Using Embryonic Stages to increase the evolvability of development

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 Embryogeny 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 introduce a novel method of cellular growth regulated by a developmental program. Genotypes are selected for their ability to develop organisms of specific shape and cell types.
Results show that the use of Embryonic Stages, which involves the incremental addition of growth programs, displays positive effects on the evolvability of development.

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 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. 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 [2] on Matrix Rewriting [3], showing how the genotype-phenotype correlation decreases with the complexity of the phenotype.

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 former, such as the Matrix Rewriting scheme [3], Cellular Encoding [6], Edge Encoding [7] and the GenRe system [8], evolve the rules of a grammar used to produce the mature organisms. The latter 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 [9]. 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 [10]. 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 behavior of the evolved phenotype after the developmental step in which the fitness was computed and when subjected to severe mutilations. Phenotypes were shown to re-grow the missing parts regaining qualitative resemblance to the target. The self repair feature is very interesting since it was not selected for during evolution.

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 stuck 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 phenotypic traits (phenes), a direct encoding method could allow the tweaking of the few corresponding incorrect 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 pleiotropic genotype. In fact, the corresponding genes might be responsible for other features of the phenotype in other developmental stages. Their change may cause interference in the maturation of the organism with catastrophic effects.

Therefore AE models may be prone to create deceptive fitness landscapes as the results in [2] 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 recursive ANNs allows finer tuning of cellular responses. Also, recursive ANN have been suggested as a model for gene regulatory networks [11].
2. Population diversity is increased rewarding fitness to individuals with rare phenotypic traits. This reduces the chances that innovation, which in developmental systems have saltatory characteristics, may favor a single strain of genotypes (see also section 4.1).
3. Development may happen in more than one Embryonic Stage [12], i.e. the growth program is composed by several sub-programs (Embryonic Stages) each one governing development at subsequent times. Stages are evolved incrementally, those governing earlier developmental steps being evolved before. Embryonic Stages resemble but are capable of differentiating from the previous ones, therefore allowing genetic refinement without interference and a ‘zoom-in effect’ in the search space (see also section 4.2).

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 an other 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.

Here we focus on the first three tasks. The different evolutionary targets are shown in figure 1. Fig.1A is a pattern composed of three colored stripes similar to the one used in [10]. Fig.1B presents a membrane which insulates the internal cells from the outside. Fig.1C contains repetitions of a simple ‘plus’ pattern. This regularity should be exploitable by AE systems. Fig.1D is a more complex ‘Norwegian flag’ pattern which also presents a high degree of symmetry. Fig.1E contains a pattern generated by a CA using Wolfram’s rule 90 and random initial conditions (different for each evolutionary run). This method has been chosen since it steadily generates targets of high complexity.

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

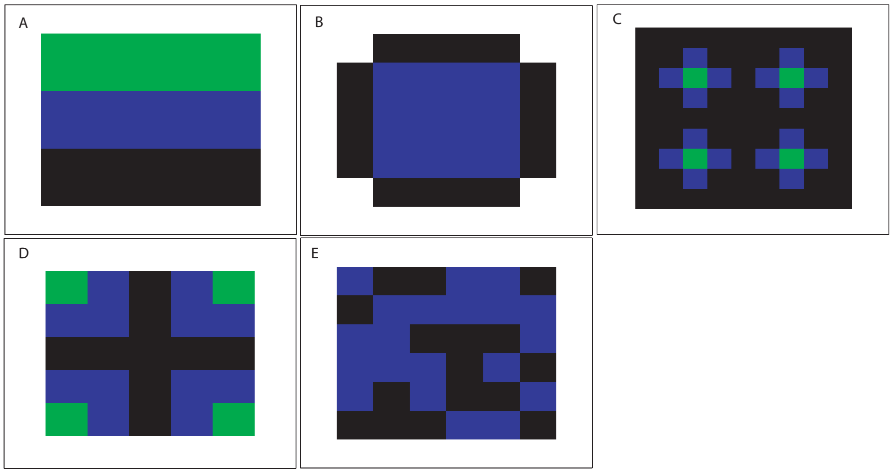

Fig. 1. The targets of the evolutionary task. A) a three-stripes pattern, B) a bounded pattern, C) a group of pluses, D) a Norwegian flag pattern and E) pattern generated by a CA using Wolfram rule 90 starting with initial random conditions (the first row). Development should take advantage of the various degrees of symmetry and modularity of the targets.

