Examining the Diversity Property of Semantic Similarity Based Crossover

Tuan Anh Pham[1], Quang Uy Nguyen[2], Xuan Hoai Nguyen[3], and Michael O'Neill[4]

[1] Centre of Information Technology, Military Academy of Logistics, Vietnam
[2] Faculty of IT, Military Technical Academy, Vietnam
[3] IT Research and Development Center, Hanoi University, Vietnam
[4] Natural Computing Research & Applications Group, University College Dublin, Ireland
{anh.pt204,quanguyhn,nxhoai}@gmail.com, m.oneill@ucd.ie

Abstract. Population diversity has long been seen as a crucial factor for the efficiency of Evolutionary Algorithms in general, and Genetic Programming (GP) in particular. This paper experimentally investigates the diversity property of a recently proposed crossover, Semantic Similarity based Crossover (SSC). The results show that while SSC helps to improve locality, it leads to the loss of diversity of the population. This could be the reason that sometimes SSC fails in achieving superior performance when compared to standard subtree crossover. Consequently, we introduce an approach to maintain the population diversity by combining SSC with a multi-population approach. The experimental results show that this combination maintains better population diversity, leading to further improvement in GP performance. Further SSC parameters tuning to promote diversity gains even better results.

Keywords: Genetic Programming, Semantic, Diversity, Locality.

1 Introduction

Similar to other evolutionary algorithms, it has been found for Genetic Programming (GP) that there two crucial properties that strongly affect its performance, namely, the diversity of a population [4,8] and the locality of operators [5,14]. The diversity of a population, which is directly affected by search operators, represents its ability to explore different parts of the search space while the locality of an operator exhibits its ability to focus on exploiting a specific area of the search space. Intuitively, these two properties seem to be contradictory. It means that an approach that maintains high diversity in the population often has low locality in search operators and vice versa.

In a recent work [10], Uy et al. proposed a new semantic based crossover for GP, Semantic Similarity based Crossovers (SSC), with the main objective to improve the locality of the standard subtree crossover. It has been shown [10] that SSC achieved its objective and increased locality in the crossover operator leading to a significant improvement in GP performance. To counter the

K. Krawiec et al. (Eds.): EuroGP 2013, LNCS 7831, pp. 265–276, 2013.

effect of reducing search diversity while enforcing locality, SSC also forbid the exchange of two semantically equivalent subtrees. However, there are still three open questions related to SSC [10], which are:

1. How the diversity of the population is impacted by SSC with its focus on operator locality?
2. Is there a way to reduce this impact on diversity while maintaining SSC locality?
3. If we can balance operator locality with population diversity will we then see additional gains in the performance of GP?

This paper tries to address these three questions. We first analyse the diversity of GP populations when the crossover operator is SSC. We then propose an approach to retain the population diversity by combining SSC with multi-population Genetic Programming. The remainder of the paper is organised as follows. In the next section, we briefly review the previous work on diversity in GP. Section 3 details Semantic Similarity based Crossovers. The experimental settings are detailed in Section 4. The results of the experiments are presented and discussed in Section 5. Finally, Section 6 concludes the paper and highlights some potential future work.

2 Related Work

It is widely believed that maintaining high population diversity is important for evolutionary algorithms [4]. Rapid loss of diversity, especially semantic diversity, has been suggested as the main cause for premature convergence of GP evolutionary search [12]. Consequently, GP systems may be trapped into local optima. When considering the diversity in GP populations, it is important to distinguish between two types of diversity. The first is syntactic or genotypic diversity and the second is the behavioural or phenotypic diversity. In this paper, we will focus on the later type of diversity. We argue that the second type of diversity is more critical to GP's behaviour than the first, as it is easy to find programs that are all syntactically distinct, yet have identical semantics.

Controlling (syntactic) diversity has been considered since the early days of GP. Much of earlier work focused on the initialisation phase of GP. Koza introduced the well-known Ramped-Half-and-Half technique for creating the initial GP population to reduce the occurrence of duplicated trees [6]. O'Reilly and Oppacher [11] and Poli and Langdon [12] tested various crossover operators to study their impact on syntactic diversity, They showed that standard crossover (SC) often leads to loss of diversity, hence is not an ideal operator.

