Multi-cellular development: is there scalability and robustness to gain?

Daniel Roggen and Diego Federici*

Autonomous Systems Laboratory, EPFL, Lausanne, Switzerland
*Norwegian University of Science and Technology, Trondheim, Norway
http://asl.epfl.ch *http://www.idi.ntnu.no/∼federici
daniel.roggen@epfl.ch *federici@idi.ntnu.no

Abstract. Evolving large phenotypes remains nowadays a problem due to the combinatorial explosion of the search space. Seeking better scalability and inspired by the development of biological systems several indirect genetic encodings have been proposed. Here two different developmental mechanisms are compared. The first, developed for hardware implementations, relies on simple mechanisms inspired upon gene regulation and cell differentiation. The second, inspired by Cellular Automata, is an Artificial Embryogeny system based on cell-chemistry. This paper analyses the scalability and robustness to phenotypic faults of these two systems, with a direct encoding strategy used for comparison.
Results show that, while for direct encoding scalability is limited by the size of the search space, developmental systems performance appears to be related to the amount of regularity that they can extract from the phenotype. Finally the importance of comparing different genetic encodings is stressed, in particular to evaluate which key characteristics are necessary for better scalability or fault-tolerance. The lack of standard tests or benchmarks is highlighted and some characterisations are proposed.

1 Introduction

The evolution of large phenotypes is one of the most serious problems in the field of evolutionary computation. With each characteristic of the phenotype encoded by a single gene, the increase of the phenotypic size imposes for direct encoding strategies a combinatorial explosion of the search space.

On the other hand, biological systems develop into mature organisms with a complex process of embryogeny. Embryogeny is mediated by the interaction of DNA, RNA and proteins to produce the cell regulatory system. This sort of interaction does not permit a one to one map from gene to phenotypic trait (phene), since each gene influences several aspects of the phenotype (pleiotropy).

Motivated by the development of biological systems, several authors have proposed indirected encoding schemes. With indirect encoding, each phenotype is developed by a process in which genes are reused several times.

In this case, development is de facto a decompression of the genotype. But since compression is generally higher for regular targets, a serious question is how much these methods will prove viable for the evolution of arbitrarily complex phenotypes. For example, the correlation between genotype and phenotype space may decrease as the complexity of the target increases [1].

In other words, when looking at system evolvability, it appears that there is a tradeoff between the combinatorial gain achieved by searching in a restricted genotypic space and hindrances of a more complex fitness landscape caused by gene reuse.

Additionally, the restriction on the search space implies that a part of the solution space becomes unreachable, and some targets (such as those of high regularity) might be more viable than others.

These considerations imply that in the analysis of such systems, performance benchmarks play a fundamental role. Still, there is little agreement on a set of evolutionary targets that can be used for assessing their quality. On the contrary, it appears that the tasks are often selected ad hoc to highlight the strengths of a particular model.

In this paper we want to compare two different developmental models, the first used in the POEtic circuit [2, 3], the latter an Artificial Embryogeny system [4] based on cell chemistry [5, 6]. The comparison is carried out for varying phenotypic sizes, against a direct encoding strategy in a task that should favour the latter.

The intention is to investigate the viability of these two indirect encoding methods without leaving doubts about the generality of the results.

To this end, we have tried to set up a 'worst case scenario' for developmental systems, pushing for results that do not depend on particular features of the targets.

The selected task is the evolution of specific 2D patterns of various complexity (figures 3 and 4) and sizes (from 8x8 to 128x128), with fitness being proportional to the resemblance to the target. In the case of the direct encoding strategy, with a gene representing a single pixel, the fitness landscape is a simple unimodal function. On the contrary, in the case of development, gene reuse may imply a multimodal deceptive fitness landscape. Thus, the comparison of the methods will allow to address the influence of search space and pleiotropy on evolvability.

Development systems also provide internal dynamics which are absent in direct encoding strategies. These dynamics may provide a way to withstand phenotypic injuries. This aspect is explored by comparing the tolerance to faults of both systems with the linear deterioration typical of direct encodings.

The rest of this paper is organized as follows. Section 2 gives an overview of the development systems, section 3 describes the evolutionary task, section 4 presents the results on fault resistance and section 5 concludes.

