Towards an evolution model of multiagent organisms

Grégory Beurier, Fabien Michel, Jacques Ferber

Towards an evolution model of multiagent organisms

Gregory Beurier[1], Fabien Michel[2], and Jacques Ferber[1]

[1] Laboratoire d'Informatique, de Robotique et de Microlectronique de Montpellier (LIRMM) 161 rue Ada, 34392 Montpellier Cedex 5 - France
beurier,ferber@lirmm.fr
[2] CReSTIC/LERI - IUT de Reims-Châlons-Charleville
rue des crayres BP 1035 51687 REIMS Cedex 2 - France
fabien.michel@univ-reims.fr

Abstract. The study of evolving artificial organisms to build hardware or software complex systems is a promising research track. In this paper, we describe a bio-inspired method that makes agents evolve to form multiagent organisms. This method mimics the functioning of *Hox genes* in the development of natural embryos and implements techniques from the evolutionary computing field. Furthermore, we study the possibilities of our approach developing multiagent systems that evolve and self-organize in various flag patterns.

1 Introduction

Many of the systems that surround us are complex. Understanding their properties motivates much if not all of scientific inquiry. That is why the study of complex systems has become recognized in recent years as a scientific discipline [1], maybe one of the most promising interdisciplinary fields. The field of involved scientific inquiries covers philosophy, chemistry, physics, computer science, etc.

The study of living organisms, their behavior and evolution, is one of the rapidly developing areas in the study of complex systems. Living systems outperform in many ways complex systems that man has produced. Seeking inspirations from these systems, there are several established and active research communities organized around themes such as self-organization [2], artificial immune systems [3], DNA computing [4], artificial neural networks [5], morphogenesis [6], evolutionary computing [7], etc. Among all these research tracks, we are especially interested in the following questions:

- How a single cell can grow through division and differentiation to a multicellular organism?
- How evolution can lead to functional and behavioral adaptation to environment?

Both questions highlight the mechanisms of complexity appearance in the living systems. How a single evolved cell can give birth to complex organisms

? To answer these questions, evolutionary computationists explore three main ways to grow artificial embryos from a single cell (process named embryogeny) [8] :

- External: the growth is hand-designed and organisms do not evolve.
- Explicit : the growth is defined in data structures and organisms evolve to "learn" growing process.
- Implicit: the growth is the result of interacting behaviours/rules and organisms evolve to "learn" these behaviours/rules.

This paper proposes an implicit embryogeny to explore possibilities of evolving a multiagent system [9]. Seeking brainwawes in biology, the interacting behaviors of agents are directly inspired by the functioning of genes involved in the morphogenesis process. Agents replicate and interact to contruct organisms following morphogenetic rules.

Morphogenesis is one of the fundamental aspects of developmental biology along with the control of cell growth and cellular differentiation. Morphogenesis is concerned with the shapes of tissues, organs and the spatial distribution of cells that arises during the embryonic development of an organism.

To achieve the learning of growing behaviour, agents evolve. In biology, evolution is a change in the traits of living organisms over generations, including the emergence of new species. The study of evolution processes has leaded to the development of many population-based metaheuristic optimizations methods. Genetic algorithms [10], evolution strategies[11], genetic programming[12], evolutionary programming [13] are usually quoted. The goal of such methods aims to evolve artificial population of individuals possessing a genetic inheritance (genotype, genome) to explore a problem search-space. The solution of the problem is thus given by an interpretation (program, subroutine), direct translation (behaviour, value, result), etc., of the genetic inheritance of the individual (genotype, genome).

The work presented in this paper is motivated by an attempt to explore the capabilities of a developmental multiagent model that encompasses the notion of "pluri-individual" evolution.

The outlines of the paper are as follows: Section 2 presents a review of some relevant works on embryogeny. In Sect. 3, a brief biological background is given in order to explain the morphogenetic process. The multiagent system model is given in Sect. 4: we first define the agents that mimic embryo cells and then describe the evolutionary model of agents that construct organisms. Simulations of the model are developed in Sect. 5 and discussed in Sect. 6. Finally, we draw conclusions from this work in Sect. 7.

2 Related works

A number of researchers have implemented morphogenetic models in order to produce realistic phenomena. One of the earliest works returns to Turing (Reaction/Diffusion) [14]. Turing proposed linear equations to achieve spatial differentiation, which is done by postulating two substances with mutual interaction and

different rates of distributions in space, namely the *morphogens*. Since Turing studies, the morphogen term is usually employed referring to substances involved in patterning processes.

In the field of Artificial Embryology, de Garis was one of the first to present a work based on cellular automata and genetic programming [15]. De Garis's work achieved to grow simple non-convex organisms until the addition of external sources of chemical gradients. Exogenous sources of substances or morphogens are commonly used in research on morphogenesis. For instance, Eggenberger used exogenous morphogens to induce a symmetry break in his digital organisms [16].

