How to evolve the Head-Tail Pattern from Reaction-Diffusion Systems

Felix Streichert, Christian Spieth, Holger Ulmer, and Andreas Zell
Centre for Bioinformatics Tübingen (ZBIT), University of Tübingen,
Sand 1, 72076 Tübingen, Germany,
streiche@informatik.uni-tuebingen.de
http://www-ra.informatik.uni-tuebingen.de/

Abstract

The possible application of evolving artificial embryos to build functional machinery is a promising area of research. Unfortunately, there are still many fundamental problems to be solved before Artificial Embryology can be applied to such tasks, not to mention the necessary hardware. In this paper we address the problem of how to evolve the head-tail pattern for an artificial embryo endogenously, without predefined asymmetric cell division or external guide through polar cells or exogenous sources of morphogenes. We examine the performance of an Evolutionary Algorithm on two different fitness functions. Further, we examine the evolvability of several mathematical models for regulatory networks, controlling the behavior of the digital embryo.

1. Introduction

Currently, there is a trend in technology to growing complexity. For example, machines are assembled from several tens of thousands of parts, and modern microchips are built from millions of transistors. If this trend continues, it becomes more and more difficult to sequentially assemble such complex hardware, and structures could also become too small and complex for common production processes.

Self-assembly and self-organization are visionary properties of future hardware, which could solve these problems. Although the actual technology will not be available for many years to come, the necessary properties of such a technology can be studied using simulation in the field of Artificial Embryology, since the required mechanisms for self-assembly and -organization resemble the developmental processes occurring in natural embryology. Another argument for this approach was given by Hornby and Pollack [10], they showed the advantage of generative encodings for Design Optimization over non-generative encodings.

In Artificial Embryology, digital organisms are meant to grow from a single cell, which is able to divide itself, differentiate and develop into an adult form. Like in natural morphogenesis regulatory networks are often used in Artificial Embryology to control the behavior of the individual cells and finally the shape of the complete organism. But to grow complex shapes, a symmetry break has to occur in the initially undifferentiated embryo. In most papers on Artificial Embryology exogenous morphogens or asymmetric cell divisions are used to induce an initial symmetry break.

In this paper we investigate, how a symmetry break can arise endogenously from homogenous cells through evolved regulatory networks. We compare two possible definitions of fitness functions and the performance of several mathematical models for regulatory networks like the activator-inhibitor model, weight matrices, S-systems and general differential equations to generate the head-tail pattern.

In the next section we give a short introduction to the related work in the field of Artificial Embryology. In Sec. 3 we give detailed description of the experimental settings, including the simulation environment, the mathematical models for the regulatory networks and the Evolutionary Algorithm. Experimental results are presented in Sec. 4 together with some graphical examples of the evolved head-tail patterns. Finally, conclusions and an outlook, how to evolve even more complex shapes, are given in Sec. 5.

2. Related Work

Regarding developmental processes two opposing approaches have been suggested on how a complex organism can develop from a single cell. The first approach postulates that the structure is already there from the very beginning. While the second approach assumes that structuring occurs during formation, i.e. epigenesis or self-organization.

Today it is commonly agreed on that self-organization is the true source of morphogenesis. Curiously, in the field

of Artificial Embryology often the path of endogenous self-organization is left in favor for exogenous structuring factors, which, although biologically motivated, actually prevent true understanding for the real mechanisms at work.
For example, one of the earliest works in this area by de Garis, based on a cellular automata, failed to grow non-convex shapes [3] until additional exogenous sources of morphogenes were added [2]. Eggenberger used similar exogenous sources of morphogenes to induce a symmetry break in his digital organisms, which were controlled by a regulatory network [7]. Although such exogenous sources of morphogenes can be supported from examples in nature, like exogenous morphogenetic factors, due to polar bodies, gravity, maternal influences, etc., they actually do not offer a solution to the problem of morphogenesis.
Other approaches utilize asymmetric cell division. For example, Dellaert claimed to give a biologically defensible model of development by evolving a regulatory network based on a random boolean network [4]. In his model an initial asymmetric cell division was implemented as bit-flip in the boolean state vector and cells adjacent to the horizontal midline of the organism were given a special input signal. This way Dellard was able to evolve bilaterally symmetric organisms. Another example that utilizes asymmetric cell division was introduced by Hogeweg [9]. Hogeweg used a sophisticated mechanical model based on the Potts model and was able to evolve very complex cell behavior like budding and engulfing. Again, examples for this kind of asymmetric cell division can be found in nature, like in *C. elegans*, whose first developmental steps have been simulated in detail by Kajita et al. [11]. But as long as the underlying mechanisms of such asymmetric cells divisions are not really understood, the problem of morphogenesis cannot be considered as solved.
There are also examples for self-organizing patterns in reaction-diffusion systems [20]. One example, especially suited for the head-tail pattern, is the activator-inhibitor model by Gierer and Meinhardt [8], which has been utilized by Duvdevani-Bar and Segel to simulate the differentiation of the animal/vegetal region of a digital embryo [6]. Another example by Chaturvedi et al. uses the Schnakenberg equation together with a Potts model to simulate the morphogenesis of a chicken limb [1].
But the problem of how to evolve a mechanism of a priori unknown structure able to induce an initial symmetry break into a formerly undifferentiated embryo has not been addressed. Techniques using exogenous factors will most likely not be able to extend to more complex structures, since this would require more and more additional exogenous factors, which will eventually define the whole body plan. Asymmetric cell division or nuclear determinants on the other hand also do not offer a solution, as long as the responsible mechanism itself is not understood.
Reaction-diffusion systems proved to be reliable mechanisms for pattern formation, but most often only exemplary equation types with known structure and behavior are utilized in Artificial Embryology. It is not clear how this can be extended to more complex structures. Therefore, we examine how reaction-diffusion systems represented by regulatory networks can be evolved, which show the desired behavior of inducing a symmetric break to an undifferentiated digital embryo simply due to noise.

