Life. Artificial Life, 1 (3), 267-289 (1994).

[12] Nawa, N., Hashiyama, T. , Furuhashi, T. and Uchikawa, Y.: A Study on Fuzzy Rules Using Pseudo-Bacterial Genetic Algorithm with Adaptive Operator. Proceedings of the IEEE International Conference on Evolutionary Computation, 589-593. IEEE Press 1997.

[13] Nawa, N., Furuhashi, T. : Bacterial Evolutionary for Fuzzy System Design. Proceedings of the 1998 IEEE International Conference on Systems, Man and Cybernetics. IEEE Press 1998.

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

[15] Oates, M., Corne, D., Loader, R.: Skewed Crossover and the Dynamic Distributed Database Problem. In the Proceedings of the Fourth Int. Conference on Neural Networks and Genetic Algorithms, pp. 280-287. Springer-Verlag 1999.

[16] Russell, P. J.: Genetics. 5th edition, Addison-Wesley 1998.

[17] Simões, A. and Costa, E.: Transposition: A Biologically Inspired Mechanism to Use with Genetic Algorithms. Proceedings of the Fourth Int. Conference on Neural Networks and Genetic Algorithms, pp. 178-186. Springer-Verlag 1999.

[18] Simões, A. and Costa, E.: Transposition versus Crossover: An Empirical Study. 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), pp. 612-619, Orlando, Florida USA, CA: Morgan Kaufmann 1999.

[19] Simões, A. and Costa, E.: Enhancing Transposition Performance. Proceedings of the 1999 Congress on Evolutionary Computation, pp. 1434-1441. Piscataway, NJ: IEEE Press 1999.

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

[21] Smith, P.: Finding Hard Satisfiability Problems Using Bacterial Conjugation. Presented at the Workshop on Evolutionary Computing, University of Sussex, April 1996.

[22] Smith, P.: Conjugation - A Bacterially Inspired Form of Genetic Recombination. Late Breaking Papers of the First International Conference on Genetic Programming. Stanford University, California, July 1996.

[23] Voss, M. and Foley, C.: Evolutionary Algorithm For Structural Optimization. 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), pp. 678-685, Orlando, Florida USA, CA: Morgan Kaufmann 1999.

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

[25] Yoshikawa,T., Furuhashi, T. and Uchikawa, Y.: 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, 126-135. Springer-Verlag 1997.