Fleischer and Barr [17] developed a simulation framework for multi-cellular pattern formation including chemical diffusion. This framework also includes mechanical factors such as cell adhesion, genetic factors, etc. Fleischer and Barr showed the difficulty of maintaining the size and shape of a multi-cellular organism and pointed out the necessity of combining multiple mechanisms in the pattern formation to guarantee the robustness of the organism. COMPUCELL 3D has been developed to simulate morphogenetic processes in different organisms and also implements phenomena such as cell growth, diffusion of chemical gradients, etc [18]. COMPUCELL 3D highlighted the necessity to model each cell state of an organism to implement differentiation processes. Still, the morphogenesis model of COMPUCELL 3D is limited by the fact that it a priori assumes the shape of the organism as a function of time.

Developing artificial neural networks(ANNs) is a great challenge in researches on morphogenesis. A number of researchers have studied the potential of Lindenmeyer systems (L-systems) [19] for generating ANNs.[3] Boers and Kuiper used evolutionary algorithms to evolve the rules of L-systems that generate ANNs[20]. Gruau abandoned L-systems to develop a language called *cellular programming* based on graphs transformations[21]. Cellular programming enables the controlling of the division of cells that grow into ANNs.

Miller developed artificial organisms based on Cartesian Genetic Programming[22]. Miller's goal is to extend classical Cartesian Genetic Programming to evolve a cell that can build a larger program by iterating the cell's program in its environment. Each cell becomes a part of the program which is literally the organism as a whole. Miller's method permits to build enormous programs that would have required very large amount of information to be specified. Still, this method seems to be expensive in resources for simple problems for which developing a single program is easier.

3 Biological background

3.1 Background

In order to describe our bio-inspired model, we now introduce the biological concepts we will use later on. Pattern formation is a key process in embryonic

[3] Due to their fractal properties, L-systems are often used for both modeling and visualizing plants.

development[23]. During this development, a fertilized egg gives rise via division to various complex structures (body, organs, etc.). Three main processes can be distinguished:

- Cell differentiation: the cells acquire specific functionality/type such as muscle, neuron, cartilage, etc.
- Morphogenesis: the cells become "aware" of their spatial position.
- Growth: the cells divide and the organism grows.

Our approach relies on the functioning of genes involved in the morphogenetic process. Morphogenesis has been first studied via the observation of alteration of developmental processes in species. Indeed, Bateson defined Homeosis in 1894 to describe natural variants of species where certain parts of the body plan exhibited morphological features typical of other regions. At the end of the 19^{th} century, Lewis discovered few mutants drosophila (a fruit fly) which part of the body was transformed into another one (See fig. 1). Lewis then spoke about homeotic alterations. Since the end of the 20^{th} century, the genes involved in these alterations are known. Those are the same which are involved in the morphogenesis process, the *Homeobox* genes, or *Hox* genes.

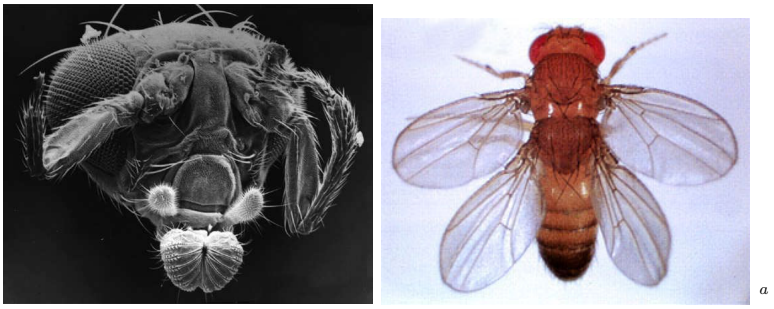

Fig. 1. (1): Antennapedia (legs on the place of antennae) - (2): Bithorax (two thoraxes instead of one).

[a] Photos from http://www.snof.org.

3.2 Genes and *Hox* genes

Genes are regions of DNA inheritance of living creatures. A gene basically encodes the chemical structure of a protein and the way this protein is produced. A protein is produced by a cell when the *coding part*(CP) of the gene is transcribed. The transcription process depends on *regulatory elements*(REs) which are other parts of the gene. Some kinds of molecules (proteins, RNA[4], etc.) named *transcription factors* are able to interact with the REs to:

[4] Ribonucleic acid.

– Activate or increase the gene transcription.
– Repress or decrease the gene transcription.

