

Increasing the evolvability of development with Embryonal Stages

Diego Federici

Norwegian University of Science and Technology
Department of computer and information science
N-7491 Trondheim, Norway
`federici@idi.ntnu.no`

Abstract. Indirect encoding methods are aimed at the reduction of the combinatorial explosion of search spaces, therefore increasing the evolvability of large phenotypes.

These so called Artificial Ebryogeny systems have so far shown increased scalability for problems involving solutions of low complexity. This leaves open the more general question about the evolvability of complex phenotypes.

In this paper, we study the evolvability of a model of cellular growth regulated by a developmental program. Genotypes are selected for their ability to develop organisms of specif shape and cell types.

Results show that a particular method involving Embryonal Stages displays positive effects on the evolvability of developmental programs.

1 Introduction

The evolution of large phenotypes is one of the most serious problems in the field of evolutionary computation (EC). 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 side, 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 phene, since each gene influences several aspects of the phenotype.

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. The term ‘Artificial Embryogeny’ (AE) has been recently proposed to describe these evolutionary systems [1].

In AE, development is de facto a decompression of the genotype. Since compression is generally higher for regular targets, a serious question is how much these methods will prove viable for the evolution of high complexity phenotypes.

Hints in this direction, also come from a recent study on Matrix Rewriting [2], showing how the genotype-phenotype correlation decreases with the complexity of the phenotype [3].

In this paper we present a model of cellular growth which is targeted to the development of multi-cellular organisms of specific two dimensional shapes and colors. These organisms must be intended as a metaphor of functional devices, in which each color represents the specific function of the cell and the 2D displacement encodes their local connectivity.

For example, such organisms could develop decentralized locally connected digital circuits [4] or layers of artificial neural networks [5].

In AE, growth methods are either based on rewriting rules or cell chemistry models. The firsts, like the well known Cellular Encoding [6], evolve the rules of a grammar used to produce the mature organisms. The seconds proceed by evolving the cell metabolism thus controlling the state and development of the phenotype.

The model presented in this paper belongs to this second category. Phenotypes are multi-cellular organisms in which each cell shares the same growth program. The growth program regulates the cell type, chemical production and replication based only on the state of the particular cell and of its neighborhood.

Also belonging to this category, Bentley and Kumar proposed a model which develops 2D tiling patterns [7]. Cells can only be of a single type and the aim is to produce perfect tessellating phenotypes. The growth program is composed by a set of rules which upon matching the state of the local neighborhood activate a specific cellular response. Results showed that the systems performed and scaled better than a direct encoding method. On the other side, the best solutions developed had very regular phenotypes.

Miller extended Bentley and Kumar’s model and developed more complex patterns [8]. He allowed 4 different cell types (colors) and a chemical undergoing diffusion. The growth program is a boolean network evolved with the Cartesian Genetic Programming. Results showed evolved phenotypes resembling the target with only very few misplaced cells.

Additionally, Miller analyzed the behaviour of the evolved phenotype after the developmental step in which the fitness was computed and when subjected to severe mutilations. Phenotypes were shown to regrow the missing parts regaining qualitative resemblance to the target. The self repair feature is very interesting since it was not selected for during evolution.

Additional references can be found in the work of Stanley and Miikkulainen which have proposed a survey and a taxonomy for AE systems [1]. Another survey, addressing more specifically AE in control systems, can be found in [9].

Albeit that AE is showing promising results, it suffers from a general difficulty connected to the evolvability of the genotypes. Miller reported that in the development of a specific ‘french flag’ pattern few runs produced satisfiable results. The other tended to be stucked in local optima.

One of the reasons for this is intrinsic to the idea of gene reuse. In fact, if we imagine an individual of high fitness with only a few misplaced phenes, a direct encoding method could allow the tweaking of the few corresponding uncorrect genes allowing the cumulative refinement of the phenotype. In the case of indirect encoding, the change of the same few phenes may require a complete redesign

of the genotype. In fact, the corresponding genes might be responsible for other features of the phenotype in other developmental stages. Their change may cause interferences in the maturation of the organism with catastrophic effects.

