Diversity Analysis in Cellular and Multipopulation Genetic Programming

G. Folino, C. Pizzuti, G. Spezzano
ICAR-CNR
Via P. Bucci 41C, I-87030 Rende (CS) - Italy
{folino,pizzuti,spezzano}@icar.cnr.it

L. Vanneschi, M. Tomassini
Computer Science Institute
University of Lausanne, Switzerland
{Leonardo.Vanneschi,Marco.Tomassini}@iis.unil.ch

Abstract- This paper presents a study that evaluates the influence of the parallel genetic programming (GP) models in maintaining diversity in a population. The parallel models used are the cellular and the multipopulation one. Several measures of diversity are considered to gain a deeper understanding of the conditions under which the evolution of both models is successful. Three standard test problems are used to illustrate the different diversity measures and analyze their correlation with performance. Results show that diversity is not necessarily synonym of good convergence.

1 Introduction

One of the major shortcomings of standard evolutionary algorithms (EAs) is their inability to maintain diversity in the population. This lack of diversity can lead to a number of problems such as converging to a *non-global* optima or not being able to react to changes in the environment. The lack of diversity is especially evident when dealing with multimodal problems or when using evolutionary algorithms to solve *dynamic* problems.

In Genetic Programming (GP), the process converges when the elements of the phenotypic pool are identical, or nearly so, in spite of the fact that the genotypic pool might still present some syntactical diversity. When this occurs, the crossover operator ceases to produce new individuals, and the algorithm allocates all of its trials in a very small subset of the program space. Unfortunately, this often occurs before the true optimum has been found; this behavior is called premature convergence. The mutation operator provides a mechanism for reintroducing lost diversity, but it does it at the cost of slowing down the learning process.

Both genotypic and phenotypic diversity play a role in GP and the two are not necessarily correlated in a straightforward manner. In particular, the phenomenon of "bloat", consisting in the tendency of code to grow in size over generation is well-known, and it often gives rise to large nonfunctional tree portions that could increase genotypic diversity but not the phenotypic one, nor the capability of the system to produce better solutions.

Many approaches have been proposed for diversity maintenance within a population. Among them fitness sharing [5, 6] works with the idea of similarity between individuals, thus requiring a consistent distance measure in the population, and multi-objective optimization [4], where fitness, size and diversity are the objectives to be satisfied.

However, diversity in parallel GP models has been little studied. One such study can be found in [12] where a systematic experimental investigation of how a multipopulation GP model helps in maintaining the phenotypic and genotypic diversity is presented.

In this work, we extend the previous analysis on multipopulation GP model with additional experiments, new analysis, and new measures. We also study the diversity in the cellular GP model. An interesting aspect of the parallel approaches is that diversity in both models is maintained without any particular algorithm beyond the simple communication among island or the diffusion principle of cellular systems.

The paper is organized as follows. Section 2 presents a classification of the parallel GP models and provides some information on their parallel implementation on distributed-memory computers. Section 3 presents the different diversity measures used for both models and those only for the cellular model. Section 4 describes the benchmark problems used and the experimental results obtained. Finally, section 5 provides the conclusions and discusses future work.

2 Parallel Genetic Programming Models

Several approaches for speeding-up the GP implementations have been recently proposed. They are directed towards two orthogonal directions: speeding-up by minimizing the computational effort of GP, and improving the numerical performance of the algorithm itself by using population structuring principles. A classification of the approaches for parallelizing GP includes three main models [16]: the *global* model, the *coarse-grained* (*island*) model [13] and the *fine-grained* (also called *cellular* or grid) model [14]. In the following we consider only the island and the cellular models.

