

# Mutation, Transposition, and Recombination: An Analysis of the Evolutionary Dynamics

Cândida Ferreira

Gepsoft, 37 The Ridings  
Bristol BS13 8NU, UK  
candidaf@gepsoft.com

In H. J. Caulfield, S.-H. Chen, H.-D. Cheng, R. Duro, V. Honavar, E. E. Kerre, M. Lu, M. G. Romay, T. K. Shih, D. Ventura, P. P. Wang, Y. Yang, eds., *Proceedings of the 6th Joint Conference on Information Sciences, 4th International Workshop on Frontiers in Evolutionary Algorithms*, pages 614-617, Research Triangle Park, North Carolina, USA, 2002.

Gene expression programming (GEP) uses mutation, transposition, and crossover to create variation. Although there exists a large body of work in genetic algorithms concerning the roles of mutation and recombination, these results not only do not apply to GEP due to the genotype/phenotype representation but also seem to contradict the GEP experience. Therefore, and given the diversity of GEP operators, it is convenient to develop some kind of understanding of their power. The aim of this work is to help develop such an understanding and to show the evolutionary dynamics and the transforming power of each genetic operator, with their advantages and limitations.

## Introduction

Everybody agrees that, by and large, evolution relies on variation coupled with some kind of selection, and all evolutionary algorithms explore these fundamental assets. However, there is no agreement concerning the best way to create variation. The first evolutionary algorithms relied on mutation only and their recent developments claim that recombination has no general advantage over mutation (for a review see, e.g. [1]). In genetic algorithms (GAs), however, recombination is considered the more powerful of the two operators [5]. More recently, Spears [6] tries to conciliate both views, attributing to mutation and recombination equally important roles: the roles of disrupters (mutation) and constructors (recombination). To make matters worse, the classical genetic programming approach (GP) stresses out the prominence of crossover [4], whereas developmental genetic programming (DGP) is more inclined towards mutation [2].

In this work I show that, in GEP, mutation is by far the single most important operator whereas recombination has a very limited power.

## 1. Search in genotype/phenotype systems

In genotype/phenotype systems, the search space remains separated from the solution space, improving consider-

ably the performance of these systems [2, 3]. Moreover, the more unconstrained the genotype/phenotype mapping the more efficient the system as virtually any operator can be used to explore the search space, including mutation. In DGP, however, the genotype/phenotype mapping does not always result in syntactically correct phenotypes and much editing is necessary to remove invalid phenotypes. Consequently, mutation does not significantly outperform crossover.

On the other hand, GEP is a totally unconstrained genotype/phenotype system as all modifications made in the genotype always result in correct phenotypes and no editing whatsoever is necessary [3]. Therefore, GEP provides, for the first time, an artificial framework where unconstrained genotype/phenotype systems can be carefully analyzed and hopefully shed some light on the role of mutation and crossover in evolution.

## 2. General settings

To make this preliminary analysis of GEP operators, the following relatively simple test function was chosen:

$$y = a^4 + a^3 + a^2 + a$$

as it can be exactly solved using relatively small populations and relatively short evolutionary times and also because it is appropriate to study all the genetic operators, including operators specific of multigenic systems like gene recombination.

In all the experiments, a set of 10 random fitness cases chosen from the interval  $[-10, 10]$  was used ( $a$  values: -4.4229, 9.7485, -1.7641, -7.0436, -6.8656, -8.1246, 5.9982, -0.1057, 5.1629, -0.7231); the fitness function used is based on the absolute error and has a selection range of 100 and a precision of 0.01, giving a maximum fitness of 1000 [3]; the selection was made by roulette-wheel sampling coupled with simple elitism; a population size  $P$  of 50 individuals and an evolutionary time  $G$  of 50 generations were used; the success rate  $Ps$  of each experiment was evaluated over 100 independent runs;  $F = \{+, -, *, /\}$  and  $T = \{a\}$ ; and three-genic chromosomes of length 39 linked by addition were used. In the

experiments where transposition was switched on, three transposons with lengths 1, 2, and 3 were used. In the evolutionary dynamics, only successful runs were chosen and  $G$  was extended to 100 generations so that the dynamics could be better perceived.