Therefore AE models may be prone to create deceptive fitness landscapes as the results in [3] seem to suggest. To reduce this effect and increase evolvability, in this paper we have adopted three strategies.

1. An Artificial Neural Network (ANN) encodes the growth program. Compared to discrete rules, the space of continuous functions representable by ANNs allows a finer tuning of cellular responses. In this case, escaping local optima should be easier.
2. Population diversity is increased rewarding fitness to individuals with rare phenes. This reduces the chances that innovation, which in developmental systems have saltatory characteristics, may favor a single strain of genotypes.
3. Development may happen in more than one Embryonal Stage. The single growth program is sub-divided in several stages each one governing the development at subsequent times. Stages are evolved incrementally, the first being evolved before. Embryonal Stages resemble but are capable of differentiate from the previous ones, therefore allowing genetic refinement without interference and a ‘zoom-in effect’ in the search space.

The remaining of the paper is organized as follows: section 2 contains a description of the evolutionary task, section 3 the developmental model, section 4 the details of the genetic algorithm, section 5 the results of the simulations and section 6 the conclusions.

2 The evolutionary task

Yet another issue in AE is the proper choice of the targets used for benchmarking. In [1] the authors suggest 4 different tasks: evolution of pure symmetry, of specific shapes, of specific connectivity patterns and of a simple controller.

In the simulations presented we have selected four shapes with various levels of symmetry. Fig.1A is a pattern composed of three colored stripes similar to the one used in [8]. Fig.1B has a bounding layer which insulates the internal cell type from the outside. Fig.1C contains repetitions of a simple ‘plus’ pattern. These sort of regularities should be exploitable by AE systems. Fig.1D is a more complex Norwegian flag pattern and can be seen as a vertical and horizontal insulated wiring.

Fitness is proportional to the resemblance of an individual to the target, and is computed as shown in equation 1.

$$FIT(P, T) = \left(\sum_{x,y} EQUALS(P, T, x, y) \cdot PheneValue(x, y) \right) / ||T|| \quad (1)$$

$$EQUALS(P, T, x, y) = \begin{cases} 0 & \text{if } P(x, y) \neq T(x, y) \\ 1 & \text{if } P(x, y) = T(x, y) \end{cases}$$

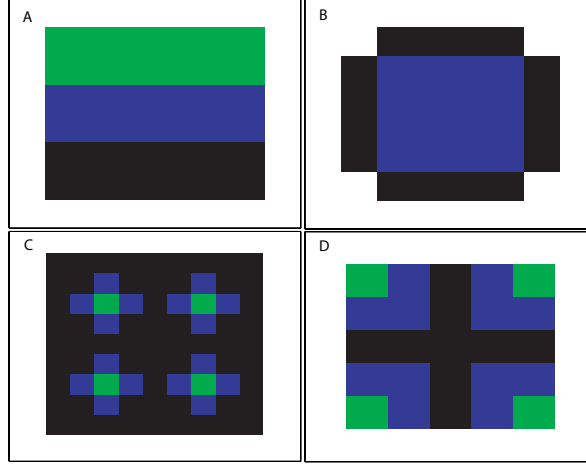


Fig. 1. The four targets of the evolutionary task. A) a three-stripes pattern, B) a bounded pattern, C) a group of pluses, D) a Norwegian flag pattern. Development could take advantage of the various degrees of symmetry and modularity of the targets.

where P is the phenotype to be evaluated, T the target, and $PheneValue(x, y)$ is the frequency in the population of the phene in position x, y . $PheneValue$ is used to increase population diversity (see also section 4.1).

3 The developmental model

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. The execution order is determined by age, older cells being taken first.

Cells (see figure 2) are characterized by internal and external variables. Internal variables define the cell state and move with it, while external ones (chemicals) belong to the environment and follow a simple diffusion law.

At each developmental step, any existing cell can release chemicals, change its own type, alter the internal metabolism and produce new cells in the cardinal directions North, West, South and East. If necessary, existing cells are pushed sideways to create space for the new cells (see figure 3). When a cell is pushed outside the boundaries of the grid it is permanently lost.

Morphogenesis is governed by an Artificial Neural Network (Morpher) defined by the genotype. The genotype is a direct gray-code representation of the Morpher. The hyperbolic tangent is used as transfer function.