The island model divides a population $\mathcal{P}$ of M individuals into N subpopulations $P_1, \ldots, P_N$, called *demes*, of M/N individuals. A standard genetic programming algo-

0-7803-7804-0 /03/$17.00 © 2003 IEEE

rithm works on each deme and is responsible for initializing, evaluating and evolving its own subpopulation. Subpopulations are interconnected according to different *communication topologies* and can exchange information periodically by *migrating* individuals from one subpopulation to another. The number of individuals to migrate (*migration rate*), the number of generations after which migration should occur (*frequency*), the migration topology and the number of subpopulations are all parameters of the method that have to be set.

In [7] a systematic experimental investigation of the behavior of semi-isolated populations in GP is presented. The model implemented consists of *demes* that evolve independently with the same parameters as panmictic GP, except for the migration of the best p individuals every t iterations from a given island to a randomly chosen one different from itself, where they replace the worst p individuals. All the experiments showed that p= 10% of the population size, and t=10 are suitable values, and thus they are used in this work. Sending and receiving blocks of individuals is done synchronously. Empirically, it has been observed in [7], as well as in other studies, that distributing the individuals among several loosely connected islands has the advantage to go beyond the obvious time savings when the system runs on multiple machines, since often multiple population also lead to statistically significantly better solution quality.

In the cellular model each individual is associated with a spatial location on a low-dimensional grid. The population is considered as a system of active individuals that interact only with their direct neighbors. Different neighborhoods can be defined for the cells. The most common neighborhoods in the two-dimensional case are the 5-neighbor (*von Neumann neighborhood*) consisting of the cell itself plus the North, South, East, West neighbors and 9-neighbor (*Moore neighborhood*) consisting of the same neighbors augmented with the diagonal neighbors. Fitness evaluation is done simultaneously for all the individuals and selection, reproduction and mating take place locally within the neighborhood. Information slowly diffuses across the grid giving rise to the formation of semi-isolated niches of individuals having similar characteristics. The choice of the individual to mate with the central individual and the replacement of the latter with one of the offspring can be done in several ways.

A scalable implementation of the cellular GP model, called CAGE, is described in [9].

CAGE is fully distributed with no need of any global control structure and it is naturally suited for implementation on parallel computers. It introduces fundamental changes in the way GP works. In the model, the individuals of the population are located on a specific position in a toroidal 2-D grid and the selection and mating operations are performed, cell by cell, only among the individual assigned to a cell and its neighbors. Three replacement policies have been implemented: direct (the best of the offspring always replace the current individual), greedy (the replacement occurs only if offspring is fitter) and probabilistic(the replacement happens according to difference of the fitness between parent and offspring). Experimental results on a variety of benchmark problems have substantiated the validity of the cellular model over both the island model and panmictic GP model. In [8] it is showed that CAGE can reduce the bloat phenomenon if used for classification problems

3 Diversity Measures

Surveys of diversity measures in panmictic GP have been presented in [1, 2]. The diversity measures that we use in this paper are based on the concepts of *entropy* and *variance*. Both these concepts are used to measure the phenotypic (i.e. based on fitness) and genotypic (i.e. based on the syntactical structure of individuals) diversity of populations. Besides, we use another measure that takes into account the spatial structure of the population, denoted as the *frequency of transition* introduced in [3], that is meaningful only for the cellular model. Phenotypic diversity is related to the number of different fitness values of the individuals. Here we use the *phenotypic entropy* $H_p(P)$ [15] of a population P as a diversity measure:

$$H_p(P) = -\sum_{j=1}^{N} f_j \log(f_j)$$

where f_j is the fraction n_j/N of individuals in P having fitness j and N is the number of fitness values in P.