$$
\begin{gathered}
FIT(P,T) = \left(\textstyle\sum_{x,y} \; EQUALS\,(\,P,T,x,y\,) \cdot PheneValue(x,y) \right) \; / \; \|T\| \\
EQUALS\,(\,P,T,x,y\,) = \begin{cases} 0 \; if \; P(x,y) \neq T(x,y) \\ 1 \; if \; P(x,y) = T(x,y) \end{cases}
\end{gathered}
\tag{1}
$$

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, a cell at a time. The execution order is determined by age, older cells being taken first.

Cells 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 conservative diffusion law.

At each developmental step, any existing cell can release a chemical, 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. When a cell is pushed outside the boundaries of the grid it is permanently lost.

The growth program is governed by feed-forward Artificial Neural Networks (Morphers) with a single hidden layer. Each Morpher is specified by 225 genes (floating values), one for each of the input, output and bias weights.

The Morpher (figure 3) has 8 inputs (the types of the 4 neighboring cells, the cell type and internal metabolism, the chemical concentration at the cell position and the cell age), 8 hidden units and 16 outputs (4 specify if new cells are produced in the N-W-E-S directions, 8 describe the type and internal metabolism of eventual new cells, 1 trigger the change of the cell type, 2 describe the new type and internal metabolism and 1 for the chemical production).

The cell age, is a variable set to 1 at birth which decays exponentially to 0.

Chemical production and internal metabolism are floating point values in the range [-1,1], while the discrete cell type is encoded/decoded as shown below:

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

4 The evolutionary model

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

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 were selected after a preliminary study on short evolutionary runs, proving to be the most effective.

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.

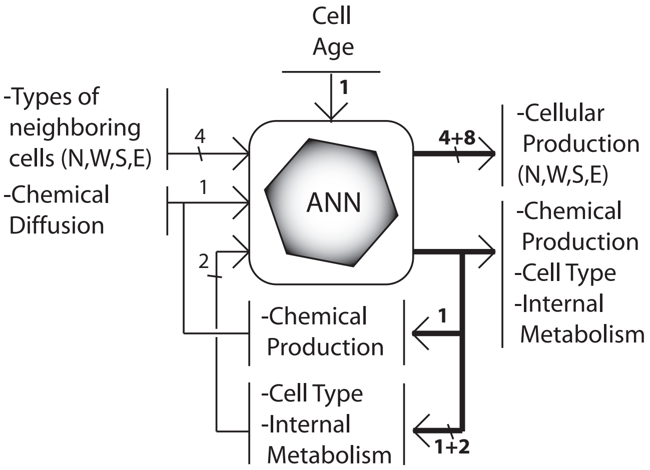

Fig. 2. The growth program (Morpher) input and output lines with their respective sizes. Each line is a floating point value $\in [-1, 1]$. The Morpher is implemented by a feed-forward ANN, even though each cell internal variables (cell type and metabolism), implement a direct feedback pathway. Chemical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

4.2 Embryonic Stages

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

> 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 [12]

This apparent relationship between ontogeny and philogeny suggests that multi-cellular organisms evolve new traits incrementally over older phenotypic characteristics.

In fact, the modification of early Embryonic Stages may disrupt development with catastrophic results. Therefore, mutations affecting only later stages of the ontogenetic process will have higher probability to be successful.

