Sub-tree Swapping Crossover, Allele Diffusion and GP Convergence

Stephen Dignum and Riccardo Poli

Department of Computing and Electronic Systems,
University of Essex,
Wivenhoe Park, Colchester, CO4 3SQ, UK
{sandig,rpoli}@essex.ac.uk

Abstract. We provide strong evidence that sub-tree swapping crossover when applied to tree-based representations will cause alleles (node labels) to diffuse within length classes. For *a*-ary trees we provide further confirmation that all programs are equally likely to be sampled within any length class when sub-tree swapping crossover is applied in the absence of selection and mutation. Therefore, we propose that this form of search is unbiased - within length classes - for *a*-ary trees. Unexpectedly, however, for mixed-arity trees this is not found and a more complicated form of search is taking place where certain tree shapes, hence programs, are more likely to be sampled than others within each class. We examine the reasons for such shape bias in mixed arity representations and provide the practitioner with a thorough examination of sub-tree swapping crossover bias. The results of this, when combined with crossover length bias research, explain Genetic Programming's lack of structural convergence during later stages of an experimental run. Several operators are discussed where a broader form of convergence may be detected in a similar way to that found in Genetic Algorithm experimentation.

Keywords: Genetic Programming, Search, Crossover Bias, Allele Diffusion, Convergence.

1 Introduction

An intrinsic feature of traditional Genetic Programming (GP) is its variable-size tree-based representation [6,8]. Sub-tree swapping crossover has also, from the inception of GP, been the predominant genetic operator [4,5]. It is essential, therefore, for GP practitioners to understand the biases inherent in using this form of representation and the primary variation operator applied to it.

Recent research has shown that sub-tree swapping crossover will sample exponentially more shorter programs for *a*-ary trees[1] when applied to a flat fitness landscape in the absence of mutation [7], i.e., when its bias is isolated. This was extended by generalisation to mixed-arity trees in [2] and to true length-classes

[1] Representations made up of internal nodes that have a single common arity, e.g., 2 for the case of Boolean induction problems which use the functions AND, OR, etc.

G. Rudolph et al. (Eds.): PPSN X, LNCS 5199, pp. 368–377, 2008.

(from internal node counts) in [3]. Strong empirical support has been found for each generalisation.

One can divide the space of all possible programs into subsets. As we have discussed, one way is to group programs by the number of nodes in the tree representing them. We will call each such set a *length class*. A finer classification would be to divide the programs by their tree shape. This is what we will call a *shape class*. Each program shape is characterised by the number of primitives/nodes of each arity it contains. This can provide a (non-unique) signature for the shape, which we will call an *arity histogram*. Of course, all shapes with a particular arity histogram also have an identical number of nodes. So, if we group programs by their arity histograms we obtain a sub-division of the program space which is between the length class and the program shape in that many shapes (but only one program size) can correspond to an arity histogram.[2]

An assumption (indirectly corroborated numerically) of the original hypothesis in [7] was that all tree shapes within a particular length class for a-ary trees would be equally likely, as all correlations present within the shapes would be removed by the crossover operator. This implies a diffusive process where any node is equally likely to be in any position within the tree shape. If this diffusion process occurs we can assume that sub-tree swapping crossover is unbiased in its exploration of the search space within each length class, i.e., it will explore all programs with equal probability within each length.

The appropriateness of bias (or lack of) is problem dependent (see No Free Lunch Theorems [11]). However, characterising the bias allows us to understand why GP has been successful in solving certain problems or classes of problems. Understanding such bias also allows us to explain how GP searches when areas of neutrality are reached or when selection reduces fitness variance in the population during the later stages of a GP run. It also provides a starting point in the analysis of the effects of combinations of GP operators.

Within Section 2 we briefly explain current findings for length bias. In Section 3 we use a cartesian node reference system to identify all possible positions within a tree. From this we provide evidence of a diffusion process showing that all correlations between nodes are broken by repeated application of sub-tree swapping crossover in the absence of selection and other reproduction operators.

We turn our attention to unique shapes within length classes in Section 4. As predicted, shape classes are shown to have equal occurrence within each length class for a-ary trees, although as predicted in [7] shapes within smaller lengths are more widely sampled than those of larger lengths. Shapes within length classes for mixed-arity trees, however, are not sampled equally. We find that only those within each distinct arity histogram class are sampled in such a way. This extends current research showing us that the repeated application of crossover distributes trees according to their arity histogram. Earlier results for a-ary representations are a special case of this more general result.