Here we use the *entropy* as a genotypic diversity measure. To be able to define structural diversity among trees, it is first useful to define a tree distance measure. A few tree distances have been proposed in the literature. We use Ekárt's and Németh's definition [6]. The distance between two trees T_1 and T_2 is calculated in three steps: (1) T_1 and T_2 are overlapped at the root node and the process is applied recursively starting from the leftmost subtrees. (2) For each pair of nodes at matching positions, the difference of their codes (possibly raised to an exponent) is computed. (3) The differences computed in the previous step are combined in a weighted sum. Formally, the distance of two trees T_1 and T_2 with roots R_1 and R_2 is defined as follows:

$$dist(T_1, T_2) = d(R_1, R_2) + \frac{1}{k}\sum_{i=1}^{m} dist(child_i(R_1), child_i(R_2))$$

where: $d(R_1, R_2) = (|c(R_1) - c(R_2)|)^z$, $child_i(Y)$ is the i^{th} of the m possible children of a generic node Y, if $i \leq m$, or the empty tree otherwise, and c evaluated on

the root of an empty tree is 0. Constant k is used to give different weights to nodes belonging to different levels and z is a constant usually chosen in such a way that $z \in \mathcal{N}$.

The genotypic entropy $H_g(P)$ of a population P is defined as follows:

$$H_g(P) = -\sum_{j=1}^{N} g_j \log(g_j)$$

where, g_j is the fraction of individuals having a given distance from the origin, which has arbitrarily been chosen as the empty tree.

The variance of a population P is defined as follows:

$$V(P) = \frac{1}{n}\sum_{i=1}^{n}(f_i - \overline{f})^2$$

If we are considering phenotypic variance, $\overline{f}$ is the average fitness of the individuals in P, f_i is the fitness of the i^{th} individual in P and n is the total number of individuals in P. To define genotypic variance, we use the notion of tree distance. In this case, $\overline{f}$ is the average of all the individual distances from the origin tree, f_i is the distance of the i^{th} individual in P from the origin tree and n is the total number of individuals in P. Then the standard deviation is the square root of the variance.

The frequency of transition of a population P regards only the cellular model and it is defined as the number of borders between homogeneous blocks of cells (individuals) having the same genotype (phenotype), divided by the number of distinct couples of adjacent cells, i. e. the probability that two adjacent cells belong to different blocks:

$$ft(P) = \frac{\sum_{i=1}^{n}\sum_{j\in N(i)}[f_i \neq f_j]}{\sum_{i=1}^{n} \| N(i) \|}$$

where $[f_i \neq f_j]$ is 1 if $f_i \neq f_j$, otherwise is 0, and $N(i)$ is the neighborhood of individual i, where f_i has the same meaning of the f_i introduced for the variance.

4 Experiments

In this analysis three well know problems, the *Even 4-Parity problem*, the *Symbolic Regression problem* and the *Artificial Ant on the Santa Fe trail problem* ([10, 11]), are considered. The parity problem takes an input of 4 Boolean variables and it returns TRUE only if an even number of variables is true. The even 4-parity fitness is the number of wrong guesses for the 2^4 combinations of 4-bit length strings. Thus a perfect individual has fitness 0, while the worst individual has fitness 16.

The Symbolic Regression problem consists in searching a program which matches a given equation, in our case the polynomial equation $f(x) = x^4 + x^3 + x^2 + x$. The input set is composed of the values 0 to 999 (1000 fitness cases), and the set of functions used for GP individuals is F={*,//,+,-}, where // is like / but returns 0 instead of *error* when the divisor is equal to 0. The fitness is the sum of the square errors at each test point.

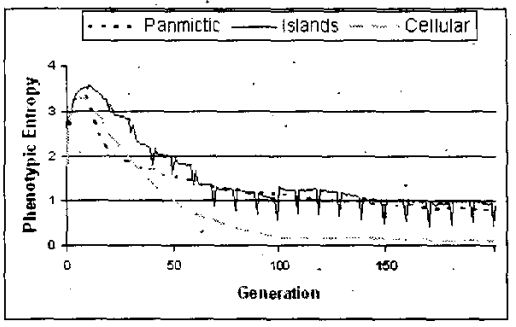
In the Artificial Ant Problem on the Santa Fe Trail the goal is to find the best strategy for picking up food pellets along a trail on a 32×32 toroidal grid. We use the same set of functions and terminals as in [10]. The fitness function is the number of pellets missed by the ant during his path.