### 3. Comparing mutation, transposition, and crossover

Due to the unconstrained genotype/phenotype mapping of GEP, several genetic operators can be easily implemented and Ferreira [3] uses seven: mutation, three kinds of transposition (IS, RIS and gene transposition), and three kinds of recombination (one-point, two-point and gene recombination). Most of these operators are important per se, and here they are going to be analyzed separately, but one of them – gene transposition – is only significant if used in conjunction with recombination and therefore will not be analyzed here.

The performance of six GEP operators is shown in Figure 1, and it clearly shows that mutation is by far the single most powerful operator, followed by RIS transposition and IS transposition, whereas recombination is the less powerful operator with gene recombination at the end of the line.

### 4. Evolutionary dynamics

The analysis of the evolutionary dynamics of some of the populations of Figure 1 can be helpful to gain some in-

sight into the adaptive strength of evolutionary systems. Obviously, the most interesting populations have the highest success rates. However, it would also be most useful to analyze the evolutionary dynamics of less efficient systems and compare them with the most efficient.

#### 4.1. Mutation

The evolutionary dynamics of four populations evolving under mutation are shown in Figure 2. In the first evolutionary dynamics,  $p_m = 0.001$ , the plot for average fitness closely accompanies the plot for the best fitness and, in later generations, with the exception of a small variation, the plots overlap. These populations are called *Moderately Innovative* as they allow the introduction of a little bit of genetic diversity among its members and evolve very sluggishly.

In the second dynamics,  $p_m = 0.005$ , we observe that, although closely accompanying the plot for best fitness, the plot for average fitness never touches the best fitness. Moreover, the oscillatory pattern on average fitness characteristic of GEP populations [3], is already evident even for such small variation rates. As shown in plot b of Figure 2, the success rate is in this case 47%. It can be said that, although adaptively healthy, the system is not very efficient. These systems are called *Healthy But Weak*. Note that the success rate increases progressively with the mutation rate until it reaches a plateau around  $p_m = 0.1$  and, in terms of dynamics, this is reflected in a more pronounced oscillatory pattern in average fitness and an increase in the gap between both plots. Indeed, around  $p_m = 0.1$  (Figure 2,