[2] Naturally, the distinction between length-class and arity histogram disappears for a-ary trees. Also, in both the single and the mixed-arity cases, the number of terminals is always determined by the rest of the arity histogram.

From our characterisation of crossover's biases we are in a position to explain the lack of structural convergence of GP solutions during experimentation [1, page 278]. Structural convergence is an effect seen in other forms of evolutionary search, notably, Genetic Algorithms (GAs) where it is often used as a stopping criterion for runs. This is discussed in Section 5 along with potential broader convergence detection measures, while in Section 6 we summarise our findings.

2 Length Distributions

In [7] we provided a mathematical model with strong experimental evidence showing that the repeated application of standard sub-tree swapping crossover with uniform selection of crossover points will push a population of a-ary trees towards a limiting distribution of tree sizes called a *Lagrange distribution of the second kind*. This distribution shows a strong tendency to sample programs of small sizes, programs including only one terminal being sampled most often.[3]

This result was generalised in [2] to show that a similar distribution exists for mixed arity trees. As an illustration, Figure 1 shows a theoretical distribution with empirical verification for a population with a mix of internal nodes with arities of 2, 2, and 3, i.e., for that of the Artificial Ant problem [4].

The predictive model used to produce the distribution in Figure 1 is

$$\Pr_g\{n\} = (1-\bar{a}p_{\bar{a}})\frac{\Gamma(\bar{a}n+2)}{\Gamma((\bar{a}-1)n+2)\Gamma(n+1)}(1-p_{\bar{a}})^{(\bar{a}-1)n+1}p_{\bar{a}}^{n} \qquad (1)$$

where $\Pr_g\{n\}$ is the probability of selecting an individual with n internal nodes, $\Gamma()$ is the Gamma function, $\bar{a}$ is the average of arities in the initial population before crossover is applied, μ_0 is the initial mean tree size within that population, and $p_{\bar{a}}$ is used to simplify the formula and is defined as follows

$$p_{\bar{a}} = \frac{2\mu_0 + (\bar{a}-1) - \sqrt{((1-\bar{a})-2\mu_0)^2 + 4(1-\mu_0^2)}}{2\bar{a}(1+\mu_0)} \qquad (2)$$

For $\bar{a} > 1$ the function in Equation (1) is decreasing. It was shown in [2] that increasing the initial mean program size reduces its slope, hence, reducing the bias to sample smaller programs.

Finally, the distribution was generalised once more in [3] to provide predictions based on exact lengths rather than internal nodes. While for a-ary trees there is a one-to-one mapping between length and internal nodes, for mixed arity trees there are occasional, if minor, discrepancies at shorter lengths and the generalisation is approximate. Nonetheless, the match between the model and experimental results is very good, any discrepancies disappearing as program size increases. The reasons for the minor deviations at shorter lengths are explained in the following sections.

[3] The 90/10 node-selection policy commonly used in GP to counter this effect was also shown in [2] to have little effect on the sampling of all but the smallest classes.

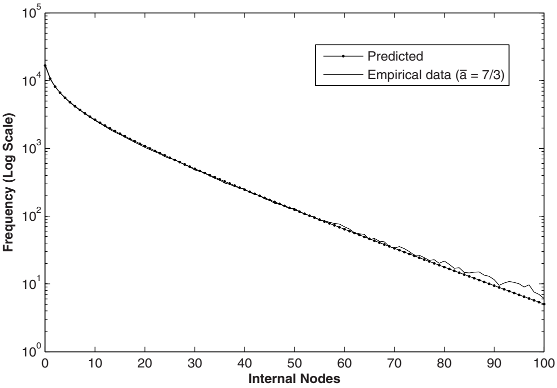


Fig. 1. Comparison between empirical and predicted internal node distributions, in the absence of selection and mutation, for trees made up of a mix of 2, 2, and 3 arity functions ($\bar{a}$=7/3) initialised with FULL method (depth = 3, initial mean size μ_0 = 21.48, mean size after 500 generations μ_{500} = 23.51). Population Size 100,000 individuals, empirical results averaged over 20 runs.