2 Multi-cellular growth and differentiation mechanisms

2.1 Morphogenetic System (MS)

The morphogenetic system [3] (**MS**) is a developmental model designed for multi-cellular systems and focusing on simplicity and compact hardware implementation, initially developed for the POEtic circuit [2]. It uses signalling and expression mechanisms which are remotely inspired by the gene expression and cell differentiation of living organisms [7], notably by the fact that concentrations of proteins and inter-cellular chemical signalling regulate the functionality of cells. Related works include the use of L-Systems [8] and various cell-based developmental processes [9, 10], and biologically plausible development models [11].

The **MS** assigns a functionality to each cell of the circuit from a set of predefined functionalities. Here functionalities are the colours necessary to draw the patterns. It operates in two phases: a *signalling phase* and an *expression phase*.

The signalling phase uses inter-cellular communication to exchange signals among adjacent cells to implement a diffusion-like process. A signal is a simple numerical value (signal intensity) that a cell owns. Special cells, called *diffusers*, own a signal of maximum intensity and start the diffusion process. Diffusion rules rely on the four neighbours of a cell to generate signal intensities which decrease linearly with the Manhattan distance from the diffuser. They do so by taking the smallest value for which the signal gradient with all the initialized neighbours is -1, 0 or 1. Figure 1 shows an example of the signalling phase in the case of a single type of signal, with two diffusers placed in the cellular circuit.

The expression phase finds the functionality to be expressed in each cell by matching the signal intensities in each cell with a corresponding functionality stored in an expression table.

The genetic code contains the content of the expression table and the position of the diffusers. A genetic algorithm is used for evolution. 16 diffusers and 4 functionalities (colours) are used. The population is composed of 400 individuals, selection is rank selection of the 300 best individuals, the mutation rate is 0.5% per bit, one-point crossover rate is 20% and elitism is used by copying best individuals without modifications into the new generation.

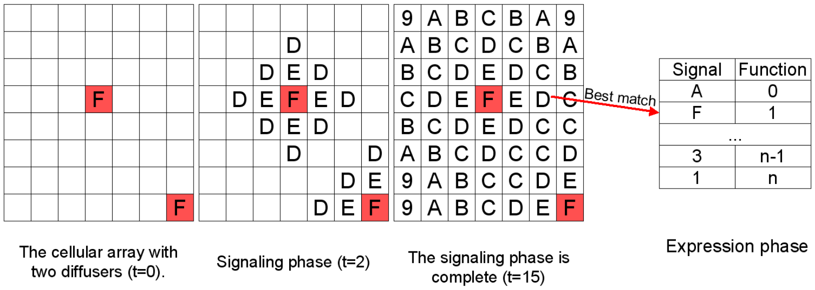
2.2 Embryogeny Model based on Cell Chemistry

Introduction Another way to develop the phenotype is to proceed with a recursive process of rewriting, which starts from a single egg cell to produce the mature organism. Among these Artificial Embryogeny (AE) systems [4], there are two main approaches.

The first is aimed at the evolution of a grammar which is repeatedly applied to the phenotype. Examples include the Matrix Rewriting scheme [12], the Cellular Encoding [10], Edge Encoding [13] and the GenRe system [14].

The second evolves the regulatory system of a cell with its metabolism and its ability to duplicate. Ontogeny results of the emergent interaction of neighboring cells and the chemical concentrations in the environment.

The model used in this paper belongs to this second category, and is an extension of the one presented in [6]. An extensive description on the model can be found in [5].


Best match
The cellular array with two diffusers (t=0).
Signaling phase (t=2)
The signaling phase is complete (t=15)
Expression phase

			F			
						F

			D			
		D	E	D		
	D	E	F	E	D	
		D	E	D		
			D			D
					D	E
				D	E	F

9	A	B	C	B	A	9
A	B	C	D	C	B	A
B	C	D	E	D	C	B
C	D	E	F	E	D	C
B	C	D	E	D	C	C
A	B	C	D	C	C	D
9	A	B	C	C	D	E
9	A	B	C	D	E	F

Signal	Function
A	0
F	1
...	
3	n-1
1	n

Fig. 1. The arrays on the left are snapshots of the signalling phase with one type of signal and two diffusers (gray cells) at the start of the signalling phase, after two time steps, and when the signalling is complete. The number inside the cells indicates the intensity of the signal in hexadecimal. The expression table used in the expression phase is shown on the right. The signal D matches the second entry of the table with signal F (smallest Hamming distance), thus expressing function F_1.