In all the experiments we use the same set of GP parameters: generational GP, crossover rate: 95%, mutation rate: 0.1%, tournament selection of size: 10, ramped half and half initialization, maximum depth of individuals for the creation phase: 6, maximum depth of individuals for crossover: 17, elitism (i.e. survival of the best individual into the newly generated population for panmictic populations. The same was done for each subpopulation in the distributed case and in the cellular case). The size of the population was set to 500 for the even 4-parity problem, 250 for the regression one, and to 1000 for the ant problem. We next present the results of our simulations. The curves represent average values over 100 independent GP runs. Note that these population sizes have been found suitable in [12], where the sizing of the islands has been thoroughly studied empirically.

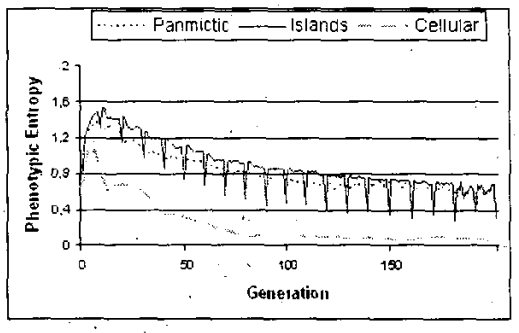
4.1 Phenotypic Diversity Behavior

We first discuss the phenotypic behavior. Figure 1 shows the phenotypic entropy for the three test problems. Entropy [15] represents the amount of disorder of the population, thus low entropy means low diversity. However, since the phenotypic measure compares the number of different fitness values, it could be interpreted as the number of groups having the same fitness value. Thus high entropy could be considered as the presence in the population of a high number of small groups of individuals, each group having the same fitness value, while low entropy would mean a low number of large groups of individuals.

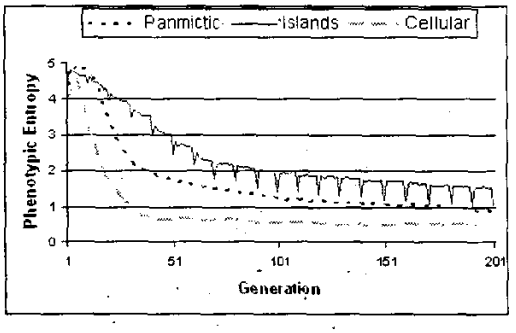
In this perspective, the fact that the cellular model has always a lower phenotypic entropy with respect to both the island and the panmictic models, as figure 1 points out, can be interpreted as the presence in the population of a low number of groups each containing many individuals having the same fitness value. This is confirmed by the low phenotypic standard deviation of the cellular model shown in figure 3 and by the frequency of transition, shown in figure 2 which counts the number of individuals having the same fitness value with their neighborhoods. The jigged be-



(a)

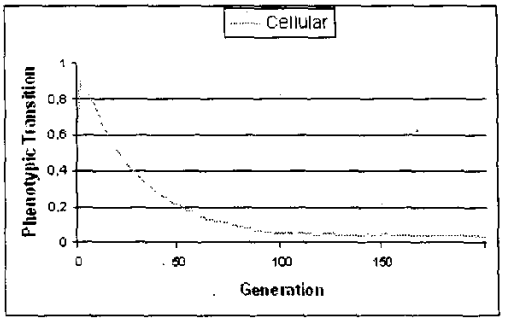


(b)

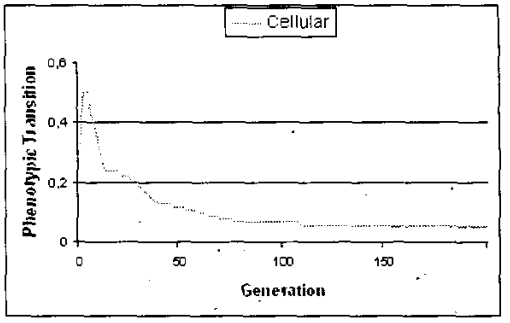


(c)