Transcription factors may come from the inside (internal) or the outside (external) of the cell. The mechanism of activation/repression is called the *regulation system* (See Fig. 2). When a protein is involved in the transcription of a gene α, this protein *regulates* the gene α. In the same way, a gene encoding this protein *regulates* the gene α. *Hox* genes are part of the homeobox gene family. A

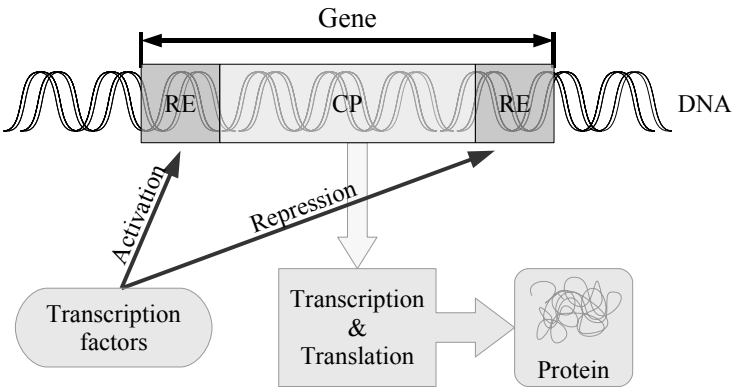

Fig. 2. The regulation system.

homeobox gene encodes a particular protein called *homeodomain protein*. Due to its chemical properties, Homeodomain proteins are transcription factors for a number of genes, especially for structural genes. Homeodomain proteins usually act with other transcription factors, such as homeodomain proteins, to form chemical complexes. *Hox* genes specifically function in patterning the body axis, especially the antero-posterior axis (axis from head to feet). Thus, by providing the identity of particular body regions via regulation of numerous genes, *Hox* genes determine where limbs and other body segments will grow in a developing foetus or larva for many species. Because of their role in the developmental process, homeodomain proteins can be considered as *morphogens*.

3.3 Drosophila larva development

In order to better understand the role of *Hox* genes, we will now describe, at a high level of abstraction, the early phases of drosophila larva development. For further information on this subject see [24]. The drosophila oocyte (the female gamete) is polarized by gradients of maternal molecules which are mRNAs[5]. It is

[5] Maternal ribonucleic acids.

important to note that mRNAs have deep effects on the development of a fertilized egg, but they are expressed by cells within the maternal ovary. Within the fertilized egg, these mRNAs are translated into proteins which are transcription factors. Those factors regulate a first set of *Hox* genes. This process induces the formation of new gradients of proteins within the cell. Those gradients regulate new genes creating new gradients, and so on (See Fig. 3). Rapidly the egg is segmented in zones by numerous proteins gradients. Then, those proteins regulate new genes, the homeotic genes. The homeotic genes are Hox genes which regulate groups of functional/structural genes. For instance, drosophila *labial* and *deformed* homeotic genes encode proteins that are expressed in head segments. Within head segments, these proteins activate the genes defining head features. If this transcription factor is damaged, the resulting individual can be a mutant(See Fig. 3.1).

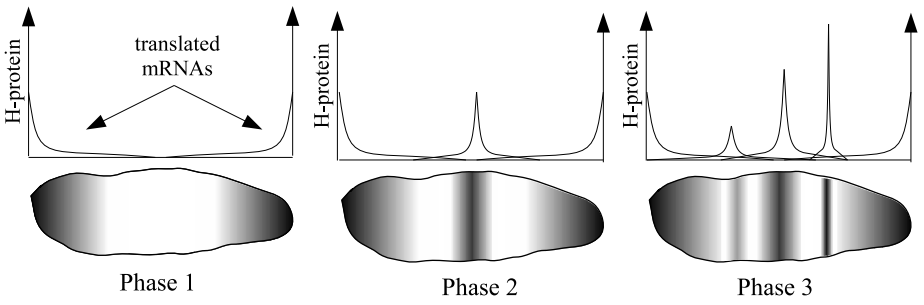

Fig. 3. Gradients of homeodomain proteins during the development of drosophila larva.

It is interresting to note that due to a considerable genetic redundancy, the vertebrates Hox genes has been highly conserved during the Evolution. For instance, there are strong similarities between Hox complexes of tilapia, pufferfish, striped bass, zebrafish, horn shark, human, and mouse; species which are separated by approximately 500 million years of evolution[25].

4 Model

The proposed MAS model is directly inspired by the functioning of hox genes in the morphogenetic process. Our approach aims to closely mimic this biological process:

- Environment takes the role of egg/embryo, in which maternal gradients of morphogens are diffused.
- Agents take the role of cells and their behavior is an interpretation of a genome composed by Hox genes and behavioural genes. They are called *cell-agent*.

4.1 Environment and information

The environment is a discretized rectangular surface (8-connexity) representing either a toroidal world or a closed world. The environment acts primarily as an interaction medium. Gradient of maternal substances are initially present to initiate the developmental process. These gradients are placed randomly or in an ad-hoc manner, depending on the organism adaptation objectives. The maternal gradients assume the role of biological mRNAs (cf. Sect. 3.3) as in natural morphogenesis. De Garis achieved to grow convex shapes with such an addition of external morphogens [15]. The evolution of the system is discrete and can be described as follows: Let $S_{(t)}$ be the system state at time t, which is characterized by a set ψ of agents in the environment E:

$$S_{(t)} = (E_{(t)}, \psi) \quad \rightarrow \quad S_{(t+1)} = (E_{(t+1)}, \psi) \tag{1}$$

The information management is comparable to a biological plain diffusion. This management involves equalizing the concentration of proteins/substances within the environment. Amount of proteins varies each step in each space unit as follows: (1) proteins importation from the 8-neighbors space, (2) proteins export towards the 8-neighbors space, (3) proteins evaporation, (4) drop of proteins by agents.