Rosca [13] proposed a method to measure semantic diversity in GP population using phenotype entropy. Langdon [7] used (explicit) fitness sharing to preserve diversity. It clusters the population into a number of groups, based on their similarity with respect to a fitness-based metric. Members of the same group are penalized by being forced to share fitness, while isolated individuals retain their full fitness. McKay [9] used implicit fitness sharing, in which the reward for each fitness case is shared by all individuals that give the same output.

More recently, semantic diversity has received more attention from GP researchers. Burke et al. [2] conducted an analysis on the effect of different diversity measures on fitness. They showed that there is a strong correlation between entropy and the edit distance on the one hand, and change in fitness on the other. Gustafson et al. [4] examined the possible effects of sampling both unique structures and behaviours in GP. The behaviour sampling results helped to explain previous diversity research and suggest new ways to improve search. Similarly, Looks [8] proposed a new method for sampling semantically unique individuals in GP, by generating a number of unique minimal programs, then combining random programs with these minimal programs to generate the population. He argued that it increases the behavioral diversity of the population, leading to significant gains in GP performance. Beadle and Johnson proposed Semantic Driven Crossover (SDC) [1]. In SDC, the semantic equivalence of the offspring produced by crossover with their parents is checked by transforming them to Reduced Ordered Binary Decision Diagrams (ROBDDs). If two trees reduce to the same ROBDD, they are semantically equivalent. If the offspring are equivalent to their parents, they are discarded and the crossover is restarted. This process is repeated until semantically new children are found. The authors argued that this results in increased semantic diversity in the evolving population, and a consequent improvement in GP performance. Overall, promoting diversity, especially semantic diversity, is important and often leads to beneficial results.

3 Methods

This section briefly presents Semantic Similarity based Crossover (SSC) more details of SSC could be found in [10]

3.1 Measuring Semantics

The *Sampling Semantics* of any (sub)tree could be defined as follows:

Let F be a function expressed by a (sub)tree T on a domain D. Let P be a set of points sampled from domain D, $P = \{p_1, p_2, ..., p_N\}$. Then the *Sampling Semantics* of T on P on domain D is the set $S = \{s_1, s_2, ..., s_N\}$ where $s_i = F(p_i), i = 1, 2, ..., N$.

The values of two parameters N and P are dependent on problem. In this paper, N is set as the number of fitness cases of problems (20 points), and we choose the set of points P uniformly randomly from the problem domain.

Based on sampling semantics (SS), *Sampling Semantics Distance* (SSD) between two subtrees could be defined. Let $U = \{u_1, u_2, ..., u_N\}$ and $V = \{v_1, v_2, ..., v_N\}$ be the SS of $Subtree_1(St_1)$ and $Subtree_2(St_2)$ on the same set of evaluating values, then the SSD between St_1 and St_2 is defined as follows [10]:

$$SSD(St_1, St_2) = \frac{|u_1 - v_1| + |u_2 - v_2| + ... + |u_N - v_N|}{N} \tag{1}$$

Thanks to SSD, a relationship between two subtree called *Semantic Similarity* is defined. Two subtrees are semantically similar on a domain if their SSD on the same set of points in that domain lies within a positive interval. The formal definition of semantic similarity (SSi) between subtrees St_1 and St_2 is as follows:

$$\begin{aligned} SSi(St_1, St_2) = \textbf{if } & \alpha < \text{SSD}(St_1, St_2) < \beta \\ & \textbf{then } \text{true} \\ & \textbf{else } \text{false} \end{aligned}$$

here α and β are two predefined constants, known as the *lower* and *upper bounds* for semantic sensitivity, respectively. In this paper, we set $\alpha = 10^{-4}$ and $\beta = 0.4$ which are good values found in the previous experiments [10].

