FRENCH FLAG TRACKING by MORPHOGENETIC SIMULATION UNDER DEVELOPMENTAL CONSTRAINTS

Abdoulaye Sarr[1], Alexandra Fronville[1], Pascal Ballet[1], and Vincent Rodin[1]

Université de Brest,
UMR CNRS 6285, Lab-STICC, CID, IHSEV,
Computer Science Department,
20 avenue Le Gorgeu, 29200 Brest, France

Abstract. Below the influence of the mechanical cues and genetic expression, constraints underlying the developmental process play a key role in forms emergence. Theses constraints lead to cells differentiation and sometimes pinpoint the directions of cells growth. To better understand these phenomena, we present in this paper our work focused primarily on a development of a mathematical model taking into account the *co-evolution* of cellular dynamics with it's environment. To study the influence of the developmental constraints, we have developed algorithms to make and explore a base of genomes in order to check conditions under which specific genes are activated and some patterns emerge to achieve a specific shape. From our model, we found a genome that can generate the *French flag*. With this *French flag* pattern and its genome starting, we addressed the following question : is there another genome in the simulated base that achieves the same shape, i.e. the *French flag* pattern ?

Keywords: mathematical modelling, simulation of biological systems, morphogenesis, multi-agent system, French flag problem

1 INTRODUCTION

1.1 MORPHOGENESIS : EMERGING of INTERESTS

Biomedical science has undergone a remarkable evolution during this last decade. Advances and innovations in biotechnology, more particularly in microscopy and imaging, have provided a large amount of data at all level of biological organization. In 2007, Melani and al. achieved a tracking of cell nuclei and the identification of cell divisions in live zebra fish embryos using `3D+time` images acquired by confocal laser scanning microscopy [11]. These kind of data allowed new description in details of many components and structures of living organisms. Observations noticed from these data like geometrical segmentations during cells proliferation have raised relevant issues in mathematical and numerical point of view. But experimental complexity usually restricts observations to a single or very restricted spatial or temporal scales. So, one of the purposes of mathematical and computational models in biology is to reconstruct from this large amount of data, gathered at different scales, integrated models that take into account the dynamic interactions between different levels of the biological organization. In cells proliferation models, it means to consider the cell as a place of integration of causalities and downgrades.

For this purpose, *tensegrity model* consider biomechanical forces between cells and the extracellular matrix. The stretching of cells adhering to the extracellular matrix may result from local reshuffle in this latter. According to this model, growth-generated strains and pressures in developing tissues regulate morphogenesis throughout development [8]. It is therefore the biomechanical forces which play a key role in this model of morphogenesis. For example by modulating cell differentiation, influencing the direction of division or deforming tissues. However, the question of cell diversity even arises before the acquisition of shape [13]. Indeed, when the embryo has only a few pairs of cells, we can already see a diversification of biochemical content or a diversification of embryonic cells morphology. That may be the result of genetic and molecular interactions. Indeed, the emergence of forms also stems from the acquisition of differential properties, of cell mobility and gene expression throughout cell development.

Moreover, *Artificial Regulatory Networks* also allow modelling morphogenesis. They define a series of regulatory genes and structural genes. The firsts consist of a network of rules determining the evolution of the system and the latter are intended to each generate a simple specific pattern. They can be seen as a dynamic system following different trajectories in a state space [9]. However, even if the detailed knowledge of genomic sequences allow to determine where and when different genes are expressed in the embryo, it is insufficient to understand how the organism emerge [12].

1.2 BELOW GENETIC EXPRESSION

So, we have looked forward to learn more about the emergent properties of cells organization and especially the importance of cellular dynamics in the emergence and evolution of shapes, both in mathematical and numerical point of view.