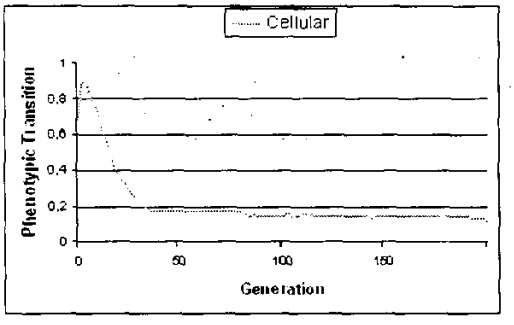
Figure 1: Phenotypic entropy for the artificial ant problem (a), the even 4 parity problem (b), the symbolic regression problem (c).



(a)



(b)



(c)

Figure 2: Phenotypic transition function for the artificial ant problem (a), the even 4 parity problem (b), the symbolic regression problem (c).

havior of the curves referring to the subpopulations in the island model is due to the sudden change in diversity when the new individuals enter the population at fixed generation numbers.

Low phenotypic diversity in the cellular model can be explained by the diffusion of the information across the grid that induces groups of individuals having similar characteristics. It is worth to point out that low phenotypic entropy does not imply worst convergence of the method. In fact, though the figure shows the experiments for 200 generations, actually the same near optimal fitness value was found at approximatively generations 80, 150, 250 by using the cellular, island and panmictic models respectively for the ant problem, at generations 200, 150, 200 for the par-

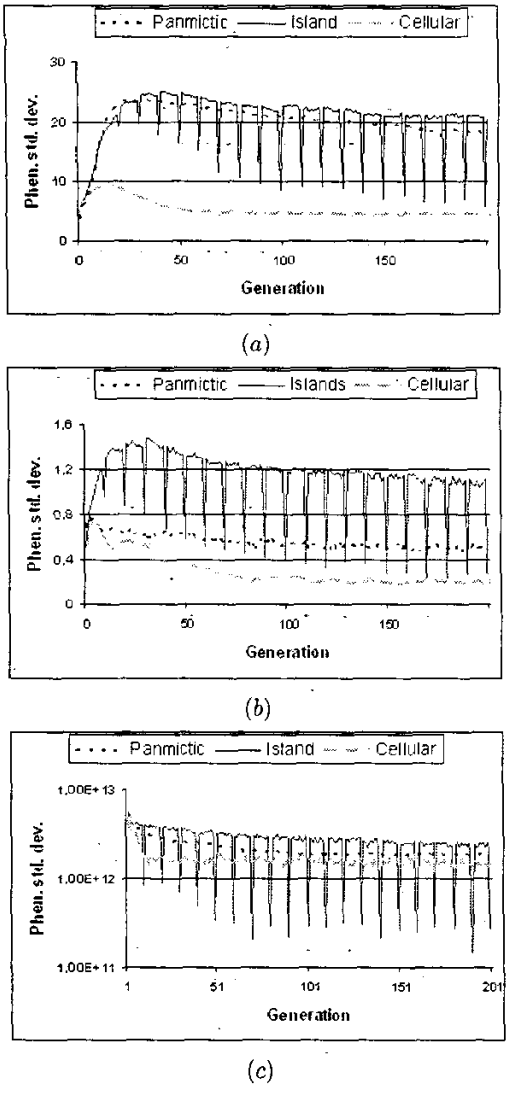


Figure 3: Phenotypic standard deviation for the artificial ant problem (a), the even 4 parity problem (b), the symbolic regression problem (c).