3.2 Semantic Similarity Based Crossover

In [10], SSC was proposed to improve the locality of crossover. It was an extension of Semantic Aware Crossover [17] in two ways. Firstly, when two subtrees are selected for crossover, their semantic similarity, rather than semantic equivalence as in SAC, is checked. Secondly, as semantic similarity is more difficult to satisfy than semantic equivalence, so repeated failures may occur. Thus SSC uses multiple trials to find a semantically similar pair, only reverting to random selection after passing a bound on the number of trials. Algorithm 1 shows how SSC operates in detail. In our experiments, the value of Max_Trial was set to 12, with this value having been calibrated by earlier experiments as the value for its good performance [10].

4 Experimental Settings

To investigate the diversity property of SSC, we used eight real-valued symbolic regression problems. The problems and training data are shown in Table 1. These functions were taken from previous work on using semantics based operators in GP [10].

The GP parameters used for our experiments are shown in Table 2. It should be noted that the raw fitness is the mean of absolute error on all fitness cases. Therefore, the smaller values are better. For each problem and each parameter setting, 100 runs were performed.

We divided our experiments into two sets. The first is to analyse the diversity property of SSC and the second aims to test a method for maintaining diversity of the population by combining SSC with multi-population GP.

Algorithm 1. Semantic Similarity based Crossover

select Parent 1 P_1;
select Parent 2 P_2;
Count=0;
while *Count<Max_Trial* **do**
    choose a random crossover point $Subtree_1$ in P_1;
    choose a random crossover point $Subtree_2$ in P_2;
    generate a number of random points (P) on the problem domain;
    calculate the SSD between $Subtree_1$ and $Subtree_2$ on P
    if $Subtree_1$ *is similar to* $Subtree_2$ **then**
        execute crossover;
        add the children to the new population;
        return true;
    else
        Count=Count+1;
if *Count=Max_Trial* **then**
    choose a random crossover point $Subtree_1$ in P_1;
    choose a random crossover point $Subtree_2$ in P_2;
    execute crossover;
    return true;

5 Results and Discussion

This section first analyses the diversity property of SSC and then introduce a method for maintaining the diversity of the population using SSC. After that, the issue of parameter tuning of SSC is addressed.

5.1 Diversity Analysis

As previously discussed, phenotypic diversity is often more important than genotypic diversity, in this paper, we analyse the diversity property of SSC using the phenotypic measurement proposed in Rosca [13]. The population phenotypic diversity is measured as

$$E(P) = -\sum_k p_k.log(p_k) \qquad (2)$$

where the population is partitioned according to fitness value, and p_k is the proportion of the population that have the fitness value in the fitness partition k^{th}. In this experiment we partitioned the population into 10 equal parts from the smallest fitness value to the greatest.

Figure 1 shows how the diversity of the population changed in GP with SSC (shorthanded as SGP) and GP with standard crossover for functions $F2$ and

Table 1. Symbolic Regression Functions

Functions	Training Data
$F1 = x^3 + x^2 + x$	20 random points $\subseteq$ [-1,1]
$F2 = x^4 + x^3 + x^2 + x$	20 random points $\subseteq$ [-1,1]
$F3 = x^5 + x^4 + x^3 + x^2 + x$	20 random points $\subseteq$ [-1,1]
$F4 = x^6 + x^5 + x^4 + x^3 + x^2 + x$	20 random points $\subseteq$ [-1,1]
$F5 = (x + 1)^3$	20 random points $\subseteq$ [-1,1]
$F6 = cos(3x)$	20 random points $\subseteq$ [-1,1]
$F7 = 2sin(x)cos(y)$	20 random points $\subseteq$ [-1,1]
$F8 = x^4 - x^3 + y^2/2 - y$	20 random points $\subseteq$ [-1,1]

Table 2. Run and Evolutionary Parameter Values

Parameter	Value
Population size	500
Generations	50
Selection	Tournament
Tournament size	3
Crossover probability	0.9
Mutation probability	0.05
Initial Max depth	6
Max depth	15
Max depth of mutation tree	5
Non-terminals	+, -, *, / (protected version), sin, cos, exp, log (protected version)
Terminals	X, 1
Raw fitness	mean absolute error on all fitness cases
Trials per treatment	100 independent runs for each value

$F4$ [1]. It can be seen from the figure that as the evolution progressed the population diversity decreased and population diversity of SGP was constantly lower than GP. It is understandable as the main objective of SSC is to improve the locality of crossover in GP, i.e to generate children that are not largely different from their parents. This results confirm our intuition that GP with SSC has to sacrifice some diversity for its contradictory counterpart - locality.