The Morpher (figure 4) receives in input the current cell internal and external variables, and the cell types of the neighboring cells in the four cardinal directions. Its output determines the new internal and external variables of the cell.

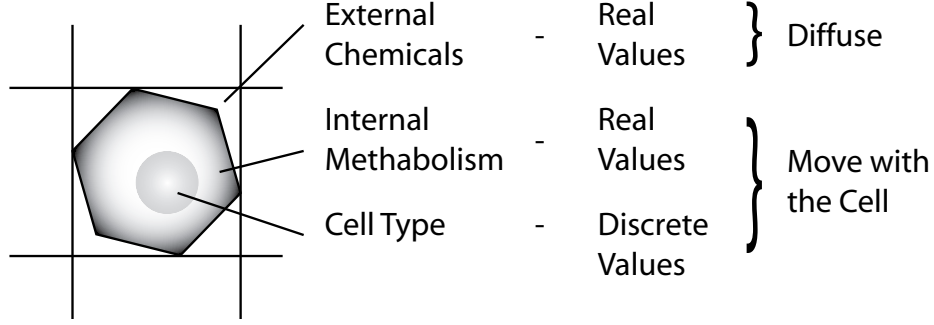


Fig. 2. Description of the variables used for development. External variables follow a diffusion law, while internal ones move with the cell. While chemical concentrations, internal and external, vary in the range $[0, 1]$, the cell type can take one of 4 discrete values

and, in case, the internal variables of the newly generated cells. An additional local variable, the cell age, is set to 1 at birth and decays exponentially to 0.

Chemical production and internal metabolism values are read directly to and from input and output lines, one line being dedicated to each different internal or external chemical. The cell type is encoded / decoded from a single value as shown in Eq.2 and 3.

$$Encode(value) = \begin{cases} 0 & \text{if } v < -2/3 \\ 1 & \text{if } v \in (-2/3, 0) \\ 2 & \text{if } v \in [0, 2/3) \\ 3 & \text{if } v > 2/3 \end{cases} \quad (2)$$

$$Decode(type) = \begin{cases} -1 & \text{if } t = 0 \\ -1/3 & \text{if } t = 1 \\ 1/3 & \text{if } t = 2 \\ 1 & \text{if } t = 3 \end{cases} \quad (3)$$

In these simulations, feedforward networks with four hidden units have been used.

4 The evolutionary model

Each population in the simulations presented is composed by 400 individuals undergoing elitarian selection with a survival share of $1/8$. A tenth of the new individuals are produced by crossover, while all the offspring undergoes mutation. The mutation operator takes each weight with a P_{mut} probability and adds to it Gaussian noise with V_{mut} variance.

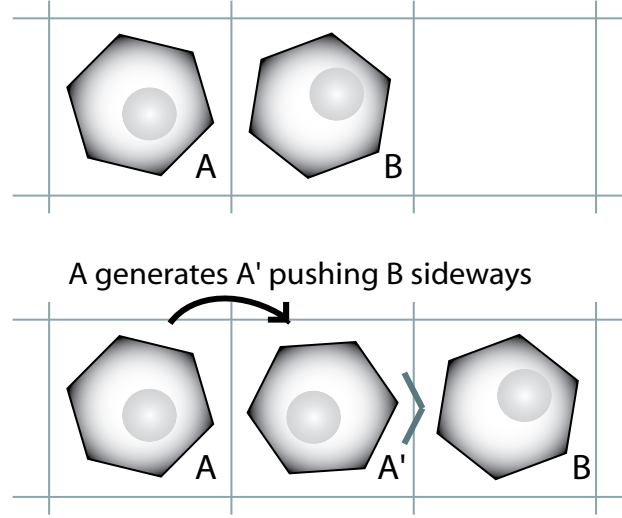


Fig. 3. Placement of new cells. If necessary, space is created by pushing sideways existing cells. Cells falling off the development grid are lost.

P_{mut} and V_{mut} vary in the ranges $[.01, .2]$ and $[.035, .1]$ respectively. Their value is proportional to the time passed from the last increase in the top fitness score, reaching the maximum in ten generations. These values have been selected after a preliminary study on short evolutionary runs, proving to be more effective than fixed ones.