This suggests, also for AE, that decreasing the chances of modification of the early steps of development will increase system evolvability. This is in general hard to achieve since gene reuse will produce a high level of pleiotropy, in this case of genes having multiple effects in different stages of development.

Since pleiotropy locks otherwise unrelated phenotypic traits, evolvability can be improved by minimizing its impact.

One possibility is to incrementally partition development in subsequent stages, each one characterized by a different chromosome specifying a complete growth program (a Morpher)[1].

Evolution of development programs with Embryonic Stages proceeds as follows (see also Figure 3):

- At the beginning of the evolutionary search, organisms will develop in a single stage (chromosome).
- When given conditions are met a new stage (chromosome) can be added.
    - The chromosome of the new stage (specifying its Morpher) is initialized as a copy of the previous one.
    - While the older stage initiates ontogeny as usual, the new one will assume control at a pre-determined developmental step completing the maturation of the organism. Notice that, since every new stage is a copy of the previous one, at first development is not altered.
- From now on, the evolutionary operators are allowed to affect only the chromosome of the new stage, i.e. all the others are fixed. Therefore the first phases of ontogeny are fixed too, and evolution can affect only the final steps of development. In other words, the addition of a new stage restricts the evolutionary task since: 1) the target is unchanged, 2) organisms develop from a, presumably, easier to tackle starting condition.
- Additional stages can be added likewise.

As a consequence Embryonic Stages yield the following properties:

- Among different stages, pleiotropy is avoided.
- As new stages are being introduced, a greater part of development is excluded from search, and evolution can focus on the refinement of fewer and fewer steps. This means that: (1) since the size of the search space is always constant (a chromosome), each additional stage increases the resolution of the map from genotype to phenotype around promising areas of the solution space; (2) the increase in resolution alters the fitness landscape and the disposition of local optima among different stages. Therefore the addition of a new stage can be beneficial against stagnation.
- Since each new stage chromosome is a copy of the one of the previous stage, the information defining the individual's ontogeny is conserved and evolution continues to operate incrementally.

5 Results

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

- 1000 generations maximum.

[1] Although such preservation mechanism could be found by means of evolution, to simplify the evolutionary task we propose an explicit mechanism for it.