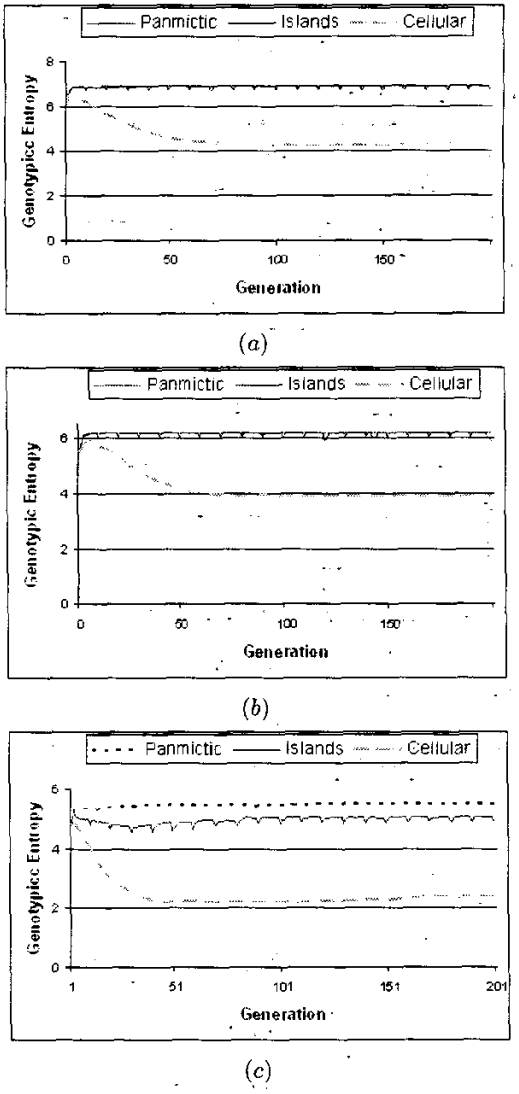


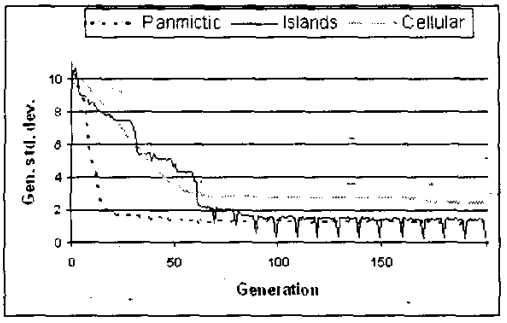
Figure 4: Genotypic entropy for the artificial ant problems (a), the even 4 parity problem (b), the symbolic regression problem (c).

ity problem, and at generations 10, 20, 50 for the symbolic regression problem.

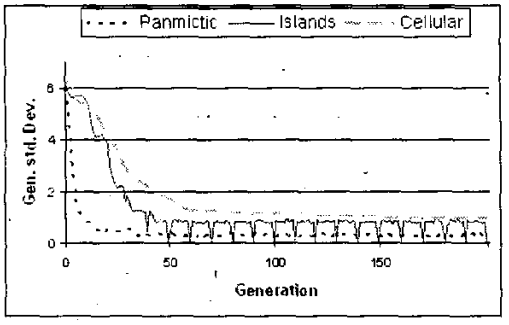
4.2 Genotypic Diversity Behavior

As in the case of phenotypic entropy, figure 4 shows that genotypic entropy is lower for the cellular model with respect to both the island and the panmictic ones, while genotypic entropy for the island model is almost the same of the panmictic model for ant and parity problems, and lower for symbolic regression.

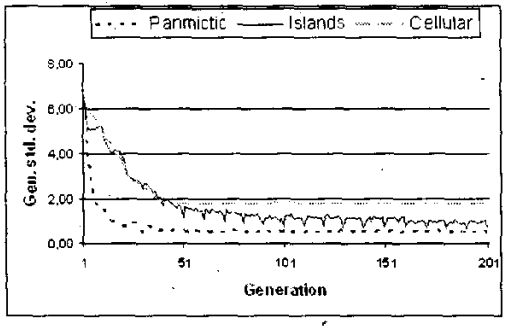
This behavior suggests that, as regards the cellular model, we have few groups of individuals having the same distance from the empty tree, each group being composed by many trees. However, as figure 5 suggests, trees in