4.1 population diversity

Often, in AE systems, evolutionary improvements have saltatory characteristics. Under these conditions a positive innovation can increase the reproductive chance of a particular strain reducing the chance of survival of all others. This increases the chances of convergence to local optima.

To increase population diversity, fitness scores are modified looking at the frequency of the phenes that individuals possess, counteracting homogenization and favoring individuals with rare characteristics:

- (1) the population is first ordered by fitness values before modification (in case of ties younger individuals go first).
- (2) fitness scores are recomputed following this order, but the value for each phene (*PheneValue* in Eq.1) decreases linearly with use from 1 to 1/100.

4.2 embryonal stages

Biological organisms have the interesting property that embryonal developmental stages of philogenetically related species share similarities:

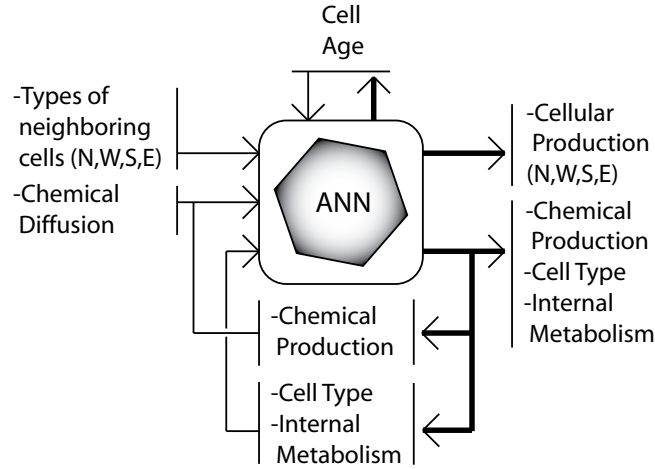


Fig. 4. Inputs and outputs of the growth program, the Morpher, implemented with a feedforward ANN. Each cell internal variables, cell type and metabolism, implement a direct feedback pathway, while chemical production and diffusion offer a channel for longer range communication.

It is generally observed that if a structure is evolutionary older than another, then it also appears earlier than the other in the embryo. Species which are evolutionary related typically share the early stages of embryonal development and differ in later stages. [...] If a structure was lost in an evolutionary sequence, then it is often observed that said structure is first created in the embryo, only to be discarded or modified in a later embryonal stage.

Wikipedia [10]

This apparent relationship between ontogeny and phylogeny, which should not be confused with the discredited Recapitulation theory, suggests that multicellular organisms evolve new traits incrementally over older phenotypic characteristics.

In fact, the modification of early embryonal stages may disrupt development with catastrophic results. Therefore, mutations affecting later stages of the ontogenetic process will have a higher probability to be useful.

This suggests, also for AE, that decreasing the chances of modification of the early steps of development will increase evolvability. Although such preservation mechanism could be found by means of evolution, to simplify the evolutionary task we propose an explicit mechanism for it.

We allow growth to be controlled by a set of programs. Phenotypes are developed in subsequent embryonal stages, each one governed by a different program.

At the beginning of the evolutionary search, organisms develop in a single stage. When a certain performance or generation are reached a new stage is