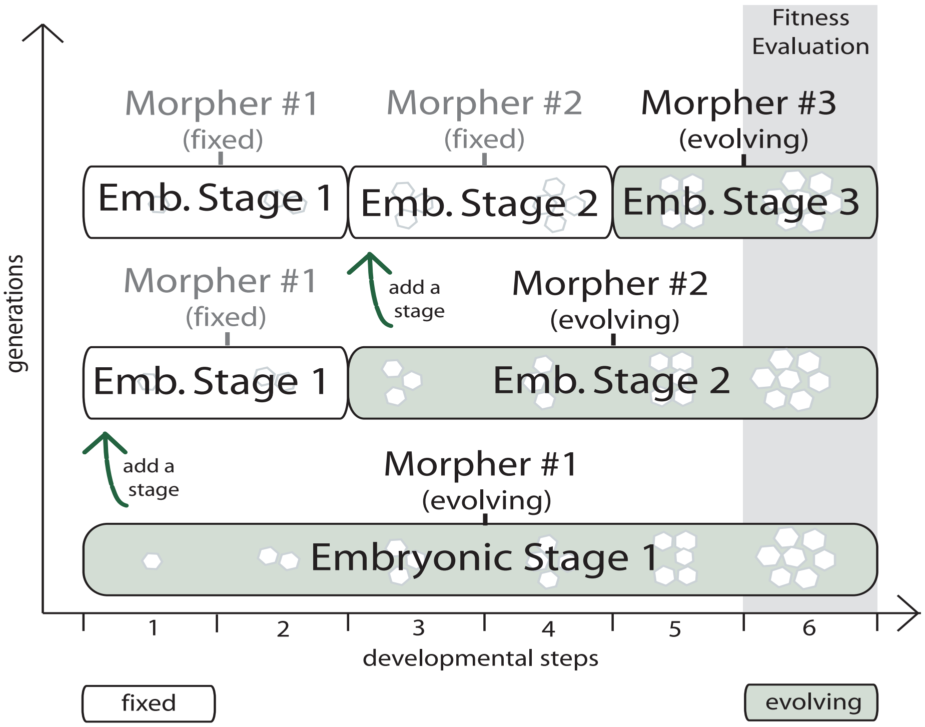

Fig. 3. Embryonic Stages. Stages are introduced incrementally during evolution, each governing a part of the ontogeny process by means of a unique Morpher. Only the latest stage, controlling the final steps of development (shaded in the figure), is subject to the evolutionary operators. This method reduces the pleiotropy caused by the reuse of genes at different developmental steps, therefore simplifying the evolutionary task. New stages are introduced when a certain generation is reached or when search appears stagnant. Fitness is computed always in the same way and at the same developmental step.

– fitness is computed at developmental steps 7 and 8. Fitness values are expressed in % of target resemblance. Fitness is calculated as an average over several steps to favor the emergence of development homeostasis [10].

– simulations can have either 1, 3 or 5 Embryonic Stages. A new stage may be introduced when fitness has not improved for the last 100 generations or in any case at generations 500 (stage 2), 750 (stage 3), 875 (stage 4) and 938 (stage 5). When 3 are used, stages control development at steps 1-6 (stage 1), 7 (stage 2) and 8 (stage 3). In the case of 5, steps 1-4 (stage 1), 5 (stage 2), 6 (stage 3), 7 (stage 4), 8 (stage 5).

– cells can release one external and one internal chemicals.

Statistics from 20 runs with each parameter settings (target and number of stages) are shown in the following table. The performance of 400 randomly generated genotypes is also given for comparison.

		target shape (see fig.1)					overall
stages		bounded	3 stripes	flag	pluses	Wolfram	average
	max	95%	89%	90%	84%	78%	
1	avg±std	82±3%	78±6%	78±5%	80±2%	72±4%	**78±5%**
	max	94%	95%	94%	88%	88%	
3	avg±std	88±3%	87±5 %	86±6%	84±2%	82±4%	**85±5%**
	max	99%	98%	94%	88%	93%	
5	avg±std	90±5%	88±5%	85±6%	84±2%	81±6%	**86±6%**
	random genotypes	35±10%	36±10%	34±7%	17±12%	33±8%	

An ANOVA test with 99% confidence was performed on the results, showing that simulations with more than one Embryonic Stage proved significantly better. Still, even if the best results were achieved with 5 stages, the difference between 3 and 5 stages its not statistically significative. Also, performance is somewhat proportional to the regularity of the target phenotypes, with the Wolfram's rule 90 targets scoring the worst average fitness.

Against intuition, the modular 'pluses' target proved to be very difficult to evolve. Development seems very chaotic until it reaches the final steps where fitness is checked (see figure 4). 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.

In figure 4 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 neutral to selection. Even though, some individuals maintain a resemblance to the target, with a few of them, such as the second reaching a developmental stasis.

Finally, we checked how well these organisms responded to damage. The best evolved individuals have been developed with various degrees of cellular necrosis. At selected steps during development, a number of cells have been removed from the phenotype (but at least one cell was left alive). The average fitness of the final organisms are plotted in figure 5.