Description Phenotypes are developed starting from a single egg (zygote) placed in the center of a fixed size 2D grid. Morphogenesis proceeds in discrete developmental steps, during which the growth program is executed for each cell, one cell at a time.

Cells are characterized by internal and external variables. Two internal variables (cell type and internal chemical concentration) define the cell state and move with it, while the external one belongs to the environment and follows a simple conservative diffusion law.

At each developmental step, any existing cell can release a chemical to the environment, change its own type, alter its internal metabolism and produce new cells in the cardinal directions North, West, South and East.

The growth program is governed by a feedforward Artificial Neural Networks (Morphers) without hidden layers. Each Morpher is specified by 144 genes (floating values), one for each of the 8 inputs, 16 outputs and bias weights (see figure 2).

Ontogeny is governed by multiple morphers, each one defining an Embryonic Stage which spans one or more developmental steps. Stages are introduced incrementally, those controlling earlier developmental steps being evolved first and only the last one undergoing evolution (please refer to [5] for the full details of the model). This system has the advantage of increasingly adding resolution to promising areas of the search space while excluding the others, also reducing pleiotropy among different developmental steps. New stages are introduced if the performance did not increase for the last 100 generations.

The population is composed of 400 individuals, the 100 best individuals survive and reproduce. Crossover is set at 10%. All the offspring undergo mutation: each of the weights of the evolving Morpher being changed with a .01 probability by adding Gaussian noise with .035 variance.

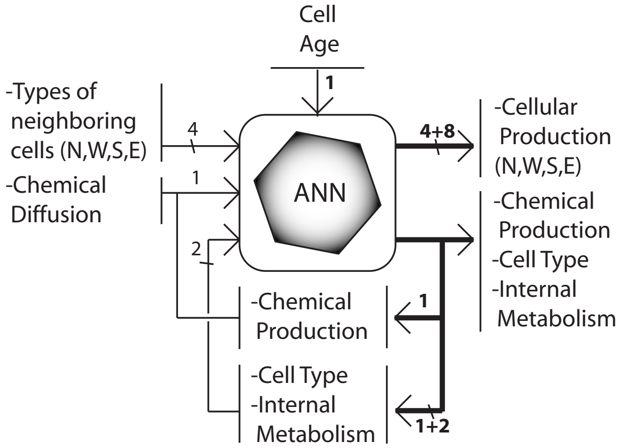


Fig. 2. The growth program (Morpher) input and output lines with their respective sizes. Each line is a floating point value $\in [-1, 1]$. The Morpher is implemented by a feedforward ANN, even though each cell internal variables (cell type and metabolism), implement a direct feedback pathway. Chemical production and diffusion offer a channel for inter-cell communication.

3 Evolution of patterns and scalability

The evolutionary task consists in evolving phenotypes resembling specific 2D patterns of increasing size. This type of problem has been selected in order to simplify the analysis of the results and avoid that the developmental models might benefit from embedded "*tricks*", which will not be applicable in other settings.

The targets are 8x8, 12x12, 16x16, 32x32, 64x64, 96x96 and 128x128 multi-cellular arrays. Each cell can take one of four possible types (colours). Two different typologies of targets are considered. The first one is a more regular 'Norwegian flag' pattern (figure 3) which presents a high degree of symmetry that should be exploitable by developmental systems. The latter, is a very complex pattern generated from a Cellular Automata using Wolfram's rule 90 and starting from random initial conditions (figure 4). Wolfram's rule 90 has been selected because it steadily produces patterns of high complexity, which are supposedly very difficult targets of developmental systems. In the case of direct encoding, the target patterns have equivalent difficulty.

Fitness is proportional to the resemblance of the individual to the target. In order to avoid premature convergence, individuals with rare phenotypic traits (pixels) are rewarded (please refer to [5] for further details).

The experiments were conducted 20 times for each target size. The population is composed of 400 individuals, undergoing elitism selection for 2000 generations. Model specific GA parameters are listed in section 2. For direct coding, the GA parameters are 10% single point crossover, mutation rate of 0.5% per gene, and each gene represents one of the 4 possible colors.

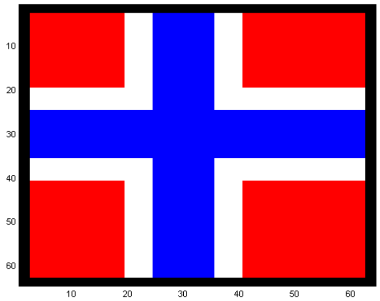


Fig. 3. Norwegian Flag Target (64x64).