5.2 Maintaining Diversity for SSC

The previous subsection showed that using SSC in GP results in the loss of population diversity. Therefore, improving its diversity while maintaining its

[1] The figures for other test functions are similar and due to space limits, they are not shown here.

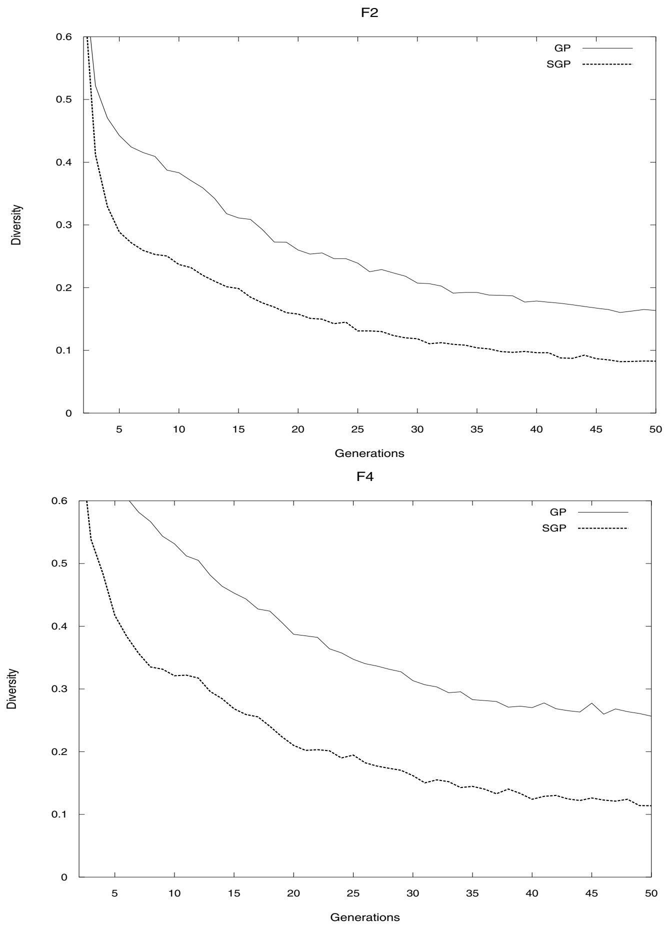

Fig. 1. The diversity of SSC compared to standard crossover

locality is potentially performance advantageous. To achieve this objective it is tempting to combine SSC with some diversity promotion mechanism which does not modify the crossover operator (e.g., a mechanisms that operates on the population structure). In this paper, we combine a multi-population approach with SSC to improve the diversity of SGP.

The idea of dividing a large population into several sub-populations is not new in itself e.g., see [15], which describes an island model approach. Individuals are allowed to migrate among sub-populations with a given frequency. This model helps to explore different parts of the search space through different sub-populations and maintaining diversity within a subpopulation thanks to the introduction of immigrants. The island model for GP was empirically studied [3,16] and the authors showed that it helped to improve GP performance by improving the diversity of GP population.