3 Allele Diffusion

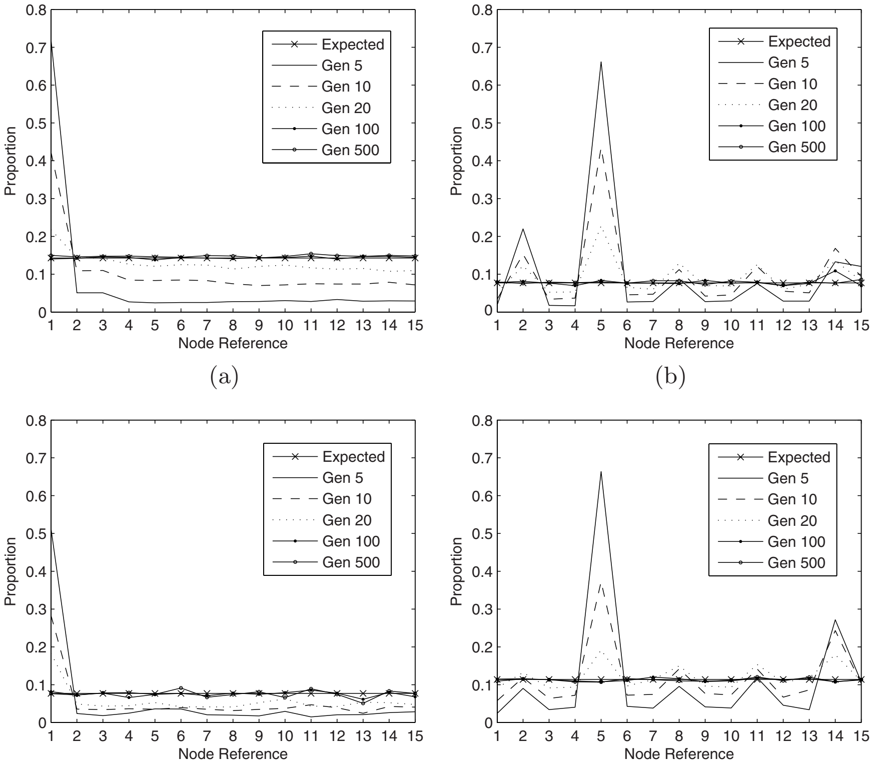
Our first task is to test the assumption that crossover will remove any correlations between nodes ensuring that all node labels are equally likely to be found at any position within trees created purely from the application of crossover.

Earlier work provided theoretical and empirical evidence to support this claim for linear GP [10], where only internal nodes of arity 1 were used. This, of course, is a specific case of the a-ary assertion in [7].

We have chosen to implement the technique used in [10] where a node marker or 'dye' is applied at specific positions within trees during initialisation. The amount of dye is then recorded for each node position in subsequent generations.

With linear GP it is possible to compare directly node positions within length classes. This is not true, however, for a-ary trees or those with mixed arities. We have chosen, therefore, to implement a cartesian node reference system to assign unique node positions for all possible trees based upon the maximum arity that may be used. The exact method is described in [9]. However, it can simply be described as producing a template based on a maximal tree, i.e., one where only the largest arity is used without terminals up until a maximum depth. Each node is assigned a unique integer number in the order of left-to-right breadth-first traversal, 1 being the position of the root node.

For each set of experiments a population of 100,000 individuals was used. Dye was placed either at reference 1 (the root node) or at reference 5. These positions have been chosen carefully to ensure dye was applied once to every tree during initialisation for all of our arity mixes, hence, simplifying theoretical calculations. For all experimentation a flat fitness landscape was used and sub-tree swapping crossover with uniform selection of crossover points was applied



(a) (b)

(c) (d)

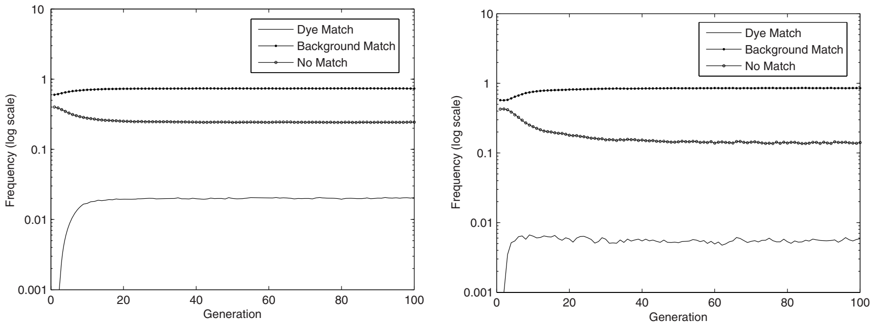
Fig. 2. Plots of the relative proportion of non-terminal dye alleles vs node references for: (a) 2-ary programs of length 11, initial dye reference 1, (b) 3-ary programs of length 13, initial dye reference 5, (c) mixed arity 1, 2, 3, 4 & 5 programs of length 11, initial dye reference 1, (d) mixed arity 2, 2 & 3 programs of length 13, inital dye reference 5. Note, selected tree lengths are smaller than the smallest trees created by the initialisation method hence data are not recorded for generation 0.

with no mutation or reproduction. All programs were initialised using the FULL method with a depth of 3 (depth 0 being the root node) and all results have been averaged over 20 independent runs.