3. Experimental Settings

Although the Activator-Inhibitor model is known to generate the head-tail pattern and parameter optimization on this model could be easily performed, we want to analyze whether we are able to evolve more general regulatory networks with the desired behavior. Therefore, we examine several mathematical models of regulatory networks in respect to their evolvability ranging from parameterized models up to arbitrary differential equations.
In this section we give implementation details on the simulation environment, the regulatory network models, the Evolutionary Algorithms and finally the fitness functions.

3.1. Simulation Environment

Each artificial cell in our simulation environment stores an individual position, a neighborhood topology, the current level of metabolites $x_i \in [0, 10]$, the diffusion behavior of metabolites and the regulatory network controlling the level of the metabolites. In this application the position is not used, since no mechanical simulations are performed and the positional information is not used for any other purpose. The neighborhood topology is used for intracellular diffusion. Regarding diffusion, each metabolite x_i has an individual diffusion rate $D_i \in [0, 0.5]$, which is also evolved by the optimization process. Both diffusion and the dynamics of the regulatory network are subject to random noise.

3.2. Mathematical Models

Several models for regulatory networks have been proposed in the field of Systems Biology, ranging from discrete networks, like boolean and random boolean networks [12], qualitative networks [18], and finally several instances of quantitative networks. Preliminary experiments proved that quantitative networks perform much better and more reliably than discrete networks [17]. Therefore, we limit this study to quantitative networks. The differential equations used were integrated with an Euler-Cauchy algorithm.
In case of parameterized models we give the ranges used for the optimization process, which were set to values similar to those suggested in [19].

3.2.1 Activator-Inhibitor

The following equations give the two dimensional activator-inhibitor model introduced by Gierer and Meinhardt [8]:

$$\frac{dx_1(t)}{dt} = s(\frac{x_1^2}{x_2} + k_1) - r_1 x_1 \quad (1)$$
$$\frac{dx_2(t)}{dt} = s x_1^2 - r_2 x_2 \quad (2)$$

x_1 is the activator, x_2 is the long range inhibitor, $r_1 \in [0,1]$ and $r_2 \in [0,1]$ give the degradation rates of x_1 and x_2, $k_1 \in [0,1]$ gives an independent production rate and the factor s gives a source density.

3.2.2 Weight Matrices

Weight matrices are linear differential equations [5]:

$$\frac{dx_i(t)}{dt} = \sum_{j=1}^{n} w_{ij} \cdot x_j + C_i \quad (3)$$

$w_{ij} \in [-3,3]$ gives the influence of metabolite x_j on metabolite x_i and $C_i \in [-2,2]$ a constant production/degradation rate.

3.2.3 S-systems

Another parameterized model based on differential equations is given by S-systems (*synergistic* and *saturable* systems) [15]. An S-system for n artificial genes is given by a parameterized set of nonlinear differential equations:

$$\frac{dx_i(t)}{dt} = \alpha_i \prod_{j=1}^{n} x_j(t)^{\mathcal{G}_{i,j}} - \beta_i \prod_{j=1}^{n} x_j(t)^{\mathcal{H}_{i,j}} \quad (4)$$

where x_i is the level of metabolite i. With $\alpha_i \in [0,2]$ and $\beta_i \in [0,2]$ the first product describes all synthesizing influences and the second product all degrading influences. Depending on the values of $\mathcal{G}_{i,j} \in [-3,3]$ and $\mathcal{H}_{i,j} \in [-3,3]$ the influence may be inhibitory, if the value of $\mathcal{G}_{i,j}$ or $\mathcal{H}_{i,j}$ is smaller than zero, or excitatory, if greater than zero.