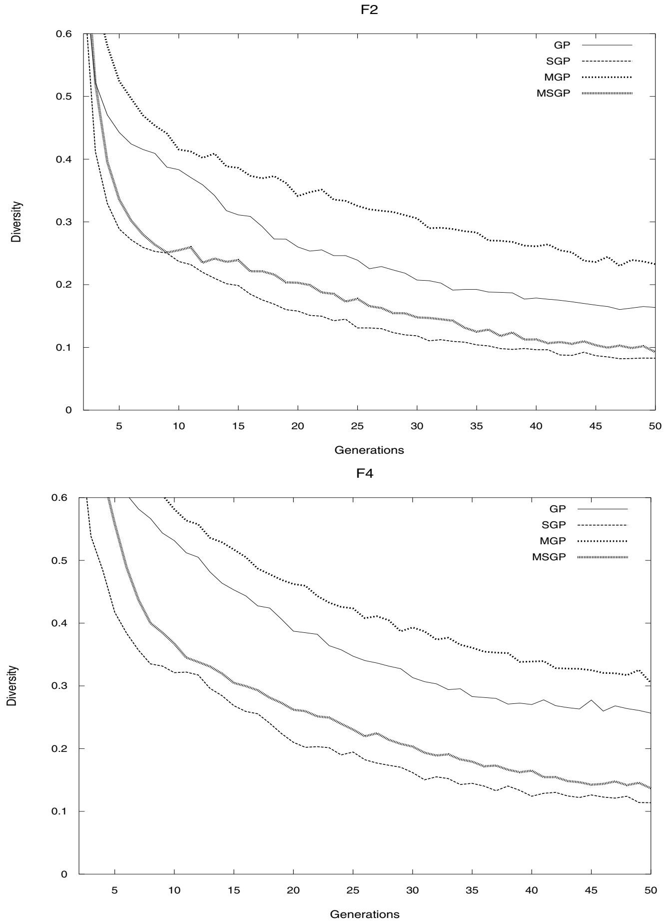

Fig. 2. The diversity of SSC in multi-population GP compared to other methods

In order to implement multi-population GP, some parameters need to be tuned. These parameters include the number of subpopulations, the number of the individuals that migrate among subpopulations and the frequency for migrating individuals. In this paper, these values were calibrated by experiments and the values for the good performance of multi-population GP are as follows: 10 subpopulations, 20 individuals in each subpopulation were migrated to others, and the frequency for migrating individuals is 2.

We implemented SSC in a multi-population GP with the above configuration and the resultant system is labelled MSGP in the following results. Figure 2 presents the comparative population diversity of four tested systems, GP, SGP, MGP (GP with muti-population) and MSGP (GP with multi-population and SSC) [2]. It can be observed from Figure 2 that MGP maintained higher diversity

[2] Again, we only show the results for $F2$ and $F4$ due to space limits.

than standard GP. This is consistent with the previous results in [16]. What is more important is that by combining SSC with MGP (MSGP), we could maintain the higher diversity in the population compared to SGP. Although the improvement of diversity of MSGP compared to SGP was not remarkably significant, this enhancement led to the better performance of SSC as shown in the following.

To compare the performances of all systems in the experiments, we use two classic metrics namely mean of the best fitness and the number of successful runs. These results are shown in Table 3 and Table 4.

Table 3. Number of successful runs

Methods	F1	F2	F3	F4	F5	F6	F7	F8
GP	46	12	9	1	4	36	11	0
MGP	58	34	14	7	7	62	18	0
SGP	65	28	19	8	15	48	45	0
MSGP	68	35	29	13	19	65	54	0

Table 4. Mean best fitness of four methods. Note that the values are scaled by 10^2.

Methods	F1	F2	F3	F4	F5	F6	F7	F8
GP	1.30	1.56	1.61	2.03	2.64	1.85	3.10	13.7
MGP	0.96	1.27	1.35	1.69	1.99	1.35	1.73	10.6
SGP	0.81	0.99	1.01	1.26	1.43	1.02	1.37	9.80
MSGP	0.62	0.85	0.90	1.01	1.21	0.93	0.95	9.21

It can be seen from these tables that implementing SSC in a multi-population GP helped to further improve the performance of SSC. Obviously, the number of successful runs of MSGP was always greater than those of SGP and the quality of solutions found by MSGP was also better than ones of SGP. We also statistically tested the significance of the results in Table 4 using a Wilcoxon signed rank test with a confidence level of 95%. The statistical results show that all the improvements of MSGP, MGP and SGP over standard GP are significant. However, MSGP performance is not significantly better than SGP and MGP though it is the best method among four tested systems in terms of the number of runs which solved the problem in each instance.