In Figure 2a we can see that for the proportion of internal nodes with dye, for 2-ary trees of length 11, we move rapidly to our expected value at each of the first fifteen possible node references.[4]

For 2-ary trees initialised with the FULL method with depth 3, each tree will have only one dye node for each of the possible seven internal nodes, hence,

[4] Note, it is possible for internal nodes to reach a position of 31 using our reference system for 2-ary trees of length 11. A limit of 15 was chosen for consistency across experimentation.



(a) (b)

Fig. 3. Plots of the mean relative frequency of co-occurrence of pairs of non-terminal alleles vs. generation for 2-ary (a) and mixed arity 1, 2, 3 , 4 & 5 (b) programs of length 11. Population initialised as Figure 2.

after diffusion has taken place we expect all positions to have a dye proportion of 1/7 for internal nodes. Consistently similar results, i.e., convergence to predetermined predicted proportions are seen in additional experiments for 3-ary trees, and mixed arity trees of 1, 2, 3, 4 & 5 and 2, 2 & 3 arity nodes.[5] These are shown in Figures 2b-d respectively.

We next turn our attention to co-occurrence of pairs of non-terminals, i.e., whether we can consistently see any correlation between dye positions. In order to do this for each generation a 15 by 15 matrix is produced. Each row and column records the first 15 positions within the node reference system. For the first row we determine if the first node is dye or background then for each column we then determine whether this matches for any of the other positions and record the match, or lack of, in the corresponding position in our matrix, i.e., row determined by node under investigation, column for nodes to be matched. Diagonals in the matrix are ignored as we will always obtain a match. In Figure 3a we can see that for 2-ary trees initialised with dye at the root position we quickly move to values predicted by a diffusive process. Dye sits on the diagonal for the initial generation and hence is not recorded but then we apply crossover and after approximately 20 generations we have reached our theoretical proportions: $(1/7)^2 \approx 0.020408$ for dye matching, $(6/7)^2 \approx 0.73469$ for background matching, and $2(1/7)(6/7) \approx 0.24490$ for no match. The same is true for our mixed arity trees. For example in Figure 3b our population of 100,000 individuals was initialised with an average of 1,297,856.85 internal nodes, 100,000 of which where marked with dye, our theoretical value for dye co-occurrence is $(100,000/1,297,856.85)^2 \approx (0.07705)^2 \approx 0.00594$. Background matching is, therefore, $(1 - 0.07705)^2 \approx (0.92295)^2 \approx 0.85184$ and finally our no match value will be $2(0.07705)(0.92295) \approx 0.14223$.

[5] All experimentation shown was subjected to a $\chi^2_{10\%}$ test which showed support for the assertion that the first 15 positions, at generation 100, would each contain a number of nodes determined by initial population proportions.

Each of these values is also obtained within 10 to 20 generations. Similar results were also found for our 3-ary and 2, 2 & 3 mixed arity experiments.[6] See [10] for similar results for linear GP, i.e., 1-ary trees.

4 Shape Bias

There is one final aspect of sub-tree swapping crossover that we can analyse before we complete our picture: how we sample shapes within length classes. The length distribution described in [7] is derived from an expectation that all shapes will be sampled uniformly within length classes for a-ary trees. In Figure 4, we can indeed provide experimental evidence for 2-ary trees for our length classes chosen. However, looking at mixed arities we can see that there is a distinct bias to sample certain shape classes within each length. It was found, however, (see Table 1 as an example) that shapes with same arity histogram would be sampled uniformly. This shape bias for mixed arities is easily explained if we look at the dynamics of the proportion of primitives of each arity in the population. On average this form of crossover will replace as much as it removes; this also holds true for node arities. To illustrate, see Figure 5 as an example of how the proportion of primitives of each arity stays constant in a population when sub-tree swapping crossover only is applied for our mixed arity experiments described earlier. There is, therefore, no bias to remove or resample certain higher or lower arities. So, not only does average size remain constant under repeated application of crossover, but also the proportions of each arity will remain constant within the population. Therefore, any (note, highly sampled) smaller shapes without