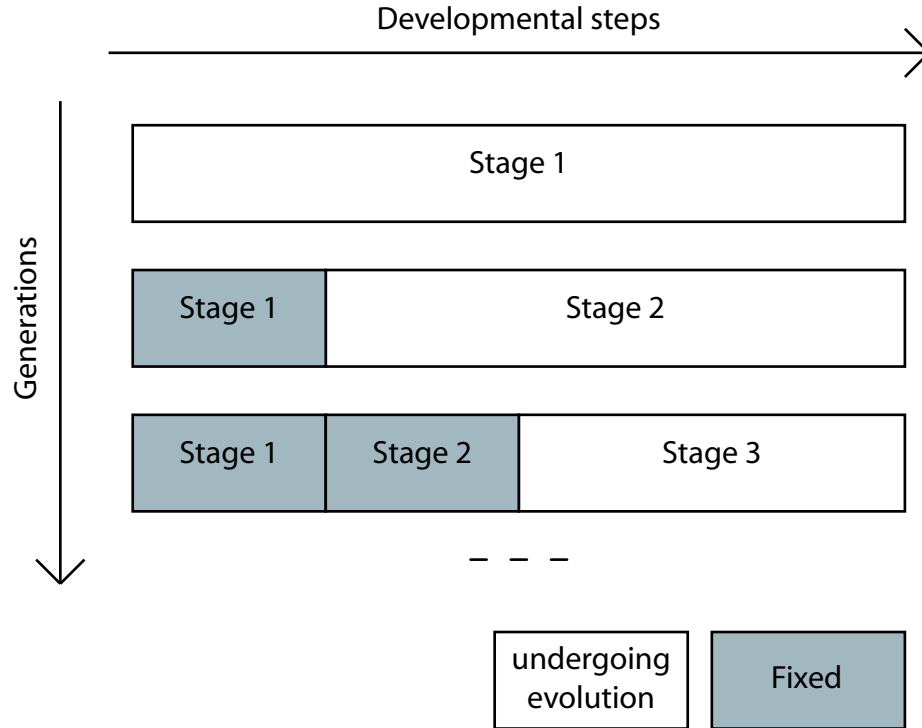


Fig. 5. Embryonal stages. Stages are introduced incrementally, and they govern ontogeny from a pre-determined developmental step. Only the latest step is subject to evolutionary operators, the others are fixed.

added. While the older stage starts the ontogenetic process as usual, the new one will assume control at a pre-determined developmental step completing the maturation of the organism (see Figure 5).

The new stage developmental program is initialized as a copy of the previous one. In this way, at first, the introduction of new stages does not alter ontogeny. On the other side, the innovation operators are allowed to modify only the program of the latest embryonal stage without affecting any of the previous ones.

Additional stages, being built upon the previous ones, add resolution to the specific spot in the search space. This allows an incremental refinement of the ontogenetic process, helping escaping local optima due to interference effects.

In the case of complex phenotypes, this positive effect should be even more visible, since the amount of information required to produce them is higher, as the chances of gene's interference.

5 Results

The performance of runs with and without embryonal stages is compared. Simulation specific details are given below:

- 1000 generations maximum.
- fitness is computed at developmental steps 7 and 8. Fitness values are expressed in % of target resemblance.
- simulations can have either 1 or 5 embryonal stages. Where used, new stages are introduced at generations 500, 750, 875 and 938, or when a maximum population fitness of 75%, 83%, 92% and 100% is reached. Each stage takes development from steps 0, 4, 5, 6 and 7.
- cells can release one external and one internal chemicals.

Statistics from 10 runs with each parameter settings (target and number of stages) are shown in the following table.

phases		target shape (see fig.1)			
		bounded	3 stripes	flag	pluses
1	max	95%	84%	86%	83%
	avg	83%	76%	78%	79%
	min	70%	68%	72%	78%
5	max	99%	98%	94%	88%
	avg	91%	88%	88%	83%
	min	89%	77%	76%	81%
fitness change		+4%	+16%	+12%	+5%

From the results above, it is evident that embryonal stages have a positive effect in all cases. Also, performance is somewhat proportional to the regularity of target phenotypes.

In figures 6–9 is possible to see the development of the best evolved individuals. Steps 7 and 8 (highlighted in the figures) are the only ones used to calculate fitness. This means that the following steps, 9 and 10, are not relevant for selection purposes. Even though, individuals maintain a resemblance to the target, with a few of them, such as the one in figure 7, reaching a developmental stasis.

The modular target of figure 9, which is also the less regular, proved to be the most difficult to evolve. Development is very chaotic until it reaches the steps in which fitness is checked. This lack of regularity during development suggests that, under such conditions, modular decomposition will be hardly achieved by means of gene reuse. In fact, neighbors and chemical concentrations will generally share little similarity in the places where modules should develop.