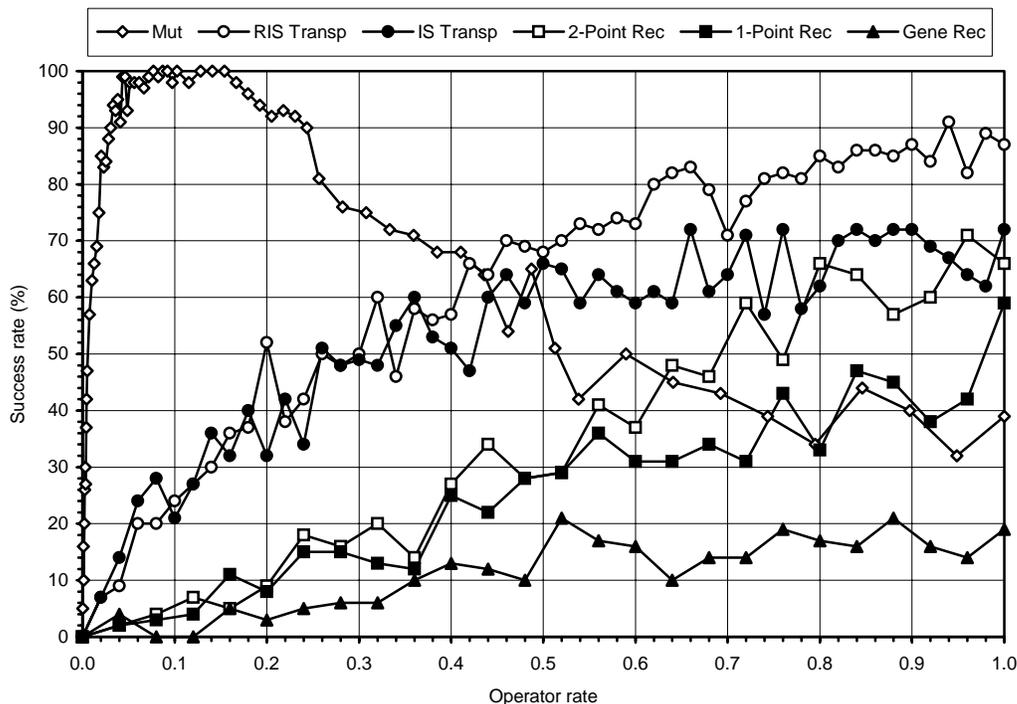
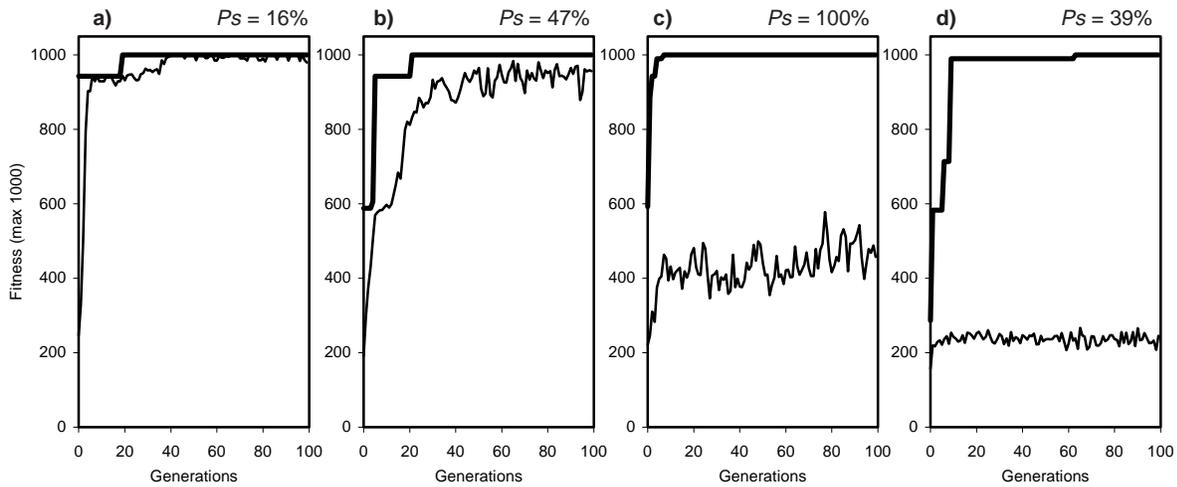


Figure 1. The transforming power of mutation, transposition, and crossover.



**Figure 2.** A gallery of evolutionary dynamics found in populations evolving by mutation. The success rate above each plot was determined in the experiment shown in Figure 1. **a)** *Moderately Innovative* dynamics,  $p_m = 0.001$ . **b)** *Healthy But Weak* dynamics,  $p_m = 0.005$ . **c)** *Healthy And Strong* dynamics,  $p_m = 0.1$ . **d)** *Totally Chaotic* dynamics,  $p_m = 1.0$ .