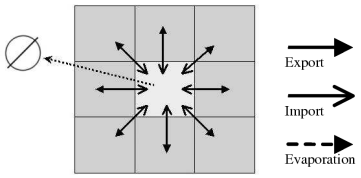

Fig. 4. Diffusion model in discretized environment.

The *coefficient of diffusion* and the *evaporation rate* are the plain diffusion parameters for every kind of proteins. Various kind of plain diffusion algorithms have been used without affecting the behaviour of the model.

4.2 Agents

Our approach extends the classical models using logical regulatory networks by distinguishing morphogenetic interactions (*Hox genes*) and functionnal interactions (*Behavioural genes*). In our model cell-agents are reactive. Each cell-agent has a genome (set of genes) which describes the agent's behaviour. There are two types of genes:

- *Hox genes* which are involved in the patterning process of the organism.
- *Behavioural genes* that encode the agents' behaviour.

A gene is composed by two parts: (1) *the coding part*(CP) and (2) *the regulatory elements*(REs). The CP encodes the primary function of the gene. The REs encode the interaction properties that enable the regulation of the gene. If a gene is correctly regulated, it expresses the function encoded in its CP.

The CP of *Hox* genes encodes the properties and the emitted quantity of a particular homedomain protein. The protein properties are the diffusion model parameters. When a *Hox* gene is activated, a diffusing protein is produced with corresponding evaporation rate and diffusion coefficient. The *Hox* gene CP is implemented in a set of binary digits, namely a bitset. The CP can easily be extended to add extra properties to the produced proteins (e.g. type, colour, etc.).

In the case of behavioural genes, the type of the CP depends on the nature of the implemented function. Indeed, the CP works as an evolutionary computing standard tool. For instance, the CP can be a tree encoding a part of a program (like in genetic programming[12]), a bitset encoding a specific parameter/value (like in genetic algorithm[10]), a graph, etc. The CP of behavioural genes encodes a behavioural/physical characteristic of the agent (e.g. size, reproduction behaviour, etc.).

The REs encode the interaction of the gene with the proteins gradients in environment to enable the regulation process. The formal description of REs is inspired by the Reaction/Diffusion model[14] (See Fig. 5). The REs encode a

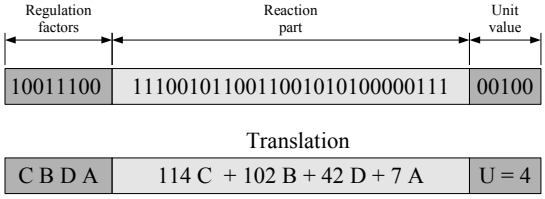

Fig. 5. Example of regulatory elements.

	Simulation 1		Simulation 2		Simulation 3	
Proteins	Quantity	Reactions	Quantity	Reactions	Quantity	Reactions
A	0	0	1000	35	336	12
B	800	1	2448	6	8000	19
C	200	0	6000	13	22000	48
D	2620	15	2000	11	2500	168
Global result	0		6		12	

Table 1. Examples of reactions of the regulatory elements from Fig. 5 with various quantities of homeodomain proteins.

chemical reaction involving the transcription factors[6] of the gene. First of all, a bitset encodes which proteins are the transcription factors of the gene. These transcription factors are randomly chosen at the beginning of the simulation (namely C, B, D, A in Fig. 5). The second part of the REs, called the *reaction part*(RP), encodes the chemical reaction. The RP is decomposed into equal parts representing reaction factors with each of the transcription factors. The last part encodes the reaction unit quantity, an integer. The REs functions as follows: An agent perceives proteins in its space unit. An integer quantity is associated to each kind of perceived proteins. If this protein is a transcription factor of a gene, the chemical reaction of this gene RP is applied: let Q be the quantity of perceived transcription factors, Rf the value of the reaction factor for this transcription factor, U the unit value and Nr the result of the reaction. (See Fig. 1 for examples).

$$Nr = Q/(Rf * U) \tag{2}$$

Method	Simulation 1	Simulation 2	Simulation 2
Standard	Inh / null	Act / HQ	Act / HQ
Treshold 1	Inh / null	Inh / null	Act / HQ
Treshold 2	Inh / null	Act / HQ	Inh / null
Function	Inh / null	Act / 240	Act / 480

Table 2. Reactions results with various reaction methods. Value are taken from Table 1. Top: Activation result for behavioural genes (activated - Act, inhibited - Inh), Bottom: Quantity of proteins emitted for *Hox* genes (HQ - Quantities encoded in Hox gene).(1) Standard (2) Treshold 1: min:8, max treshold:14. (3) Treshold 2: min treshold:1, max treshold:6. (4) Function: Quantity emitted = number of reactions $*$ unit $*$ 10

.