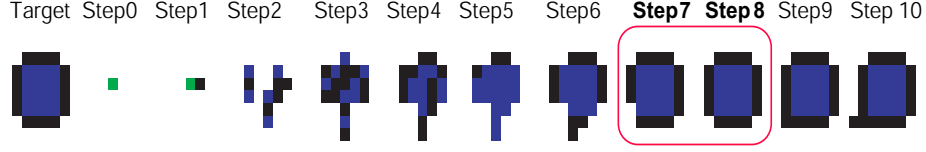


Fig. 6. development of the best ‘bounded’ individual

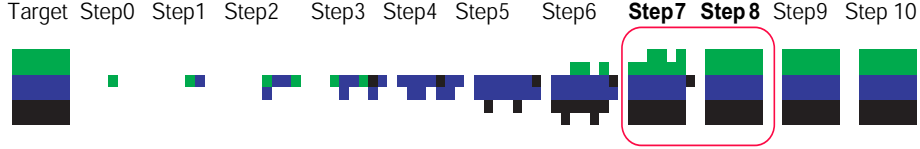


Fig. 7. development of the best ‘3 stripes’ individual

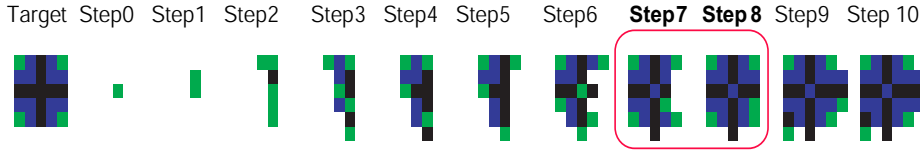


Fig. 8. development of the best ‘norwegian flag’ individual

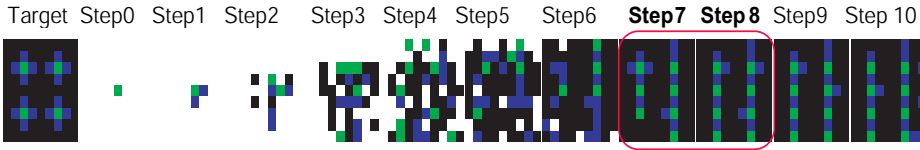


Fig. 9. development of the best ‘pluses’ individual

We have also checked how well these organisms responded to damage. The best evolved individuals have been developed with various degrees of cellular necrosis. At selected developmental steps, a number of cells have been removed from the phenotype (but at least one cell was left). The average fitness of the final organisms are plotted in figure 10.

Since cells in early developmental steps are more important for development, one would expect a domino effect on fault propagation. Still performance degrades linearly the earlier damage is applied. Organisms seem to limit the effect of faults during development. This behavior was not selected for and is similar to what reported in [8].