5.3 Tuning SSC Parameters for Better Diversity

The previous section showed that MSGP helped to improve the performance of GP compared to SGP, nevertheless, the margin of the improvement, in terms of mean best fitness, was not remarkable. The reason may lie in the fact that MSGP had only slightly higher population diversity than SGP and still rather lower than standard GP. Therefore, we hypothesized that by reducing the value of Max_Trial in SSC we can further increase its diversity and this potentially lead to further improvements of MSGP performance. We tested this hypothesis by conducting an experiment with smaller values of Max_Trial, namely 6, 8, and 10. MSGP with these configurations are denoted as MSGP6, MSGP8 and MSGP10 respectively. We measured the performance of these MSGP configurations and compared them with other systems in the previous subsection. The results are shown in Table 5 and Table 6 [3].

Table 5. Number of successful runs of three new configurations

Methods	F1	F2	F3	F4	F5	F6	F7	F8
GP	46	12	9	1	4	36	11	0
MGP	58	34	14	7	7	62	18	0
SGP	65	28	19	8	15	48	45	0
MSGP	68	35	29	13	19	65	48	0
MSGP6	74	39	26	17	20	68	50	0
MSGP8	68	43	36	23	15	68	54	0
MSGP10	68	52	32	17	21	58	51	0

Table 6. Mean best fitness of three new configurations. Note that the values are scaled by 10^2.

Methods	F1	F2	F3	F4	F5	F6	F7	F8
GP	1.30	1.56	1.61	2.03	2.64	1.85	3.10	13.7
MGP	0.96	1.27	1.35	1.69	1.99	1.35	1.73	10.6
SGP	0.81	0.99	1.01	1.26	1.43	1.02	1.37	9.80
MSGP	0.62	0.85	0.90	1.01	1.21	0.93	0.95	9.21
MSGP6	**0.26**	**0.52**	**0.68**	0.90	**1.04**	**0.60**	**0.86**	9.17
MSGP8	**0.27**	**0.54**	**0.55**	**0.86**	**1.12**	**0.54**	**0.65**	9.20
MSGP10	**0.26**	**0.42**	**0.67**	**0.87**	**0.96**	**0.68**	1.01	**9.04**

It can be seen from these tables that the new configurations of MSGP helped to improve the performance of GP to a further extent. The number of successful runs of MSGP6, MSGP8 and MSGP10 was often greater than MSGP and the mean best fitness was usually far smaller than that of MSGP. We also statistically

[3] We also measured the performance of SGP6, SGP8, SGP10 and their performances are inferior to SGP12. These results are consistent with the results in [10].

tested the significance of improvement of the results in Table 6 using a Wilcoxon signed rank test with a confidence level of 95%. In this table, if a result of MSGP6, MSGP8 and MSGP10 is significantly better than the result of SGP, it is printed in bold face. The results of statistical tests show that in most cases, the improvement of MSGP6, MSGP8 and MSGP10 over SGP is statistically significant.

6 Conclusions and Future Work

In this paper, we investigated the diversity property of Semantic Similarity based Crossover (SSC). Since SSC aims to improve locality, it may lead to the loss of diversity and the experimental results presented in the paper confirmed this. We then proposed an approach to maintain diversity for SSC by combining it with multi-population GP (MSGP). We tested the new method on eight symbolic regression problems and the results showed that multi-population GP with SSC has higher diversity than standard GP with SSC (SGP). This led to the superior performance of MSGP to SGP. However, the improvement was not significant. Then, we tuned the parameter of SSC to achieve higher diversity and resulted in better performance of MSGP.

There are a number of areas for future work which arise from this paper. First, we want to test more values of Max_Trail of SSC to figure out the suitable values for a class of problems. Second, we would like to combine SSC with other methods for promoting diversity such as fitness sharing [7] to see if it gains further improvement. Last but not least, we aim to investigate the impact of this method on some more difficult problems such as text summarization, time series prediction, etc. For these problems, we predict that maintaining high diversity along with locality is critical for GP performance.