In *standard* reactions, if Nr is higher than 1 for all the transcription factors of the gene, the regulatory elements activate the gene (See line 1 of table 2). REs can be extended to perform more sophisticated or realistic tasks. Actually, the role of the RP is limited to the activation of the CP. Some experiments are led with a second RP. Instead of activating the CP like the first RP does, this second RP inhibits it. This permits to increase the way a gene can be regulated by considering transcription factors as activators and inhibitors at the same time. Another extension deals with activation/inhibition *thresholds*. The quantities of perceived transcription factors may be high enough to react several times (See line 2-3 of table 2). So, a part can be added to REs to limit the regulation to a certain number of reactions: under or over (or both) a number of reactions

[6] The transcription factors are the diffusing homeodomains proteins.

encoded in the gene, the CP will be inhibited. This principle is fundamental in the *Hox* genes functioning. *Hox* genes encode diffusing proteins that are transcription factors. The quantity of these emitted proteins is implemented into the CP of the gene. To model more sophisticated phenomena, this coding can be modified by calculating the quantities of emitted proteins as a *function* of the number of reactions of the RP (See line 4 of Table 2).

4.3 Genetic model

The evolutionary principles of the model are inspired by classical approaches of the evolutionary computing field. The evolutionary algorithm has been extended to encompass the notion of organism. Every agent possesses a behavioural gene encoding the replication function (*mitosis*). *Hox* genes are generated randomly, and behavioural genes depend on designer objectives.

The evolutionary algorithm works as follows:

1. Generate maternal gradients in the environment
2. Generate a population of agents with a random genome
3. Assign fitness to each agent according to objective function on the formed organism
4. Select n individuals according to fitness for reproduction
5. Reproduce offspring by taking two parents at a time and using reproduction operators
6. Apply mutation operators on the offspring
7. Evaluate each offspring and assign fitness
8. Replace old population by offspring according to replacement strategies
9. Unless stopping criterion reached return to step 4

Step 1 is a trivial one: the number of generated gradient depends on the organism adaptation objectives. Gradients generation can be either done randomly or by the simulation designer (in order to accelerate the population convergence. See Sect. 5.1).

Step 2 consists in randomly generating coding parts and regulatory elements for every genes of the agents. This generation can be different according to the nature of the gene (i.e. bitsets, tree of instructions, graphs of values, etc.).

The third step and the seventh step are crucial ones. They consist in simulating each single agent in the pre-generated environment and evaluating the quality of the formed organism. The fitness is given by external(s) observer(s) based on the designer desiderata. It is important to note that fitness function(s) are computed on formed organisms but is assigned to the single agent that has generated the whole organism.

Step 4 involves selection methods of evolutionary computing. Selection consists in choosing the agents that are allowed to reproduce and then transmit their characteristics to the next generation. A lot of methods can be used there. Roulette Wheel Selection, Tournament Selection, Rank Selection, Boltzmann selection are usual selection criteria [12].

Step 5 consists in creating new agents (with new genomes) from pre-selected agents. The methods are related to the genes type. For instance, a crossover operator is a standard method for binary encoded genes. The offspring are generated in the hope that they will be better (in the sense of fitness) than their parents. Partial Mapping crossover, Order crossover, Cycle crossover, Edge Recombination crossover, Edge Assembly crossover are examples of crossing techniques [12].

Step 6 is used to alter genetic information on genes. This operation is used to prevent the evolutionary algorithm from stagnating at local optima. Mutation is dependant on the gene type. For instance, a mutation on a binary gene consists in a probabilistic switching of one or more digits.

Step 8 determines which offspring will replace old agents in order to generate the new population. Elitism, steady state replacement, CHC selection, are techniques commonly used for this purpose [12].

The evolution algorithm is a classical model of evolutionary programming. This algorithm has been extended to fits our embryologic/organism point of view. The genome of an individual neither encodes the solution of a problem nor a program for solving a problem but rather encodes the growth of an organism which would have selected properties to resolve the considered problem.

5 Evolutionary experiments

The model has been implemented on the MAS platform MadKit [26] using the TurtleKit framework [27]. TurtleKit is a simulation engine which provides tools for exploiting multiagent simulations based on agents evolving in a discretized world. TurtleKit also provides tools for plain diffusion management.

In this paper, two experiments are analyzed in order to present step by step the two types of genes. In both experiments, we explore the possibilities of the model by creating organisms that construct flag patterns.