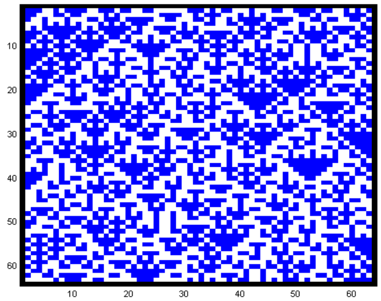


Fig. 4. Target generated using a cellular-automata with rule 90 of Wolfram starting from a random initial line (64x64).

The genotype dimension for the various encodings and target sizes are listed in table 1. The size of the genetic code with the **MS** scales with the logarithm of the size of the array because the number of bits used to encode the position of the diffusers depends on the size of the array. The size of the genetic code using the embryogeny model remains constant because the morpher neural network relies only on the state of immediate neighboring cells to update the state of the current cell and hence needs no information about the size of the array. Size of direct encoding scales with the size of the array.

Encoding	Search space by target dimension						
	8x8	12x12	16x16	32x32	64x64	96x96	128x128
Direct coding	64	144	256	1024	4096	9216	16384
MS	192	224	224	256	288	320	320
AES	144	144	144	144	144	144	144

Table 1. Search space size of the 3 encoding methods presented for each target size. Genes, in the Direct Encoding determine the color at a given position, in the Indirect Encodings regulate the ontogeny of the phenotype. In the **MS** each gene is a bit, in the Embryogeny model is a floating point number in the range $[-1, 1]$.

Scalability is shown in figure 5 for the Norwegian flag and the CA-generated pattern. Direct encoding steadily reaches 100% fitness for arrays up to size 32x32. For larger targets, the explosion of the search space limits the overall performance.

Both development approaches perform similarly for small target sizes where they tend to get high fitness scores. Larger targets show a reduced performance which tends to stabilize around a certain level. In the case of the Norwegian flag, this level is determined by the complexity of the target pattern, which is constant with its size. In the case of the CA-generated target, complexity increases with size and solutions tend to exploit more the spatial frequency of the colours than their exact position.

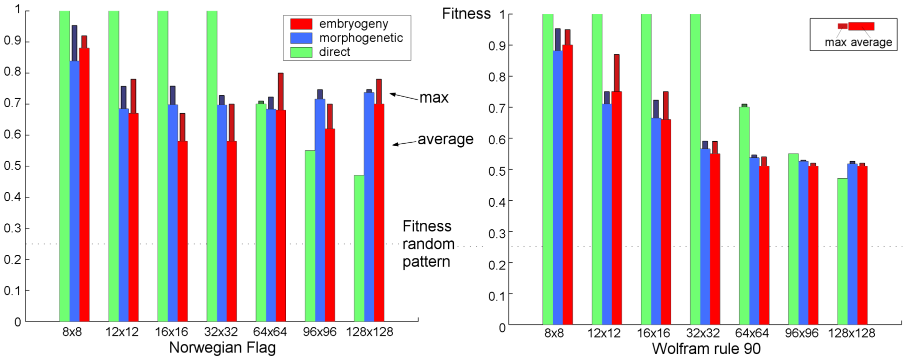


Fig. 5. Scalability for the morphogenetic system (**MS**), embryonic model (**AES**) and direct encoding on evolution of the Norwegian flag and CA-generated pattern.

Figure 6 shows the best evolved solutions for 64x64 targets with the different encoding schemes. Notice that the solutions generated by development systems show artifacts, due to their decoding scheme (diamond-shaped patterns for the **MS** and regular repetitions for the **AES**). On the other hand, direct encoding exhibits Salt and Pepper noise.

4 Robustness

Natural organisms exhibit recovery capabilities, for example in case of injuries. In this section, we explore how these models behave when subjected to faults.

In order to have a meaningful deterioration mechanism for both developmental models, we consider here transient events which damage the state of the cell (e.g. by means of radiation corrupting memory elements). As development continues to operate normally, cell functionality could be recovered. Notice that individuals were not selected for their fault resistance.

In the case of the **MS**, faults modify the chemical content of a cell[1]. For the embryogeny model, faults kill selected cells, while for direct encoding they alter their colour.