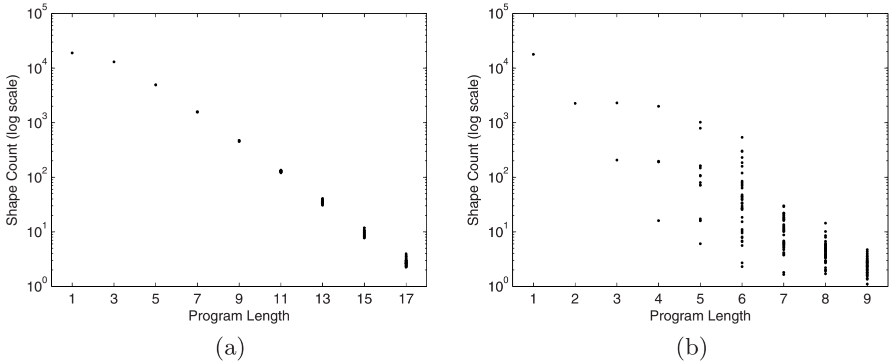
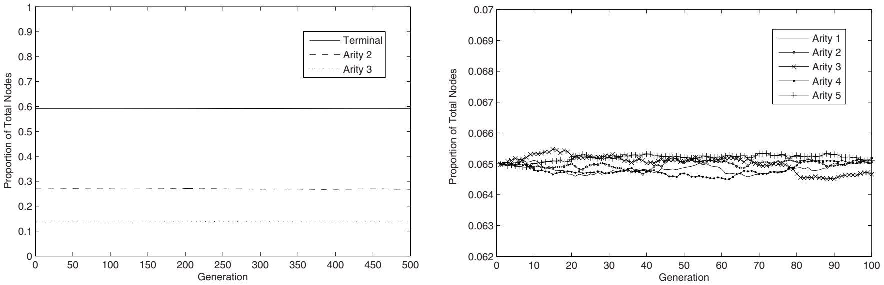


Fig. 4. Scatter plots of shape counts for 2-ary (a) and mixed arity 1, 2, 3, 4 & 5 (b) programs, first 9 possible lengths at generation 500. Population initialised as Figure 2. Note, there are far more possible shapes for larger length classes. Also, these classes are sampled far less often than those of smaller lengths.

[6] For all experiments tree lengths up to a maximum of 40 nodes were analysed, each showed similar results.

Table 1. Averaged counts at generation 500 for all program shapes for 2, 2 & 3 arity programs of length 7. Population initialised as in Figure 2.

S-Expression	Count
(2 0 (2 0 (2 0 0)))	407.10
(2 0 (2 (2 0 0) 0))	407.95
(2 (2 0 0) (2 0 0))	401.05
(2 (2 0 (2 0 0) 0))	404.25
(2 (2 (2 0 0) 0) 0)	410.40
(3 0 0 (3 0 0 0))	258.75
(3 0 (3 0 0 0) 0)	258.05
(3 (3 0 0 0) 0 0)	258.75



(a) (b)

Fig. 5. Plots of the proportions of arities for each generation. (a) shows the first 500 generations for a population initialised with 2, 2 & 3 arities. (b) shows the first 100 generations of a population initialised with 1, 2, 3, 4 & 5 arities, note the highly reduced scale in this example. Due to the reduced scaling terminals are not shown in (b) but follow a consistent proportion as shown in (a) in this case centering tightly around a proportion of 0.675. Populations initialised as in Figure 2.

an equal proportion of arities, or those that can be produced using only a single arity, will reduce those node arities available for larger classes. Further work is required to produce a model to exactly predict such proportions. However, we do know that the generalised model for mixed arities (Equation 1) has been corroborated by extensive empirical work, so such a model must explain why such a generalisation has been successful.

5 Convergence

First suggested in [10], we can now provide strong evidence that GP's inability to structurally converge is caused primarily through crossover's bias to first distribute a population in terms of length and arity histogram and then to diffuse node labels within those classes. As fitness converges during the later stages of a run, crossover sampling will become predominant. Hence, the processes described in this paper will prevent any structural convergence taking place. No matter how strong the selection scheme, e.g. even if the mating pool was populated solely by copies of a single individual (say by using a tournament size equal to that of the population), the resulting child population created by sub-tree swapping crossover would first contain individuals of differing lengths and secondly node labels would be dispersed within those individuals.[7] This would not be true in a GA system using n-point crossover acting on traditional fixed length vector representations as there is no opportunity to alter individual lengths or to move node labels to different locations.