3.2.4 Arbitrary Differential Equations

An example for non-parameterized quantitative networks are arbitrary systems of differential equations, which are more powerful and flexible to describe the interactions between metabolites, since a suitable structure for the regulatory network is a priori unknown. The most prominent method, which is able to optimize the structure and the parameters of general mathematical equations at the same time, is Genetic Programming [13]. Genetic Programming has also been used to model regulatory networks [14].

3.3. Evolutionary Algorithm

To search for regulatory networks that produce the desired head-tail pattern, we applied Evolutionary Algorithms (EAs) a stochastic population-based search algorithm. EAs mimic the natural evolutionary process of repeated selection, reproduction and mutation. Depending on the underlying mathematical model we either applied Evolution Strategies (ES) [16] for parameter optimization of parameterized models, like the activator-inhibitor model, weight matrices and S-systems, or Genetic Programming (GP) [13] for the arbitrary differential equations.

3.3.1 Evolution Strategies

Specialized on real-valued parameter optimization, ES omit a redundant genotype/phenotype mapping and apply sophisticated mutation operators on the real-valued phenotype. The crossover operator is often a secondary evolutionary operator and the population sizes are small. ES use a deterministic selection scheme to select μ parents for the next generation, either selecting only from the λ offsprings, i.e. (μ, λ)-strategy, or by selecting from the pooled offsprings and their μ parents, i.e. an elitist $(\mu + \lambda)$-strategy.

3.3.2 Genetic Programming

Genetic Programming is an extension of Genetic Algorithms and acts on a genotype able to represent computer programs. A common genotype is based on a tree representation using functional elements as nodes and inputs and numerical constants as leafs of the tree. Mutation and crossover operators act on the genotype by altering a subtree or exchanging subtrees between two individuals. For GP typically a generational population strategy is used together with elitism and bigger population sizes [13]. We use a genotype of n program trees to represent the right hand side of an n-dimensional system of arbitrary differential equations.

3.4 Fitness Functions

The fitness function needs to be orientation invariant, since the pattern relies on noise during simulation. The most straightforward implementation of a fitness function for the head tail-pattern is to set a threshold value for an arbitrary metabolite, use this indicator to differentiate a cell to head or tail cells, and count the correctly assigned cells, see the left picture of Fig. 1. For fitness case A1 the inverse of this sum is to be minimized. This function could be easily extended to include the French flag problem [22, chap. 1.13]. The French flag problem aims at segmenting an array of cells into three equally sized partitions of three different cell types colored blue, white and red.

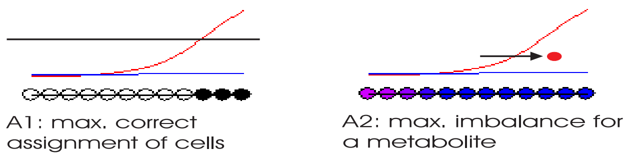


Figure 1. The different fitness cases, using the pattern generated by the activator-inhibitor model as reference solution.

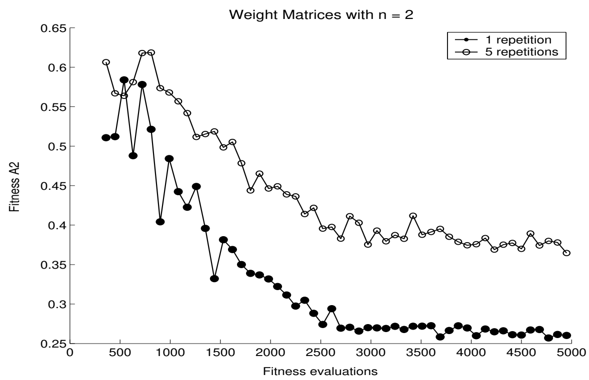


Figure 2. Effect of repetitions on fitness values, normalized by the number of repetitions.