(a)

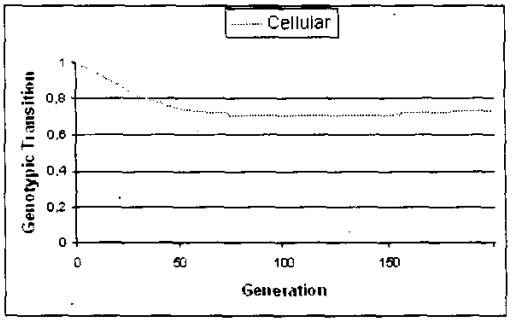


(b)

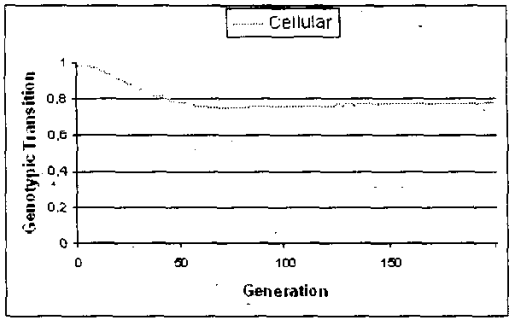


(c)

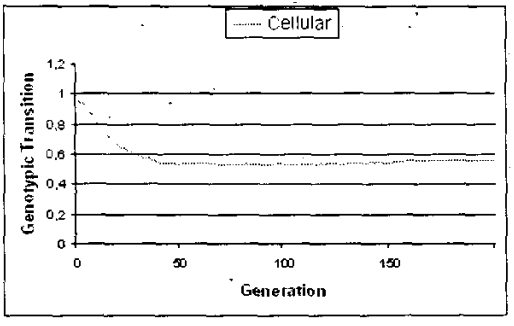
Figure 5: Genotypic standard deviation for the artificial ant problem (a), the even 4 parity problem (b), the symbolic regression problem (c).



(a)



(b)



(c)

Figure 6: Genotypic transition function for the artificial ant problem (a), the even 4 parity problem (b), the symbolic regression problem (c).

the population are very dissimilar among them because the standard deviation is high, thus the distance of each tree from the origin tree is substantially different from the average distance of all the trees from the origin tree. High diversity in the tree structure is confirmed by the genotypic transition function which, as figure 6 shows, maintains values near the optimum, that is 1, during the evolutionary process.

Having high genotypic diversity and low phenotypic diversity in the cellular model could seem contradictory. However this apparent conflicting behavior can be explained by the fact that though the trees are structurally different, this does not imply that their fitness must be different too. In the cellular case it means that almost all the trees

have good fitness values and this explains the better convergence of the cellular model.

5 Conclusions

The paper analyzed the phenotypic and genotypic diversity of a population in the island and cellular parallel genetic programming models with respect to the panmictic one. The experiments showed that, as regard the phenotypic diversity, the cellular model presents a lower value than the panmictic one, while the island model presents an higher value than the panmictic model. In any case the convergence of cellular and island models is faster. Thus a diversity measure based on fitness of individuals does not seem to give enough information to infer that higher phenotypic diversity implies better performance. Genotypic diversity is again lower in the cellular model and almost the same for the island and panmictic models. However, in such a case, the genotypic standard deviation is higher for both the cellular and island models. This implies that the trees are much more dissimilar and this dissimilarity could explain the faster convergence of the parallel models. The study thus pointed out that diversity does not necessarily means that the system is capable to obtain fitter solutions. Future work aims at considering new diversity measures, and at a thorough investigation and further experiments on more problems to find a tight correlation between diversity and performance.

Acknowledgments

The authors of ICAR-CNR in this work have been partially supported by Project "FIRB GRID.IT" funded by MIUR.