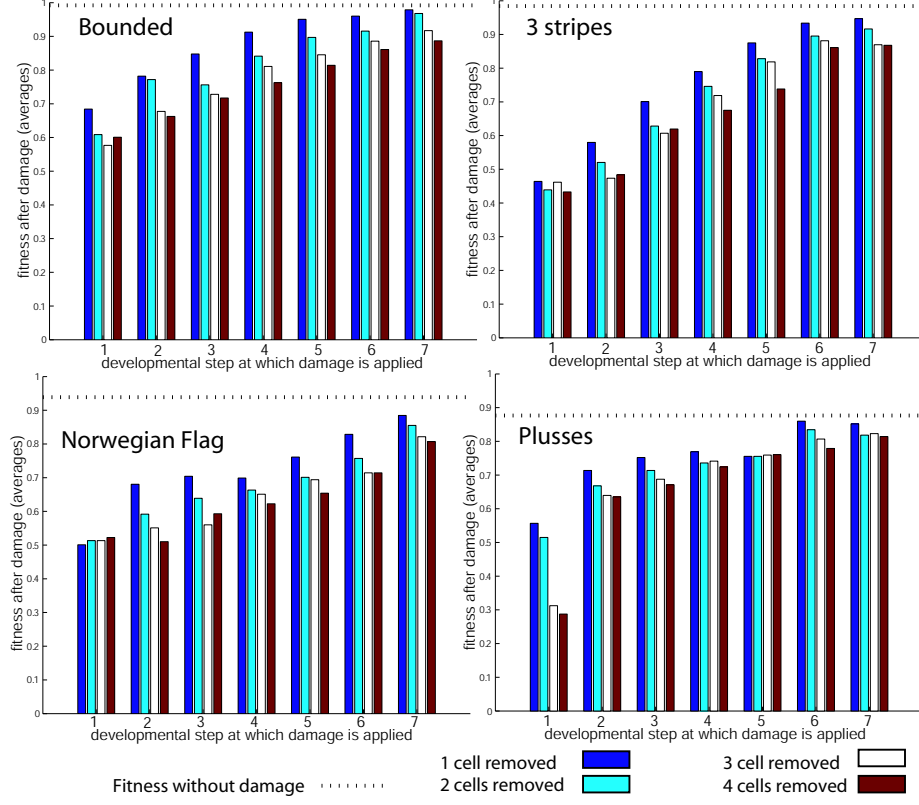


Fig. 10. Impact of cell death at different developmental steps. Average fitness computed over 20 random development faults consisting of 1 to 4 cell deaths. Contrary to intuition, faults do not propagate exponentially during development.

6 Conclusions

An artificial embryogeny (AE) model has been proposed and tested in the development of four two dimensional targets of specific morphology. The system is based on the evolution of a growth program that regulates the ontogeny of multicellular organisms starting from a single egg cell.

This model is aimed at the construction of an evolutionary platform for the development of functional organisms, such as locally connected digital circuits [4] or neural networks [5].

The primary aim of the paper is to investigate the effects of multiple embryonal stages on evolvability. This method of incremental development is devised to reduce the catastrophic interference caused by the change of genes that regulate ontogeny in different growth phases.

Results show that embryonal stages have positive effects on performance, even if specific targets still prove hard to evolve.

Also, the behavior of phenotypes undergoing different degrees of damage during development was analyzed. Similar to results in [8] individuals showed a good resistance to faults. This is particularly interesting since there was no selection for this characteristic.

6.1 further work

The role of chemicals, both internal and external, deserves additional investigation. For example, we have noticed that the evolution of the ‘bounded’ target benefits from the presence of an external chemical, while the ‘pluses’ performs better when only internal chemicals are present. For the other two targets, performance increases when chemicals are not present at all.

Also, the scalability of these development systems should be put to the test, searching for the relation between evolutionary effort and size of the search space.

Acknowledgements

I wish to thank Julien Miller for the useful discussions that inspired the work presented in this paper, and Keith Downing and Gunnar Tufte for the valuable suggestions. The simulations presented in this work were run on the inexpensive ClustIS cluster [11].

References

1. Stanley, K., Miikulainen, R.: A taxonomy for artificial embryogeny. *Artificial Life* 9(2):93–130 (2003)
2. Kitano, H.: Designing neural networks using genetic algorithms with graph generation system. *Complex Systems*, 4(4):461–476 (1990)
3. Lehre, P., Haddow, P.C.: Developmental mappings and phenotypic complexity. *Proceeding of CEC 2003* (2003)
4. Tufte, G., Haddow, P.C.: Insertion of functionality into development on an sblock platform. *Proceeding of CEC 2003* (2003)
5. Federici, D.: Evolving a neurocontroller through a process of embryogeny. submitted to SAB conference 2004 (2004)
6. Gruau, F., ed.: *Neural Network Synthesis using Cellular Encoding and the Genetic Algorithm*. PhD Thesis, Ecole Normale Supérieure de Lyon (1994)
7. Bentley, P., Kumar, S.: Three ways to grow designs: A comparison of embryogenies for an evolutionary design problem. *Proceeding of GECCO 1999*, 35–43 (1999)
8. Miller, J.: Evolving developmental programs for adaptation, morphogenesis, and self-repair. *Proceeding of ECAL 2003*, 256–265 (2003)
9. Kodjabachian, J., Meyer, J.: Evolution and development of control architectures in animats. *Robotics and Autonomous Systems*, 16(2–4) (1995)
10. Wikipedia: Ontogeny and phylogeny. http://en2.wikipedia.org/wiki/Ontogeny_and_phylogeny (2003)
11. Cassens, J., Fülöp, Z.C.: It’s magic: Sourcemage gnu/linux as hpc cluster os. in *Proceedings Linuxtag 2003*, Karlsruhe, Germany (2003)