Some modifications have been done to the model to ensure a faster convergence of the system. First of all, in both experiments, the maternal gradients have been deposited according to symmetry axis: bottom-up, up-bottom, left-right, right-left. In natural embryos, these gradients are emitted by the mother during the early phase of the development in order to bootstrap the morphogenetic process. The gradients do not diffuse nor evaporate: the perception of these proteins by agents remains unchanged during simulation. In sect. 6, we discuss the possibility of designing experiments without maternal gradients. Secondly, coding parts of some genes have been simplified by defining in an ad-hoc manner some behaviours or proteins properties: it permits to ignore evolution process on such part and quicken the global evolution of the population.

In both experiments, the fitness of agents is given by external observers and agents eligible for reproduction are chosen via a roulette wheel selection tournament. The fitness consists in a percentage of similarity with a pre-defined pattern. The maximum fitness value is 100 when the formed organism matches perfectly with the pre-defined pattern. Agents are simulated one by one and placed in the center of the environment in order to facilitate evaluation.

5.1 French flag

The French flag model of Wolpert [28] has been the inspiration for the first task the model has to achieve. This model has already been studied by Miller [29] using CGP. The evolved organism has to grow a recognizable French flag.

In order to exhibit the role of maternal gradients, the first system has been implemented without hox genes. All agents' genes are directly regulated by proteins gradients initially present in the environment. Agent's genome consists in four genes: *mitosis*, *blue*, *white* and *red*. The *mitosis* gene controls the duplicating behaviour of agents. If the *mitosis* gene is activated, then agents duplicates in the free neighbor spaces. If the colour genes are activated, then agents take the colour corresponding to the gene. By default, agents are green. There is an order relation between colour genes: $blue > white > red$.

Several simulations have been made using 30 individuals as a population. Figure 6 shows the growth of French flag organisms. The convergence of population is obtained with an average of 250 generations[7].

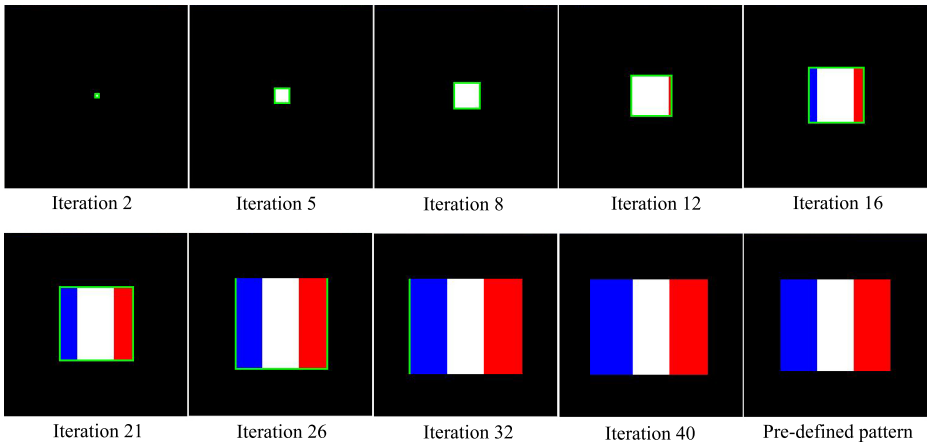

Fig. 6. Growth of fittest program from a single agent to a mature French flag organism.

5.2 Japanese flag

The second system has been implemented to exhibit the role of hox genes in the patterning process. Moreover the Japanese flag organism has been chosen as a more complex organism. Indeed, circular shapes seem to be a difficult task to achieve in embryologic developmental approaches. Agent's genome consists

[7] Data taken from 25 simulations.

of three genes: *mitosis*, *Hox* and *red*. By default, agents are white. The same environment (size, maternal gradients) than in French flag experiments has been used. It permitted us to reuse the mitosis gene of evolved French flag agents. Indeed, the mitosis gene specifically controls the size of the formed organism. We used a Japanese flag pattern of the same size as the French flag pattern. It is an important feature of the model: by reusing evolved genes, the performance of the evolution algorithm can be significantly increased. This notion is discussed in sect. 6.

Several simulations have been made using diverse size of population. Figure 7 shows the growth of Japanese flag organism.

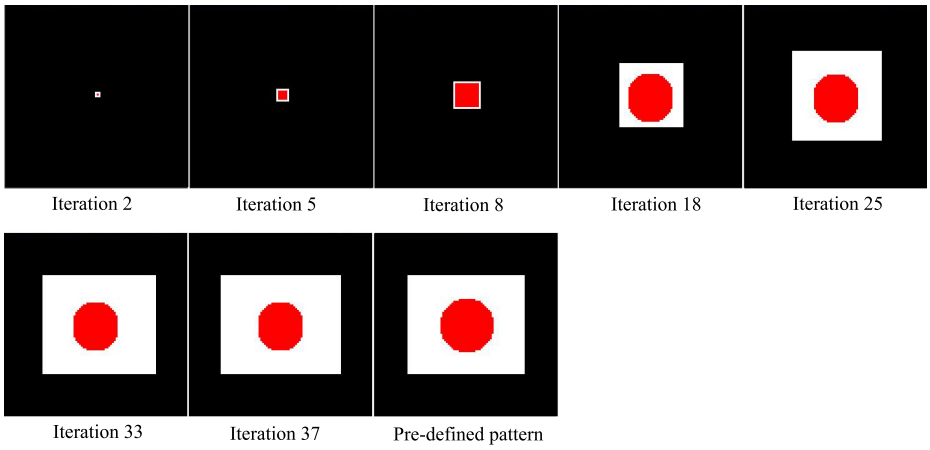

Fig. 7. Growth of fittest program from a single agent to a mature Japanese flag organism.