A second variant checks for asymmetry by calculating the weighted center of gravity of a metabolite and taking the difference to the true center of gravity, see right picture of Fig. 1. The inverse of this value is to be minimized in fitness case A2.
Preliminary experiments with a target gradient produced unsatisfying results, because the EAs often got stuck in local optima. But we will continue to search for more suitable fitness functions especially on behavior based fitness functions. Instead of abstract fitness functions we want to simulate the organisms in more complex environments and examine if they are able to perform a given task, like collecting light from a directed light source.
Finally, experiments indicated that random initialization and noise in the simulation made fitness assignment rather difficult. If only one simulation is performed, a risky strategy, which relies on a favorable initial distribution, might beat a more conservative but reliable solution. Further, if the risky strategy multiplies in the population, the chance that a single instance gets suitable starting conditions is increased and thus strengthening the risky strategy. Therefore, a limited amount of repetitions would favor risky strategies. This effect can be counterbalanced by multiple repetitions per fitness evaluation, which causes the EA to favor more reliable strategies, see Fig. 2 for a comparison.

4. Experimental Results

We compared the performance of four different models for regulatory networks on two different fitness cases on two problem instances. The first problem instance simulated a static organism of twelve neighboring cells. The second problem instance represented a dynamic environment simulating a growing organism. Starting with a single cell, every fifth simulation step a random cell duplicated, until the maximum number of twelve cells was reached. In all cases the mechanical simulation was disabled and the cells remained linear aligned for easier fitness assignment.
For parameterized network models we used a (μ, λ)-ES with $\mu = 50$, $\lambda = 150$ to enable self-adaption with discrete crossover ($p_c = 0.5$) and k-dimensional normal distributive mutation ($p_m = 1.0$), where k is the number of decision variables. In case of the GP, we used an elitist generational population strategy with a population size of 250, tournament selection ($t_{group} = 10$), one-point crossover ($p_c = 1.0$) and mutation ($p_m = 0.5$) on each subtree. We used 10 ephemeral constants and x_i as inputs and $\{+, -, \cdot, \%\}$ as function set. In case of GP, diffusion rates where encoded as program trees only with constants as leafs.
Beside the impact of the different mathematical models for regulatory networks and their evolvability, we further examined the effect of the number of metabolites involved. Our previous experiments showed that the number of metabolites can have a crucial impact on the evolvability of a regulatory network [17].
For each problem instance we performed 25 multi-runs, except for GP, where only 10 multi-runs were performed. Every optimizer was terminated after 10,000 fitness evaluations had been performed. Each fitness evaluation was repeated five times and simulated the digital organism for 250 time steps each trial, and the fitness was normalized by the number of simulation steps, the number of cells and the number of repetitions.
It is necessary to mention that the original version of the activator-inhibitor model requires at least 10,000 simulation steps to establish the head-tail pattern. In our evolutionary framework we were bound to fall short of that amount of simulation effort. But we believe that the reduced amount of simulation effort complies with the requirements necessary to evolve much more complex structures with a much higher number of cells involved.

4.1. Static Environment

In the first test series we tried to establish the head-tail pattern in an adult 'fully' grown organism of twelve interconnected cells.