Although GP using sub-tree swapping crossover will prevent convergence to a single syntactic structure, it will start to search within ever tighter bounds and begin to resample heavily smaller classes (see [3] for details). With this in mind we can suggest possible run stopping criteria based solely on convergence as found in GAs. A very simple method would be to determine the undue influence of crossover by detecting a greater ratio of smaller programs. An inexpensive resampling measure based on simple program hashes could also be used, possibly causing run termination when a program has been resampled a pre-specified number of times. Additional more sophisticated methods may look at the length distribution as a whole, i.e., a convergence to the theoretical distribution or in conjunction with fitness measures such as a corresponding reduction in fitness variance.

6 Conclusions and Future Work

This paper has analysed the biases presented by GP sub-tree swapping crossover. We have provided strong evidence that there is a diffusive process that takes place within length classes when sub-tree swapping crossover is repeatedly applied to a flat fitness landscape in the absence of selection. All node labels (alleles) are equally likely to be found within any possible node position for each length class.

We now know that program shapes will be uniformly sampled within arity histogram classes. a-ary trees are a special case in that there is only one arity histogram per length class. Hence, programs will be sampled uniformly within each length. This, however, is not true for mixed arities and a more sophisticated process is taking place. The reasons for this lie within the constant population proportions of each arity during each generation and the highly sampled smaller programs with unequal arity proportions.

In the future we hope to be able to develop a mathematical model similar to that described in [7] to provide the probability of an individual containing a certain proportion of internal node arities. This model will need to explain the theoretical and empirical results found within this paper and those presented in [7,2,3].

Although we now know that GP using sub-tree swapping crossover is highly unlikely to converge in terms of individual program structure, we do have an understanding of a broader, population based, form of structural convergence. This

[7] Barring the unlikely situation where the same crossover points are chosen in all cases.

allows us to propose a set of convergence measures that may be used for stopping conditions similar to those found in GA experimentation. Further research is required to establish the effectiveness of such measures.

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, San Francisco (1998)
2. Dignum, S., Poli, R.: Generalisation of the limiting distribution of program sizes in tree-based genetic programming and analysis of its effects on bloat. In: Thierens, D., et al. (eds.) GECCO 2007: Proceedings of the 9th annual conference on Genetic and evolutionary computation, London, vol. 2, pp. 1588–1595. ACM Press, New York (2007)
3. Dignum, S., Poli, R.: Crossover, sampling, bloat and the harmful effects of size limits. In: O'Neill, M., Vanneschi, L., Gustafson, S., Esparcia Alcázar, A.I., De Falco, I., Della Cioppa, A., Tarantino, E. (eds.) EuroGP 2008. LNCS, vol. 4971, pp. 158–169. Springer, Heidelberg (2008)
4. Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge (1992)
5. Koza, J.R.: Genetic Programming II: Automatic Discovery of Reusable Programs. MIT Press, Cambridge (1994)
6. Langdon, W.B., Poli, R.: Foundations of Genetic Programming. Springer, Heidelberg (2002)
7. Poli, R., Langdon, W.B., Dignum, S.: On the limiting distribution of program sizes in tree-based genetic programming. In: Ebner, M., O'Neill, M., Ekárt, A., Vanneschi, L., Esparcia-Alcázar, A.I. (eds.) EuroGP 2007. LNCS, vol. 4445, pp. 193–204. Springer, Heidelberg (2007)
8. Poli, R., Langdon, W.B., McPhee, N.F.: A field guide to genetic programming (With contributions by J. R. Koza) (2008), `http://lulu.com`, `http://www.gp-field-guide.org.uk`
9. Poli, R., McPhee, N.F.: General schema theory for genetic programming with subtree-swapping crossover: Part I. Evolutionary Computation 11(1), 53–66 (2003)
10. Poli, R., Rowe, J.E., Stephens, C.R., Wright, A.H.: Allele diffusion in linear genetic programming and variable-length genetic algorithms with subtree crossover. In: Foster, J.A., Lutton, E., Miller, J., Ryan, C., Tettamanzi, A.G.B. (eds.) EuroGP 2002. LNCS, vol. 2278, pp. 212–227. Springer, Heidelberg (2002)
11. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1(1), 67–82 (1997)