Robustness is tested on the best evolved phenotype of the Norwegian flag on the 64x64 array. This pattern and size has been selected because the fitness of the three genetic encodings is very similar and higher than the trivial solution

[1] It is assumed that no faults occur in the expression table, since in any case it can be recovered from neighboring cells with a majority voting scheme.

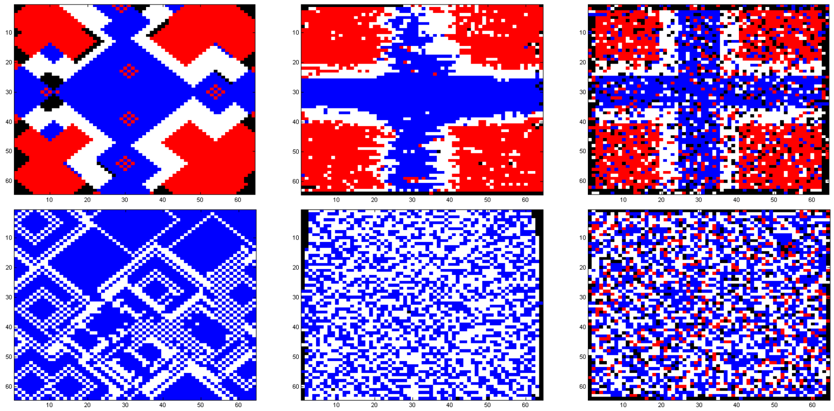


Fig. 6. Best evolved 64x64 solutions using, from left to right, **MS**, **AES** and direct coding. Norwegian flag above, CA-generated pattern below. Please refer to figures 3 and 4 for the actual targets.

consisting of exploiting only the frequency of colours as is the case with the CA-generated target. The damage rate (percentage of faulty cells) is varied between 0% and 100%. The damage process is repeated 100 times for each damage rate.

Figure 7 illustrates the results. While direct encoding is subject to a linear decrease in fitness, both developmental systems show a superior resistance to faults.

The **MS** benefits from the fact that chemical concentrations vary with continuity, and can be reconstructed with little effort. Also, evolution assigned the most frequent colour in the target to the default cell type, which explains the fitness value with 100% of faults.

In the case of the **AES**, fault recovery is a byproduct of ontogeny. These results are in support to what was previously observed in [6, 5].

5 Conclusions

We have tested the scalability of two developmental and one direct encoding strategy on a minimal task involving the evolution of specific target phenotypes.

Results show that the selected task, which was intended to be favorable to the direct encoding scheme, is easily solvable by the latter only for reasonably small target sizes. In these cases direct encoding greatly outperforms the developmental systems. Direct encoding benefits from the fact that each gene contributes independently to the fitness and therefore its landscape is both unimodal and non-deceptive.

On the other side, developmental systems suffer from the pleiotropy introduced by gene reuse. Also, as there is a single optimal solution, this utterly

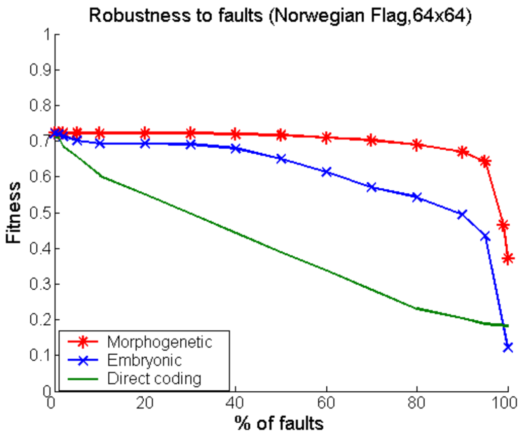


Fig. 7. Robustness of the **MS**, the **AES** and the direct coding on the Norwegian flag (size 64x64). Average over 100 tests. The fitness with 0% faults, is the one of the evolved solutions (.72 for development systems, .71 for direct encoding). Phenotypes with similar fitness scores were selected for better comparison. Fault recovery was not selected for.

complicates the evolutionary task, since it is not guaranteed to fall within the space of expressible phenotypes.

In any case, with bigger targets, development systems can take advantage of their reduced search spaces, and this is reflected in their performance levels starting at the 64x64 Norwegian flag and 96x96 CA-generated targets.