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

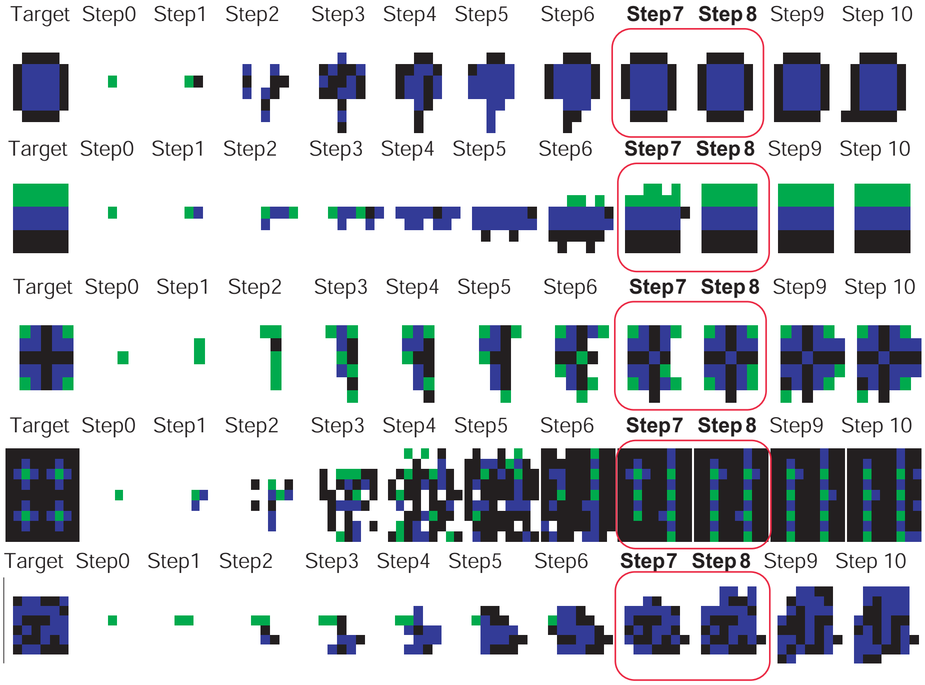

Fig. 4. The development of some of the best evolved individuals

of faults during development. This behavior was not selected for and is similar to what reported in [10] on a similar developmental model based on cell chemistry.

6 Conclusions

An artificial embryogeny (AE) model has been proposed and tested in the development of 2D 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 Embryonic 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.

Embryonic Stages operate by incrementally locking early phases of development while increasingly adding resolution to promising areas of the search space. Results show that this method has 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 [10] individuals showed a good resistance to faults. This is particularly interesting since there was no selection for this characteristic.

6.1 Further work

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.

Acknowledgments
I wish to thank Julian 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 [13].

References

1. Stanley, K., Miikulainen, R.: A taxonomy for artificial embryogeny. Artificial Life 9(2):93–130 (2003)
2. Lehre, P., Haddow, P.C.: Developmental mappings and phenotypic complexity. Proceeding of CEC 2003, 62–68 (2003)
3. Kitano, H.: Designing neural networks using genetic algorithms with graph generation system. Complex Systems, 4(4):461–476 (1990)
4. Tufte, G., Haddow, P.C.: Insertion of functionality into development on an sblock platform. Proceeding of CEC 2003, 731–738 (2003)
5. Federici, D.: Evolving a neurocontroller through a process of embryogeny. to appear in proceedings of SAB conference 2004 (2004)
6. Gruau, F.: Neural Network Synthesis using Cellular Encoding and the Genetic Algorithm. PhD Thesis, Ecole Normale Superieure de Lyon (1994)
7. 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
8. Hornby, G.S., Pollack, J.B.: The advantages of generative grammatical encodings for physical design. (In: Proceedings of CEC 2001, 600–607)
9. 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)
10. Miller, J.: Evolving developmental programs for adaptation, morphogenesys, and self-repair. Proceeding of ECAL 2003, 256–265 (2003)
11. Somogyi, R., Sniegoski, C.A.: Modeling the complexity of genetic networks: understanding multigenic and pleiotropic regulation. Complexity, 1, 45-63 (1996)
12. Wikipedia: Ontogeny and phylogeny. http:// en2.wikipedia.org /wiki /Ontogeny_and_phylogeny (2003)
13. Cassens, J., Fülöp, Z.C.: It's magic: Sourcemage gnu/linux as hpc cluster os. in Proceedings Linuxtag 2003, Karlsruhe, Germany (2003)