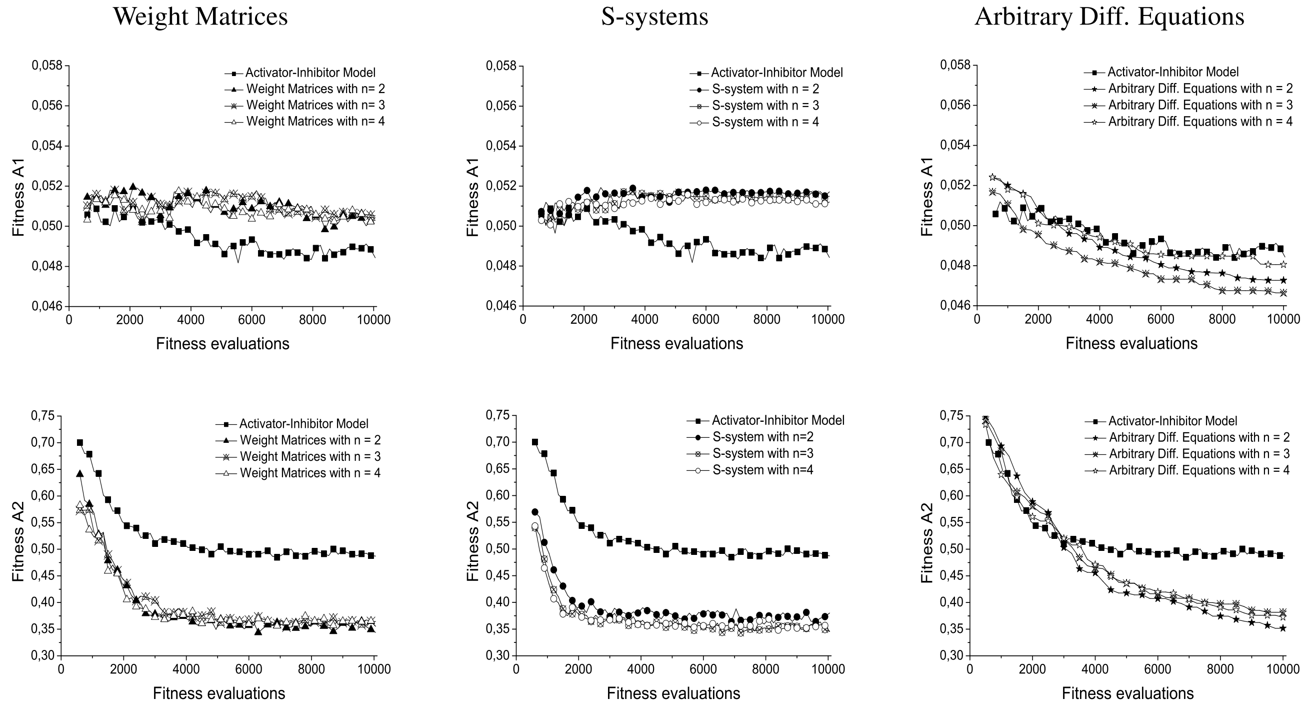


Figure 3. Results on A1 and A2 in a static environment.

4.1.1 Pattern Test

On the most straight-forward implementation of the fitness function, the A1 function, we see that both the weight matrices and the S-systems fail to produce satisfying results, see Fig. 3. For these two models the resulting regulatory networks simply produced random alternating patterns. These results are not affected by the number of metabolites involved.

With the activator-inhibitor model on the other hand we were able to find solutions, which resulted in compact areas of activation at the borders of the organism, see Fig. 4. In the figure one can see four different trials with different random seed simulating the same regulatory network, parameters are also given in Fig. 4. In almost all repetitions a clear segmentation occurred within 250 simulation steps.

Arbitrary differential equations were best performing, regarding the fitness values are. They even outperform the activator-inhibitor model. This could be due to the different selection scheme used for the GP, which imposes a much higher selection pressure. But also due to the bigger flexibility compared to the parameterized models.

4.1.2 Deviation from the Center of Gravity

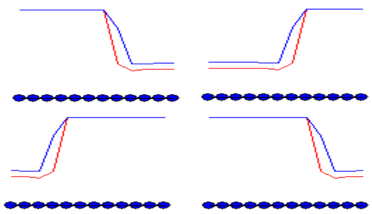
In case of maximizing the deviation of the center of gravity weighted by a metabolite from the true center of gravity the activator-inhibitor model is clearly outperformed by the alternative models, see Fig. 3.

When comparing the models for regulatory networks, the GP optimizing arbitrary differential equations is the slowest to converge, despite the higher selection pressure applied, and S-systems slightly outperform weight matrices in respect to the speed of convergence.

In case of S-system one can also see an effect regarding the number of metabolites involved. With $n = 3$ and $n = 4$ the EA is able to find slightly better solutions than in case of only $n = 2$ metabolites.

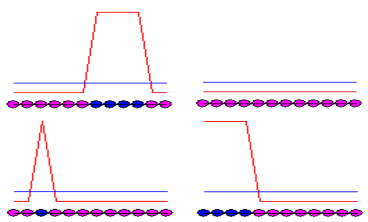
But regarding the quality of the solutions found the results are rather disappointing. The EA seems to favor risky strategies over more reliable ones despite the high number of repetitions used. In Fig. 5 one can see an exemplary result for S-systems. Most strategies are based on amplifying the initial noisy distribution of metabolites to the maximum concentration level. This way, the fitness values are bigger since the weights for calculating the weighted center are maximal. This particular strategy utilizes an extremely low diffusion factor of the metabolites that is actually used for fitness calculation to locally confine the peaks and achieves much higher fitness values this way. Although this solution achieves high fitness values, it is so specialized that it actually does not solve the head-tail pattern, because this strategy is too unreliable.

To summarize, on the static environment only the



$$\frac{dx_1(t)}{dt} = s(\frac{x_1^2}{x_2} + 0.540) - 0.437x_1$$
$$\frac{dx_2(t)}{dt} = sx_1^2 - 0.929x_2$$
with $D_1 = 0.099, D_2 = 0.087$

Figure 4. Four exemplary simulations for an activator-inhibitor based regulatory network.