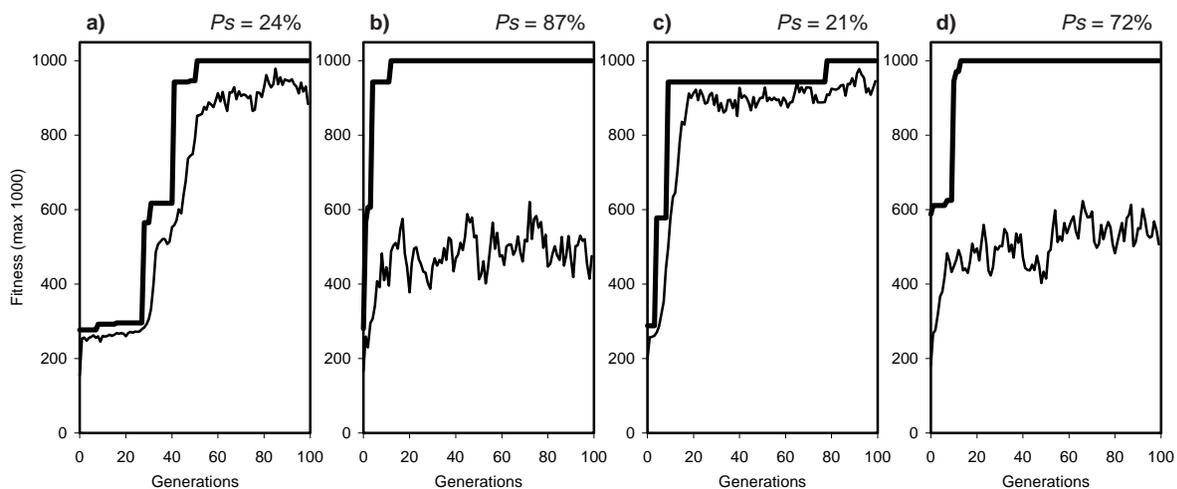
plot c), the system is at its best with a success rate of 100%. These populations are called *Healthy And Strong*. Note that the gap between best and average fitness is considerably wider than in plots a and b, and the capacity to evolve is also higher. Obviously, all populations evolving with maximum performance (approximately from  $p_m = 0.033$  to  $p_m = 0.20$ ) have dynamics of the type *Healthy And Strong*.

The last dynamics of Figure 2,  $p_m = 1.0$ , is an example of *Totally Chaotic* systems. Note that, despite elitism, this dynamics is completely chaotic, for elitism is insignificant as each generation a totally random population is generated. Note also the less pronounced oscillatory pattern in average fitness and that the gap between average and best fitness is now maximal.

#### 4.2. Transposition

As shown in Figure 1 and, against all expectations, RIS transposition is more efficient than IS transposition. Recall that with RIS transposition the root itself is always the target, modifying drastically the expression trees [3]. Note also that the transforming power of this kind of transposition is slightly less than mutation but superior to crossover.

Figure 3 compares the evolutionary dynamics obtained for small and high transposition rates. Note the appearance of the type *Healthy And Strong* for  $p_{ris} = 1.0$  (plot b) and for  $p_{is} = 1.0$  (plot d). The other plots were obtained for  $p_{ris} = 0.1$  (plot a) and  $p_{is} = 0.1$  (plot c), and are of the kind *Healthy But Weak*. In fact, and as observed for mu-



**Figure 3.** A gallery of evolutionary dynamics found in populations evolving by RIS (plots a and b) and IS transposition (plots c and d). The success rate above each plot was determined in the experiment shown in Figure 1. **a)** *Healthy But Weak* dynamics,  $p_{ris} = 0.1$ . **b)** *Healthy And Strong* dynamics,  $p_{ris} = 1.0$ . **c)** *Healthy But Weak* dynamics,  $p_{is} = 0.1$ . **d)** *Healthy And Strong* dynamics,  $p_{is} = 1.0$ .

tation, an increase in transposition rate results in an increase in the gap between best and average fitness.

### 4.3. Recombination

The performance of the three kinds of GEP recombination is shown in Figure 1. It is worth emphasizing that 2-point recombination is the most disruptive of the recombination operators and, as shown in Figure 1, it is also the most efficient kind of recombination. Not surprisingly, the most conservative gene recombination is also the less efficient.