A Multicellular organism is a complex system which can be defined as a composition of a significant number of elements interacting locally to produce a global behaviour. Complex systems are also characterized by a high capacity of self-adaptation and self-organization, such as multicellular organisms are. They can evolve and learn through feedbacks between their external environment and their internal architecture. And according to Doursat [4], whether inanimate structures or living organisms, all processes of form emergence are instances of decentralized morphological self-organization. When cells evolve, they modify their organism which in its turn impacts their behaviour. This is what biologists mean by *co-evolution*. Epigenetics considers that this coupling between organism and environment can not be ignored in understanding the development of living organisms [15].

In mathematics, the *viability theory* [1] offers concepts and methods to control a dynamical system in a given fixed environment, in order to maintain it in a set of constraints of viability. Applied to morphogenesis, this means that we should have at least one co-viable evolution of the cells state and their environment based on each state-environment pair. This formalization allows us to establish feedback rules in terms of direction growth changing and cells differentiation. This mathematical model was formalized in [6]. So in our computational model, every cell has its own rules monitored by a set of controls and stay aware of its neighbourhood. This ensures cells to be autonomous while knowing when the dynamic stops being viable and consequently when to implement feedback mechanisms. The integration of the behaviours of each component of the system allows to determine its global state in a finite time.

In this paper, we are going first to present our morphogenesis simulation tool (`Sect. 2`). Then, to better study and understand the influence of shape emergence on cellular dynamics, we present here original algorithms based on the mathematical model to explore and simulate a base formed by all possible genomes from a same set of genes. The aim of this exploration is to verify the conditions under which certain genes are activated and especially if these conditions are unique for a targeted form (`Sect. 3`). We test the algorithms on our morphogenesis simulation tool with the *French flag* as being the form to reach (`Sect. 4`). Finally, we conclude before highlighting some relevant applications and future prospects we could give to them (`Sect. 5`).

2 SIMULATION TOOL

In this section, we describe `Dyncell` which is a tool developed for simulating generative systems [5]. The platform was created to experiment our theoretical model of morphogenesis.

The program is implemented in `C++` using a tool kit of Virtual Reality : `ARéVi` [14, 3]. `ARéVi` is a simulation library of autonomous entities with a `3D` rendering developed at *European Center of Virtual Reality* (Brest, France).

2.1 ARCHITECTURE

The architecture of our simulation tool is based on the concept of shape. All classes inherit from a generic parent class called `Form`. During simulation, each instantiated object is a form by definition.

This class has two main subclasses: `Environment` and `Cell`, which are the two types of objects to be instantiated before starting any simulation.

Running a simulation always consists of instantiating an object `Environment` and some `Cells` objects evolving in this environment.

Class `Cell` is responsible for representing cells. Then, an object `Cell` is defined by its *age*, *color*, *division speed* and *energy amount.* During mitosis, another `Cell` object is created next to the mother `Cell` object that triggered the division according to a route choice as defined in the mathematical model.

Environment is a `Form` object which has a lifetime. It is used to delimit the space in which cells population operates or to exert external forces on them.

2.2 FEATURES

The order of scheduling has a significant impact on the results of the simulation [10, 2]. It determines how local interactions (self-organization) have been held, consequently how and when the system reaches the final shape. Different behaviours can be observed in virtual models depending on the scheduling mode. When modelling natural systems, asynchronism seems more appropriate. In morphogenesis, for instance, synchrony is likely to cause false correlations between cells. In ARéVi, scheduling of agents is handled implicitly in asynchronous and stochastic way. Routines of each agents are executed sequentially and entirely, the same number of times. However the running order of agents is randomly determined.

We set as a basic principle that cells are autonomous agents and ignorant of the whole system even if they can perceive their environment and adapt appropriately. They are represented on screen by spheres and can proliferate in a discrete environment (cellular automaton) or in a "continuous" one. In the latter case, the movement of cells is more precisely described and we can have more complex interactions.

A graphical interface has been implemented in order to set the options of the simulation. It allows dynamic change of parameters and selecting mechanisms (e.g. apoptosis) that will be active/inactive during the simulation. Options are available to allow choice between 2D/3D, discrete or continuous simulations. The size and shape of both the environment and the cells can also be defined and adjusted.