Acknowledgments. This work was funded by The Vietnam National Foundation for Science and Technology Development (NAFOSTED), under grant number 102.01-2011.08. Michael O'Neill is grateful for the financial support of Science Foundation Ireland under grant numbers 08/IN.1/I1868 and 08/SRC/FM1389.

References

1. Beadle, L., Johnson, C.: Semantically driven crossover in genetic programming. In: Proceedings of the IEEE World Congress on Computational Intelligence, pp. 111–116. IEEE Press (2008)
2. Burke, E.K., Gustafson, S., Kendall, G.: Diversity in genetic programming: An analysis of measures and correlation with fitness. IEEE Transactions on Evolutionary Computation 8(1), 47–62 (2004)
3. Fernandez, F., Tomassini, M., Vanneschi, L.: An empirical study of multipopulation genetic programming. Genetic Programming and Evolvable Machines 4(1), 21–51 (2003)
4. Gustafson, S., Burke, E.K., Kendall, G.: Sampling of Unique Structures and Behaviours in Genetic Programming. In: Keijzer, M., O'Reilly, U.-M., Lucas, S., Costa, E., Soule, T. (eds.) EuroGP 2004. LNCS, vol. 3003, pp. 279–288. Springer, Heidelberg (2004)

5. Hoai, N.X., McKay, R.I., Essam, D.: Representation and structural difficulty in genetic programming. IEEE Transaction on Evolutionary Computation 10(2), 157–166 (2006)
6. Koza, J.: Genetic Programming: On the Programming of Computers by Natural Selection. MIT Press, MA (1992)
7. Langdon, W.B.: Genetic Programming and Data Structures: Genetic Programming + Data Structure = Automatic Programming! Kluwer Academic, Boston (1998)
8. Looks, M.: On the behavioral diversity of random programs. In: GECCO 2007: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, July 7-11, vol. 2, pp. 1636–1642. ACM Press (2007)
9. McKay, B.: An investigation of fitness sharing in genetic programming. The Australian Journal of Intelligent Information Processing Systems 7(1/2), 43–51 (2001)
10. Nguyen, Q.U., Nguyen, X.H., O'Neill, M., McKay, R.I., Galvan-Lopez, E.: Semantically-based crossover in genetic programming: application to real-valued symbolic regression. Genetic Programming and Evolvable Machines, 91–119 (2011)
11. O'Reilly, U.M., Oppacher, F.: Program Search with a Hierarchical Variable Length Representation: Genetic Programming, Simulated Annealing and Hill Climbing. In: Davidor, Y., Männer, R., Schwefel, H.-P. (eds.) PPSN 1994. LNCS, vol. 866, pp. 397–406. Springer, Heidelberg (1994)
12. Poli, R., Langdon, W.B.: On the search properties of different crossover operators in genetic programming. In: Genetic Programming: Proceedings of the Third Annual Conference, pp. 293–301. Morgan Kaufmann (1998)
13. Rosca, J.P.: Entropy-driven adaptive representation. In: Proceedings of the Workshop on Genetic Programming: From Theory to Real-World Applications, July 9, pp. 23–32 (1995)
14. Rothlauf, F.: Representations for Genetic and Evolutionary Algorithms, 2nd edn. Springer (2006)
15. Tenese, R.: Parallel genetic algorithms for a hypercube. In: Greenstette, J.J. (ed.) Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms, pp. 177–183. Lawrence Erlbaum
16. Tomassini, M., Vanneschi, L., Fernández, F., Galeano, G.: A Study of Diversity in Multipopulation Genetic Programming. In: Liardet, P., Collet, P., Fonlupt, C., Lutton, E., Schoenauer, M. (eds.) EA 2003. LNCS, vol. 2936, pp. 243–255. Springer, Heidelberg (2004)
17. Nguyen, Q.U., Nguyen, X.H., O'Neill, M.: Semantic Aware Crossover for Genetic Programming: The Case for Real-Valued Function Regression. In: Vanneschi, L., Gustafson, S., Moraglio, A., De Falco, I., Ebner, M. (eds.) EuroGP 2009. LNCS, vol. 5481, pp. 292–302. Springer, Heidelberg (2009)