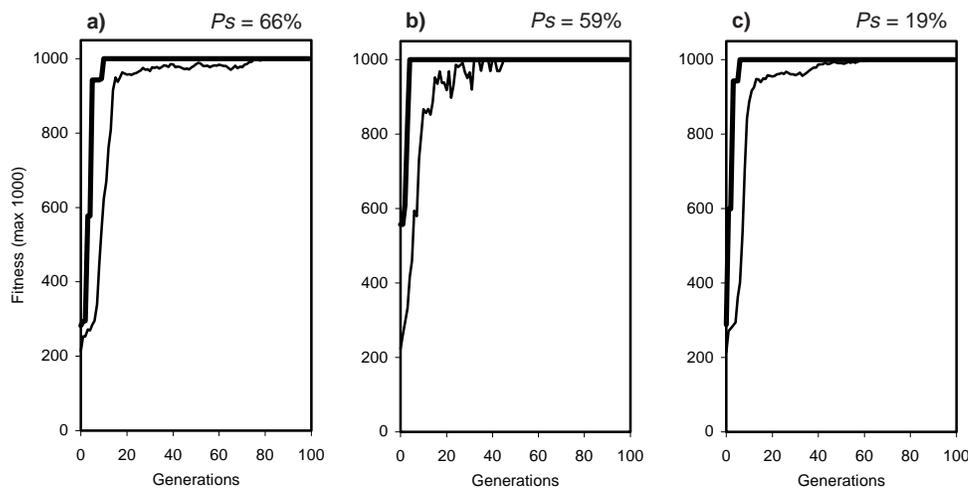
The dynamics characteristic of recombination (Figure 4) also show an extremely important feature of recombination, that is, the homogenizing effect of all kinds of recombination, from the most conservative to the most disruptive. For obvious reasons, these recombination-specific dynamics are called *Homogenizing* dynamics. Note that, in all cases, after a certain time the plot for average fitness overlaps the plot for best fitness. This indicates that the populations lost all the diversity and all the individuals have the same genetic makeup. Obviously, if populations converge to this stage before finding a good solution, they become irrevocably stuck in that point if no other, non-recombinatorial operators are available. As the small success rates obtained for recombination emphasize (Figure 1), when populations evolve by recombination only, most of the times, they converge before finding a good solution. This tells us that recombination should never be used as the only source of genetic variation.

Furthermore, it is worth noticing that GEP recombination, even the most conservative gene recombination, is more disruptive than the homologous recombination that occurs in nature during sexual reproduction as the exchanged genes rarely are homologous. One of the unsolved questions of biology is the role of sex in evolution and,

most of the times, biological sex in its overwhelming diversity is confounded with the homologous recombination that occurs during sexual reproduction. Consequently, many erroneously assume that homologous recombination creates great diversity. The comparison of GEP operators, especially GEP recombination operators, suggests that a more conservative recombination like homologous recombination would only be useful to maintain the status quo in periods of stasis.

## Bibliography

1. Bäck, T., *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*, Oxford University Press, 1996.
2. Banzhaf, W., Genotype-phenotype-mapping and neutral variation – A case study in genetic programming. In Y. Davidor, H.-P. Schwefel, and R. Männer, eds., *Parallel Problem Solving from Nature III*, Vol. 866 of *Lecture Notes in Computer Science*, Springer-Verlag, 1994.
3. Ferreira, C., 2001. Gene expression programming: A new adaptive algorithm for solving problems. *Complex Systems*, 13 (2): 87-129.
4. Koza, J. R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, Cambridge, MA: MIT Press, 1992.
5. Schaffer, J. D. and L. J. Eshelman, 1991. On crossover as an evolutionary viable strategy. *Proceedings of the Fourth International Conference on Genetic Algorithms*, 61-68, Morgan Kaufmann, La Jolla, CA.
6. Spears, W. M., Crossover or mutation? In D. Whitley, ed., *Foundations of Genetic Algorithms*, Vol. 2, 221-237, Morgan Kaufmann, 1992.



**Figure 4.** A gallery of evolutionary dynamics found in populations evolving by 2-point recombination (plot a), 1-point recombination (plot b), and gene recombination (plot c). The success rate above each plot was determined in the experiment shown in Figure 1. In all the plots the evolutionary dynamics are of the kind *Homogenizing*. **a)**  $p_{2r} = 1.0$ . **b)**  $p_{1r} = 1.0$ . **c)**  $p_{gr} = 1.0$ .