α_i	$\mathcal{G}_{ij}$	
1.635	1.304	0.765
1.134	2.605	0.042

β_i	$\mathcal{H}_{ij}$	
1.771	-2.271	1.548
1.474	-0.136	-1.994

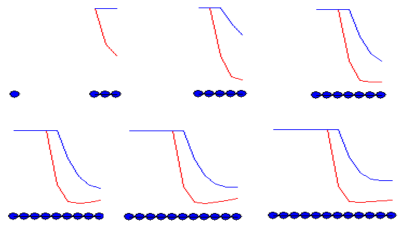
with $D_1 = 5.975E - 4, D_2 = 0.164$

Figure 5. Four exemplary simulations of an S-system based regulatory network.

activator-inhibitor model is able to identify satisfying results, although the alternative models achieve better fitness values. This indicates that the fitness functions or the environment may be not suitable to support the evolution of the desired head-tail pattern.

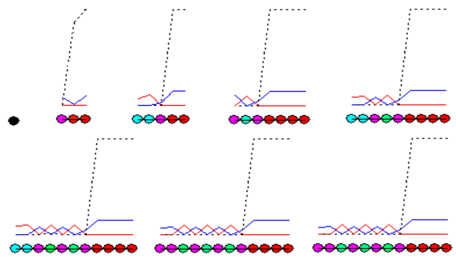
4.2. Dynamic Environment

To increase the performance and to make the simulation environment more realistic we implemented a dynamic problem instance where the digital organism grows from a single cell to a final size of twelve cells. This is done by splitting a random cell every five simulation steps until the maximum number of cells is reached. For the sake of simplicity we coded this behavior of limited growth by hand instead of evolving this behavior as we did in previous experiments [17]. When a cell splits, we half the metabolites for the two resulting cells, assuming that the two cells im-



$$\frac{dx_1(t)}{dt} = s(\frac{x_1^2}{x_2} + 0.634) - 0.351x_1$$
$$\frac{dx_2(t)}{dt} = sx_1^2 - 0.108x_2$$
with $D_1 = 0.064, D_2 = 0.366$

Figure 6. Exemplary simulation for an activator-inhibitor based regulatory network, on a growing organism.



C_i	w_{ij}		
1.522	-2.738	-0.746	-1.863
-0.739	1.653	-0.896	0.213
1.135	-2.205	-2.657	1.184

with $D_1 = 0.063, D_2 = 0.337, D_3 = 0.344$

Figure 7. Exemplary simulation for a weight matrix based regulatory network, on a growing organism.

mediately grow to the original cell size.

We consider this dynamic environment to be more interesting for two reasons. First, this environment resembles the actual task given in Artificial Embryology, where the simulation of a growing digital organism is dynamic, not static. Second, we tried to examine a feasible variant of asymmetric cell division. Instead of performing a perfect cell division by giving both cells an equal share of the metabolite concentration, we implemented a noisy cell division. If we were able to evolve regulatory networks that utilized this asymmetric cell division based on noise, we would have a biologically defensible model for asymmetric cell division since such noise could be caused by discrete distribution of metabolites within a cell.

The choice of a reasonable noise level is critical for the model to be really biologically defensible. On the one hand,