Developmental systems seem ‘smarter’ at finding exploitable regularities in the targets, such as shape and most frequent cell types. On the other side, this is impossible for direct encodings, so that errors in the evolved phenotypes must take the shape of high frequency noise. This gives a different ‘psychological’ perception of incomplete phenotypes and seems to affect performance for larger targets.

Finally, both developmental systems behave very well against phenotypic faults and are capable of recovering from significant amount of damage, even if the tested individuals were not selected for this characteristic.

As a last remark, we want to stress the lack of standard tasks usable for benchmarking developmental systems. In [4] the authors suggest 4 different tasks: evolution of pure symmetry, of specific shapes, of specific connectivity patterns and of a simple controller.

We believe that some of these, albeit interesting to demonstrate capabilities of a model in principle, leave doubts about the generality of the results. For example, the evolution of a controller is a task that imposes complex fitness landscapes with usually many optimal and suboptimal solutions, and therefore is difficult to analyse in relation to system evolvability.

On the other side, testing a developmental model against targets of various phenotypic complexity[2], from pure symmetry to total lack of it, may offer a good indication of the system strengths and weaknesses in more general settings.

[2] possibly calculating phenotypic complexity as its compressibility with standard algorithms

6 Acknowledgments

Daniel Roggen is funded by the Future and Emerging Technologies programme (IST-FET) of the European Community, under grant IST-2000-28027 (POETIC). The information provided is the sole responsibility of the authors and does not reflect the Community's opinion. The Community is not responsible for any use that might be made of data appearing in this publication. The Swiss participants to this project are funded by the Swiss government grant 00.0529-1.

References

1. Lehre, P., Haddow, P.C.: Developmental mappings and phenotypic complexity. Proceeding of CEC 2003, 62–68 (2003)
2. Tyrrell, A.M., Sanchez, E., Floreano, D., Tempesti, G., Mange, D., Moreno, J.M., Rosenberg, J., Villa, A.: POEtic Tissue: An Integrated Architecture for Bio-Inspired Hardware. In Tyrrell, A.M., et al., eds.: Proc. of the 5th Int. Conf. on Evolvable Systems (ICES 2003), Berlin, Springer (2003) 129–140
3. Roggen, D., Floreano, D., Mattiussi, C.: A Morphogenetic Evolutionary System: Phylogenesis of the POEtic Tissue. In Tyrrell, A.M., et al., eds.: Proc. of the 5th Int. Conf. on Evolvable Systems (ICES 2003), Berlin, Springer (2003) 153–164
4. Stanley, K., Miikulainen, R.: A taxonomy for artificial embryogeny. Artificial Life 9(2):93–130 (2003)
5. Federici, D.: Using embryonic stages to increase the evolvability of development. to appear in proceedings of WORLDS workshop at GECCO 2004 (2004)
6. Miller, J.: Evolving developmental programs for adaptation, morphogenesys, and self-repair. Proceeding of ECAL 2003, 256–265 (2003)
7. Coen, E.: The art of genes. Oxford University Press, New York (1999)
8. Haddow, P.C., Tufte, G., van Remortel, P.: Shrinking the Genotype: L-systems for EHW? In Liu, Y., et al., eds.: Proc. of the 4th Int. Conf. on Evolvable Systems (ICES 2001), Berlin, Springer (2001) 128–139
9. Eggenberger, P.: Cell interactions as a control tool of developmental processes for evolutionary robotics. In Maes, P., Mataric, M.J., Meyer, J.A., Pollack, J., Wilson, S.W., eds.: From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior, Cambridge, MA, MIT Press-Bradford Books (1996) 440–448
10. Gruau, F.: Automatic definition of modular neural networks. Adaptive Behavior **3** (1994) 151–183
11. Kumar, S., Bentley, P.J.: Biologically inspired evolutionary development. In Tyrrell, A.M., et al., eds.: Proc. of the 5th Int. Conf. on Evolvable Systems (ICES 2003), Berlin, Springer (2003) 57–68
12. Kitano, H.: Designing neural networks using genetic algorithms with graph generation system. Complex Systems, 4(4):461–476 (1990)
13. Luke, S., Spector, L.: Evolving graphs and networks with edge encoding: Preliminary report. In Koza, J.R., ed.: Late Breaking Papers at the Genetic Programming 1996 Conference. (1996) 117–124
14. Hornby, G.S., Pollack, J.B.: The advantages of generative grammatical encodings for physical design. In: Proceedings of CEC 2001, 600–607. (2001)