The formation of the red circle intervenes when (1) *Hox* gene encodes a homeodomain protein which is a regulation factor of the *red* gene, (2) this homedomain protein has fittest parameters, (3) the *red* gene encode an adequat reaction with the homeodomain protein. The convergence of 30-individuals population is obtained with an average of 200 generations[8]. The simulations without *mitosis* gene reusing showed convergences with an average of 290 generations.

6 Discussion

The experiments exhibited the potentiality of the model. This work is in its first stage and there are many ways in which the system can be improved. First of all, the model can be improved to be more realistic by adding, for instance, adhesion

[8] Data taken from 25 simulations.

laws[9] between agents or chemical interactions between proteins. However, since the aim of the work is toward technological applications, it is important to keep the model as simple as possible.

In real biological development, cells acquire functionalities during the growing process. This is done by stem cells that gradually differentiate. One could imagine mimic this process in order to better control growth of organism. Indeed, in the model agents are stem cells that duplicate without specifying activated functions in offspring. It can be interesting to fix functional cells in order to produce new behaviours in the evolution process.

Secondly, we are working on substituting maternal gradients by the agents' behaviour themselves. To apply this research on real-world problems, we need to compensate the lack of external signals presence by a better local control. To this end, we believe that Reaction/Diffusion techniques can be useful. Indeed, Gierer and Meinhardt [30] have shown that interactions between heterogeneous gradients of morphogens can lead to symmetry breaks and polarity gradients formations. Nevertheless, it is our conviction that the maternal gradients are fundamental in the natural embryogenic process and cannot be ignored in order to design sophosticated artificical embryogenies. At the same, we are exploring some modifications of the gene reaction part: the chemical reaction is extended to a function of transcription factors. The preliminary experiments showed great improvements in the pattern refining and symmetry breaking of simulated organisms.

Another great improvement of the model concerns the evolution process. In Sect. 7, we discussed the possibility of reusing evolved genes in order to improve evolutionary convergence. We plan to incorporate this technique into the evolutionary engine to facilitate the design of systems by programmers. In the same way, we plan to modify the mutation operators in order to permit modification of the genome structure of agents. In fact, following the establishment of new sources of gradients in a pattern as a consequence of *Hox* genes regulation, new gradients may be fired leading to more refined patterns. Nevertheless the refining can be insufficient and lead to local minima. So, we plan to introduce the possibility of adding new *Hox* genes during the evolution process in order to ensure a better refining of complex organisms.

7 Conlusion

We have presented and discussed a bio-inspired developmental model of multiagent organisms. Bentley and Kumar demonstrated [8] that an evolutionary approach of embryogeny provides significant benefits in evolutionary computation. Indeed, evolutionary models seem to be easier to design in a way to evolve complex solutions. Our embryogeny model is closely inspired by the morphogenetic process and uses a multiagent paradigm. The presented multiagent model involves the evolution of reactive agents sensitive to proteins gradients.

[9] Chemical or mechanical laws that postulate adhesion between agents.

The system has been implemented in order to exhibit its capabilities in forming complex organisms.

Perspectives of such a work are various, but we are actually interested in three issues. The first one deals with complex systems design and especially multiagent systems design. We plan to construct swarm organisms giving the formed organism functions to resolve problems which require distributed resolution. Another interesting research track deals with data encryption and data compression. It can be interesting to consider pre-defined patterns of organisms as a data to compress or to encrypt. Evolved genomes would be thus considered as compressed data which have to evolve to be readable. In the same way, we can imagine to encrypt data in a genome. Then, the relevant key to decrypt data could be either the environment properties or the agents' behaviour.

Finally, we claim that a general model with a high level of abstraction and simple principles can contribute to a better understanding of real biological development processes. Moreover, the multiagent paradigm seems to be a relevant approach to model closely bio-inspired complex systems. It is our conviction that seeking inspiration from biology can help to enrich our understanding of the functioning of complex systems.