Bibliography

[1] Edmund Burke, Steven Gustafson, and Graham Kendall. A survey and analysis of diversity measures in genetic programming. In *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 716–723, New York, 9-13 July 2002. Morgan Kaufmann Publishers.

[2] Edmund Burke, Steven Gustafson, Graham Kendall, and Natalio Krasnogor. Advanced population diversity measures in genetic programming. In *Parallel Problem Solving from Nature - PPSN VII*, number 2439 in Lecture Notes in Computer Science, LNCS, page 341 ff., Granada, Spain, 7-11 September 2002. Springer-Verlag.

[3] Mathieu Capcarrère, Marco Tomassini, Andrea Tettamanzi, and Moshe Sipper. A statistical study of a class of cellular evolutionary algorithms. *Evolutionary Computation*, 7(3):255–274, 1999.

[4] Edwin D. de Jong, Richard A. Watson, and Jordan B. Pollack. Reducing bloat and promoting diversity using multi-objective methods. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 11–18, San Francisco, California, USA, 7-11 July 2001. Morgan Kaufmann.

[5] Kalyanmoy Deb and David E. Goldberg. An investigation of niche and species formation in genetic function optimization. In J. David Schaffer, editor, *Proceedings of the 3rd International Conference on Genetic Algorithms*, pages 42–50, George Mason University, June 1989. Morgan Kaufmann.

[6] Anikó Ekárt and Sandor Z. Németh. Maintaining the diversity of genetic programs. *Lecture Notes in Computer Science, EuroGP 2002*, 2278:162–171, 2002.

[7] Francisco Fernandez, Marco Tomassini, and Leonardo Vanneschi. An empirical study of multipopulation genetic programming. *Genetic Programming and Evolvable Machines*, 4(1):21–51, March 2003.

[8] G. Folino, C. Pizzuti, and G. Spezzano. A cellular genetic programming approach to classification. In *Genetic and evolutionary conference, GECCO99*, volume 2, pages 1015–1020. Morgan Kaufmann, San Francisco, CA, 1999.

[9] G. Folino, C. Pizzuti, and G. Spezzano. A scalable cellular implementation of parallel genetic programming. *IEEE Transactions on Evolutionary Computation*, 7:37–53, February 2003.

[10] J. R. Koza. *Genetic Programming*. The MIT Press, Cambridge, Massachusetts, 1992.

[11] W. B. Langdon and R. Poli. *Foundations of Genetic Programming*. Springer, Berlin, 2002.

[12] F. Fernández M. Tomassini, L. Vanneschi and G. Galeano. Diversity in multipopulation genetic programming. In *Genetic and Evolutionary Computation Conference, GECCO 2003*, volume 2610 of *LNCS*, Springer-Verlag, 2003.

[13] W. N. Martin, J. Lienig, and J. P. Cohoon. Island (migration) models: evolutionary algorithms based on punctuated equilibria. In Thomas Bäck, David B. Fogel, and Zbigniew Michalewicz, editors, *Handbook of Evolutionary Computation*, pages C6.3:1–16. Institute of Physics Publishing and Oxford University Press, Bristol, New York, 1997.

[14] C. C. Pettey. Diffusion (cellular) models. In Thomas Bäck, David B. Fogel, and Zbigniew Michalewicz, editors, *Handbook of Evolutionary Computation*, pages C6.4:1–6. Institute of Physics Publishing and Oxford University Press, Bristol, New York, 1997.

[15] Justinian P. Rosca. Entropy-driven adaptive representation. In Justinian P. Rosca, editor, *Proceedings of the Workshop on Genetic Programming: From Theory to Real-World Applications*, pages 23–32, Tahoe City, California, USA, 1995.

[16] M. Tomassini. Parallel and distributed evolutionary algorithms: A review. In P. Neittaanmki K. Miettinen, M. Mkel and J. Periaux, editors, *Evolutionary Algorithms in Engineering and Computer Science*, J. Wiley and Sons, Chichester, 1999.