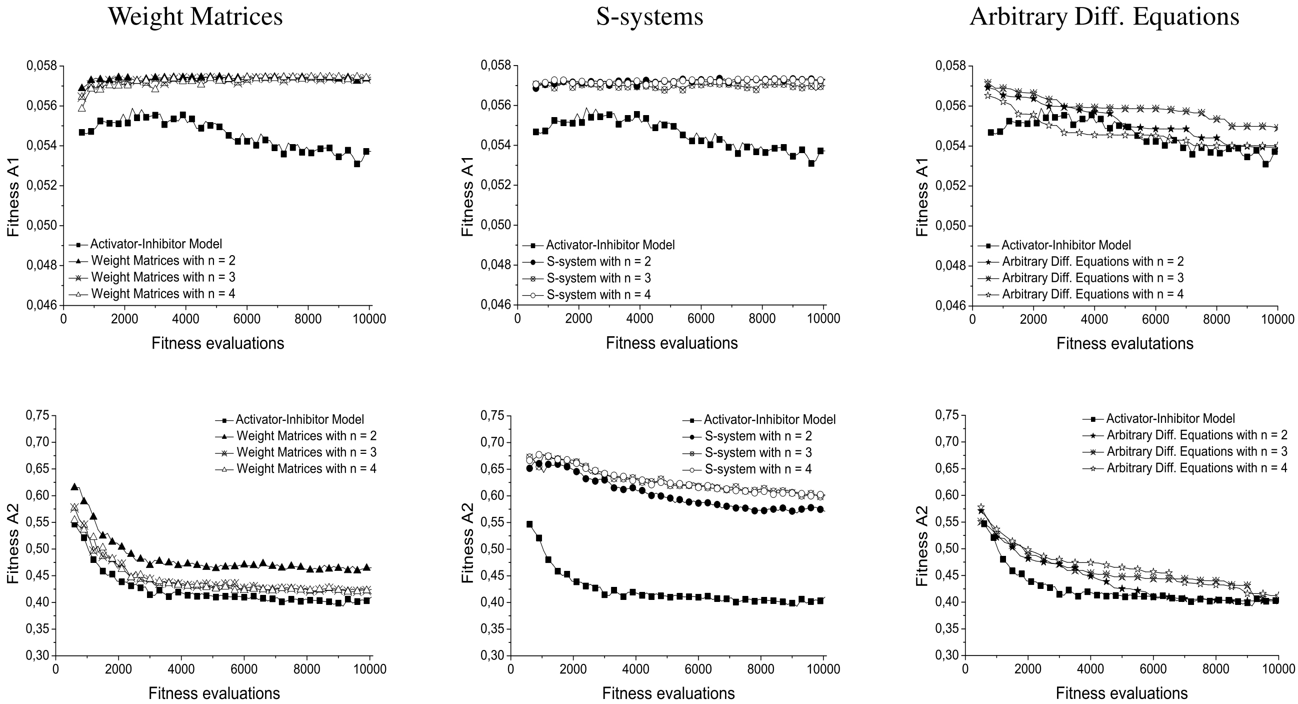


Figure 8. Results on A1 and A2 in a dynamic environment.

beautiful results can be achieved, if the noise acts like an XOR operator as used together with discrete regulatory networks in [4, 9]. But on the other hand such an operator can not be accounted for by simple noise anymore. To keep reasonable we applied gaussian noise with $\sigma = 0.01$ to the two resulting cells in the following experiments.

4.2.1 Pattern Test

On the A1 fitness function all parameterized regulatory network models fail, see Fig. 8. Only the activator-inhibitor model and the GP were able to evolve behaviors, which lead to the head-tail pattern. But the patterns are very unreliable, similar to those in Fig. 5. A positive example with parameters for an activator-inhibitor model is given in Fig. 6.
It is possible that a higher noise level for cell division would cause the EA to find more reliable regulatory networks and that the noise level applied is too small.

4.2.2 Deviation from the Center of Gravity

On the A2 fitness function the activator-inhibitor model performs best regarding the fitness values, and GP and weight matrix based regulatory networks perform better than the S-system based networks, see Fig. 8. But with increasing number of metabolites the weight matrix based networks are able to equal the performance of the activator-inhibitor models. Interestingly enough the increased number of metabolites has an opposite effect on S-system based regulatory networks. Here the performance decreases with in increasing number of metabolites, most likely due to the increased dimension of the search space.
An positive example for weight matrix based regulatory networks is given in Fig. 7, which is not as sensitive to the initial conditions as the activator-inhibitor example given in sec. 4.2.1, and which leads to a stable head-tail pattern with an additional stripe pattern in the tail. This network uses the second cell division to identify the head. Those two cells are marked as head, same as all offsprings of these cells. The ratio between the number of cells belonging to the head and to the tail depends on the occurrence of the cell division.

5. Discussion

In this paper we have shown that the head-tail pattern can be evolved utilizing general mathematical models for regulatory networks in contrast to specialized models with an a priori fixed network structure. This gives the prerequisite for successful applications of Artificial Embryology in the area of Evolvable Hardware as discussed in Sec. 2.
Further, we proved that regarding the evolvability the GP is superior to two parameterized models. Unfortunately, GP is the most CPU-intensive model and the resulting regulatory networks are the least comprehensible. Comparing

the parameterized models, the weight matrices only performed better than S-systems on the A2 function within the dynamic environment. Regarding the number of metabolites involved, two metabolites seemed to be sufficient on all problem instances except for the weight matrix on A2 in the dynamic environment, where at least three metabolites were necessary.

And finally, we could point out that the choice of the fitness function and the environment is crucial for the success of the evolutionary process. The most straightforward A1 implementation of the fitness function showed the worst results. While the A2 performed much better especially in case of the dynamic environment, see the result in Fig. 7. The difference between A1 and A2 can be accounted to the fact that A2 offers a much smoother path to an optimum, where small deviations from symmetry can be gradually exploited. In case of the A1 function solutions resulting from pure noise prevent the EA to find a possible path to the real head-tail pattern. Comparing the static to the dynamic environment, the solutions of the general regulatory networks found in the dynamic environment were much more reliable than those found in the static environment. This proves that the evolutionary process was able to exploit the small asymmetric cell division used in the dynamic environment.