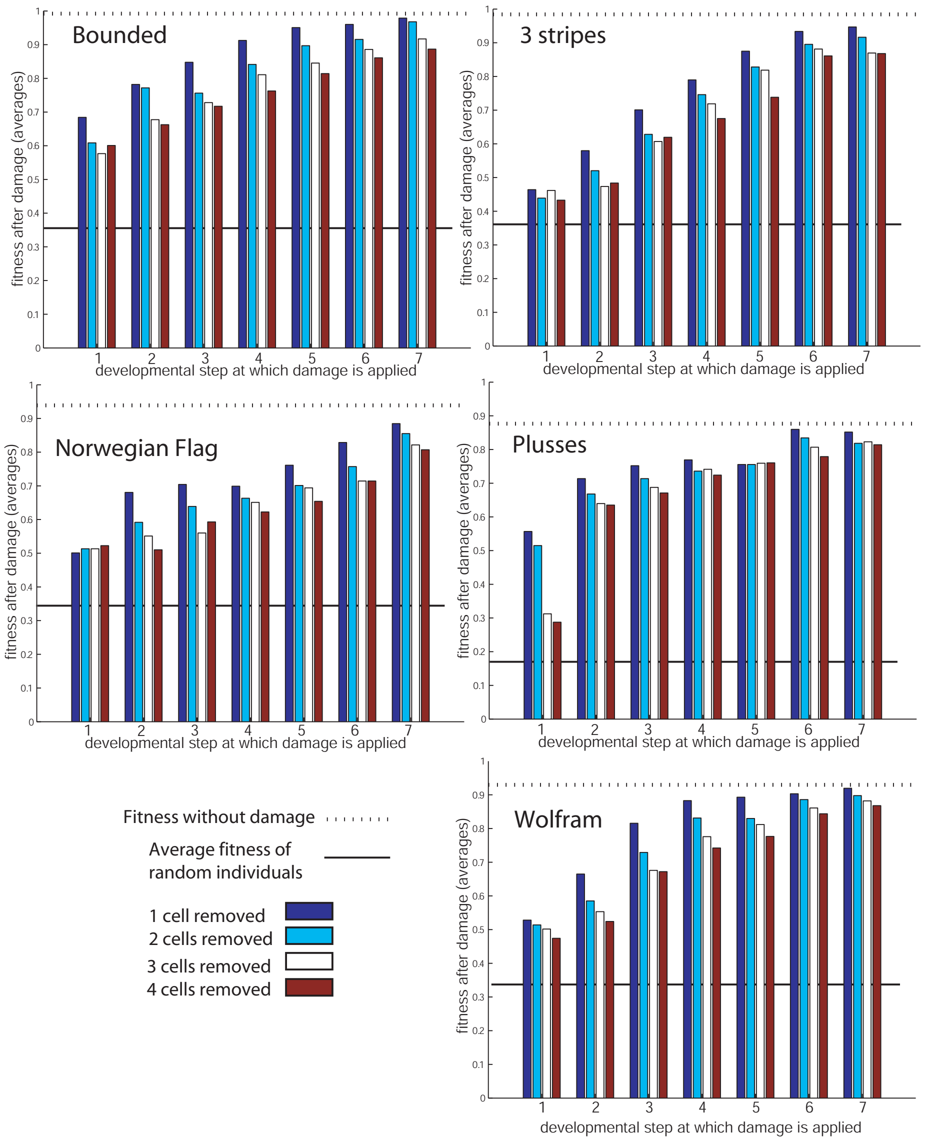

Fig. 5. Impact of cell death at different developmental steps. Average fitness computed over 20 random development faults consisting of 1 to 4 cell deaths. Contrary to intuition, faults do not propagate exponentially during development.