References

1. Waldrop, M.: Complexity: The emerging science at the edge of order and chaos. Touchstone, New York, USA (1992)
2. Camazine, S., Deneubourg, J.L., Franks, N.R., Sneyd, J., Theraulaz, G., Bonabeau, E.W.: Self-Organization in Biological Systems. Princeton University Press (2001)
3. Hightower, R.R., Forrest, S., Perelson, A.S.: The evolution of emergent organization in immune system gene libraries. In: Proceedings of the 6th International Conference on Genetic Algorithms, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1995) 344–350
4. Boneh, D., Dunworth, C., Lipton, R.J., Sgall, J.: On the computational power of dna. Discrete Appl. Math. **71** (1996) 79–94
5. Yao, X.: A Review of Evolutionary Artificial Neural Networks. Commonwealth Scientific and Industrial Research Organization., Victoria, Australia (1992)
6. Roggen, D., Floreano, D., Mattiussi, C.: A morphogenetic evolutionary system: Phylogenesis of the poetic circuit. In: ICES. (2003) 153–164
7. Koza, J.R.: Genetic programming. In Williams, J.G., Kent, A., eds.: Encyclopedia of Computer Science and Technology. Volume 39., Marcel-Dekker (1998) 29–43
8. Kumar, S., Bentley, P.J.: Computational embryology: past, present and future. Advances in evolutionary computing: theory and applications (2003) 461–477
9. Ferber, J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley Longman Publishing Co., Inc. (1999)
10. Holland, J.H.: Genetic algorithms and classifier systems: Foundations and future directions. In: ICGA. (1987) 82–89
11. Rechenberg, I.: Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Frommann-Holzboog, Stuttgart (1973)
12. Koza, J.R.: Introduction to genetic programming. In Kinnear, Jr., K.E., ed.: Advances in Genetic Programming. MIT Press, Cambridge, MA, USA (1994) 21–42

13. Fogel, L., Owens, A., Walsh, M.: Artificial Intelligence Through Simulated Evolution. John Wiley, New-York (1966)
14. Turing, A.M.: The chemical basis of morphogenesis. Philosophical Transactions of the Royal Society (B) **237** (1952) 37–72
15. de Garis, H.: Artificial embryology and cellular differentiation. In Bentley, P.J., ed.: Evolutionary Design by Computers. Morgan Kaufmann, San Francisco, CA (1999) 281–295
16. Eggenberger, P.: Evolving morphologies of simulated 3d organisms based on differential gene expression (1997)
17. Fleischer, K., Barr, A.H.: A simulation testbed for the study of multicellular development: The multiple mechanisms of morphogenesis. In Langton, C., ed.: Proceedings of the 3rd Workshop on Artificial Life, Addison-Wesley (1992) 389–416
18. Izaguirre, J.A., Chaturvedi, R., Huang, C., Cickovski, T., Coffland, J., Thomas, G., Forgacs, G., Alber, M.S., Hentschel, G., Newman, S.A., Glazier, J.A.: Compucell, a multi-model framework for simulation of morphogenesis. Bioinformatics **20** (2004) 1129–1137
19. Prusinkiewicz, P., Lindenmayer, A.: The algorithmic beauty of plants. Springer-Verlag New York, Inc., New York, NY, USA (1990)
20. Boers, E., Kuiper, H., Happel, B., Sprinkhuizen-Kuyper, I.: Designing modular artificial neural networks. In Wijshoff, H., ed.: Proceedings of Computing Science in The Netherlands, SION, Stichting Mathematisch Centrum (1993) 87–96
21. Gruau, F.: Genetic synthesis of modular neural networks. In: Proceedings of the 5th International Conference on Genetic Algorithms, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1993) 318–325
22. Miller, J.F.: Evolving developmental programs for adaptation, morphogenesis, and self-repair. Advances in Artificial Life. 7th European Conference on Artificial Life **2801** (2003) 256–265
23. Wolpert, L., Beddington, R., Jessell, T., Lawrence, P., Meyerowitz, E., Smith, J.: Principles of Development 2nd edition. Oxford University Press (2002)
24. Li, H., Harrison, D., Jones, G., Jones, D., Cooper, R.L.: Alterations in development, behavior, and physiology in drosophila larva that have reduced ecdysone production. The Journal of Neurophysiology **85** (2001) 98–104
25. Santini, S., Boore, J.L., Meyer, A.: Evolutionary conservation of regulatory elements in vertebrate hox gene clusters. Genome Research **13** (2003)
26. Gutknecht, O., Ferber, J.: MadKit: Organizing heterogeneity with groups in a platform for multiple multi-agent systems. Technical Report 97188, LIRMM, 161, rue Ada - Montpellier - France (1997)
27. Michel, F.: Introduction to turtlekit: A platform for building logo based multi-agent simulations with madkit. Technical Report RR LIRMM 02215, LIRMM, 161, rue Ada - Montpellier - France (2002)
28. Wolpert, L.: The french flag problem: A contribution to the discussion on pattern development and regulation. In: Towards a Theoretical Biology. Volume 1. Waddington (1968) 125–133
29. Miller, J.F., Banzhaf, W.: Evolving the program for a cell: from french flags to boolean circuits. In Kumar, S., Bentley, P.J., eds.: On Growth, Form and Computers. Academic Press (2003)
30. Gierer, A., Meinhardt, H.: A theory of biological pattern formation. In: Kybernetik. (1972)