Our future research will concentrate on evolving more complex patterns on dynamically growing cell clusters. One way to build more complex structures could be gene duplication [21]. We hope that by gene duplication already discovered functional subnetworks could be multiplied in the genome and through mutation be activated in different regions within the organism and therefore resulting in more complex patterns.

References

[1] R. Chaturvedi, J. A. Izaguirre, C. Huang, T. Cickovski, P. Virtue, G. Thomas, G. Forgacs, M. Alber, G. Hentschel, S. A. Newman, and J. A. Glazier. Multi-model simulations of chicken limb morphogenesis. In *Multi-model simulations of chicken limb morphogenesis*.

[2] H. de Garis. Artificial embryology - the genetic programming of cellular differentiation. In *Artificial Life III Workshop*, Santa Fe, New Mexico, USA, June 1992.

[3] H. de Garis. Artificial embryology: The genetic programming of an artificial embryo. In B. Souček, editor, *Dynamic, Genetic, and Chaotic Programming*, pages 373–393. John Wiley, New York, 1992.

[4] F. Dellaert and R. Beer. Toward an evolvable model of development for autonomous agent synthesis. In R. Maes and P. Maes, editors, *Artificial Life IV*. MIT Press Cambridge, 1994.

[5] P. D'haeseleer, X. Wen, S. Fuhrman, and R. Somogyi. Linear modeling of mRNA expression levels during CNS development and injury. *Pacific Symposium on Biocomputing*, 4:41–52, 1999.

[6] S. Duvdevani-Bar and L. Segel. On topological simulations in developmental biology. *Journal of Theoretical Biology*, 131:33–42, 1994.

[7] P. Eggenberger. Evolving morphologies of simulated 3d organisms based on differential gene expression. In P. Husbands and I. Harvey, editors, *Proceedings of the 4th European Conference on Artificial Life*, Cambridge, MA., 1997. The MIT Press.

[8] A. Gierer and H. Meinhardt. A theory of biological pattern formation. Kybernetik 12, 30-39, 1972.

[9] P. Hogeweg. Evolving mechanisms of morphogenesis: on the interplay between differential adhesion and cell differentiation. *Journal of Theoretical Biology*, 203:317–333, 2000.

[10] G. S. Hornby and J. B. Pollack. The advantages of generative grammatical encodings for physical design. In *Proceedings of the 2001 Congress on Evolutionary Computation (CEC2001)*, pages 600–607, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, 27-30 2001. IEEE Press.

[11] A. Kajita, M. Yamamura, and Y. Kohara. Computer simulation of the cellular arrangement using physical model in early cleavage of the nematode caenorhabditis elegans. *Journal of Bioinformatics*, 19(9):704–716, 2003.

[12] S. Kauffman. *The Origins of Order*. Oxford University Press, New York, 1993.

[13] J. R. Koza, David Andre, F. H. Bennett III, and M. Keane. *Genetic Programming 3: Darwinian Invention and Problem Solving*. Morgan Kaufman, Apr. 1999.

[14] E. Sakamoto and H. Iba. Inferring a system of differential equations for a gene regulatory network by using genetic programming. In *Proceedings of Congress on Evolutionary Computation*, pages 720–726. IEEE Press, 27-30 2001.

[15] M. Savageau. 20 years of S-systems. In E. Voit, editor, *Canonical Nonlinear Modeling. S-systems Approach to Understand Complexity*, pages 1–44, New York, 1991. Van Nostrand Reinhold.

[16] H.-P. Schwefel. *Evolution and Optimum Seeking*. John Wiley & Sons, New York, 1995.

[17] F. Streichert, C. Spieth, H. Ulmer, and A. Zell. Evolving the ability of limited growth and self-repair for artificial embryos. In *Advances in Artificial Life - Proceedings of the 7th European Conference on Artificial Life*, pages 289–298, 2003.

[18] D. Thieffry and R. Thomas. Qualitative analysis of gene networks. *Pacific Symposium on Biocomputing*, 3:77–88, 1998.

[19] D. Tominaga, M. Okamoto, Y. Maki, S. Watanabe, and Y. Eguchi. Nonlinear numerical optimzation technique based on genetic algorithm for inverse problem: Towards the inference of genetic networks. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 251–258. Morgan Kaufmann, 2000.

[20] A. M. Turing. The chemical basis of morphogenesis: A reaction-diffusion model for development. *Philosophical Transactions, Series B: Biological Sciences*, 237:37–72, 1952.

[21] D. Whitehead. Personal communication. Goettingen, Germany, 2003.

[22] L. Wolpert. *Principles of Development*. Oxford University Press, New York, 2002.