Spatial constraints are crucial for evolution of the cells. If they are too heavy, the cell is not viable as it can no longer divide. A maximal constraint parameter sets up a threshold below which the cell can remain viable.

To account for the influence of the environment, a parameter is defined as the maximum number of cells that a cell is able to push when it divides. When the current strain of the cell is greater than the maximum stress threshold, the cell can no longer divide. To stay alive, a cell can undertake two modes of mitosis which are feedback mechanisms to face this kind of constraints. Firstly, the cell chooses to divide in the direction where the stress is less intense; secondly where the direction of cell division is predetermined.

It is also possible to assign an amount of energy to each cell. The basic idea is to consider that a cell has a store of energy assimilate from its environment. A percentage of the store is used to maintain structure and growth. And the remaining part is used for maturation and reproduction. The cell dies when the energy level becomes too low.

3 ALGORITHMS

At the initial stage of a living organisms creation, all cells have the same genome. But during the developmental process, they don't all express the same genes. Why the dynamics of these cells have changed to bring about a differentiation ? What are the factors that come into play ? What is the mechanism by which this takes place ?

These are the questions that underlie any understanding of collective self-organization and self-adaptation mechanisms that target then achieve the well-guided form of living organisms. To address them, it will be important to consider the influence of the developmental process on the overall shape and the underlying constraints that it imposes locally to the cells. The idea of the algorithms we have developed to study this influence is greatly inspired by the concept that Waddington introduced in 1940 as *the epigenetic landscape* [16]. All possible paths through this landscape (whose barriers are similar to the constraints of the form) represent the possible forms from the same genome, but subject to different conditions in which only some specific genes are activated (see Fig. 1). Therefore, the identification of the obstacles across the landscape allows to know where and when feedback mechanisms are involved (i.e differentiation that gives a particular character to the form at this time).

To study the effect of the developmental process constraints, we verify the conditions under which feedback mechanisms are triggered. To do so, we have generated a base of genomes formed from the same set of genes. Having a genome that achieves a given shape, we aim to explore the entire space of possible genomes to see if we can find another genome which allows to reach it.

In this paper we have chosen to target the well known *French flag*. That means we seek a genome, when simulated, will give a form whose development constraints will lead :

- to the activation of the same genes as the *French flag* (same colors),
- to the choice of same segmentation directions as the *French flag* (same shape).

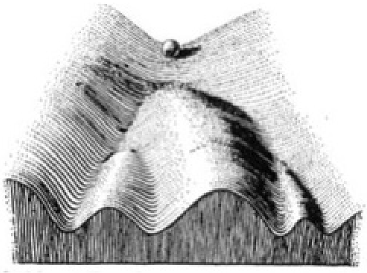


Fig. 1. In 1957, Conrad Waddington proposed the concept of an epigenetic landscape to represent the process of cellular decision-making during development. At various points in this dynamic visual metaphor, the cell (represented by a ball) can take specific permitted trajectories, leading to different outcomes or cell fates [7].

In Waddington's point of view, that implies that from two different origins on the top of the *epigenetic landscape*, the ball can go through different trajectories but encounter the same obstacles and finally hit the same endpoint.

3.1 SOFTWARE ARCHITECTURE

In our software architecture, we have two main classes which handle exploration and simulation activities.

1. `Genome` : represents a genome and allow us to manage all treatments that can be performed on a genome. As we designed it, a genome is composed of one or more genes that are activated or not during the simulation.
2. `GenerateGenome` : includes all the activities of exploration, initialization, launching and stop. It carries initialization method of the exploration parameters, start and stop genome simulation methods, methods allowing treatment of genomes, methods for managing a set of genomes, methods for recording exploration results.

3.2 GENOMES BASE CONSTRUCT

Genes are defined in an `xml file` with all their attributes (*color coded*, *route choice applied*, *number of repetitions in the genome* ...). From this file, they are loaded into a `vector` from which we create all possible combinations of genomes based on the characteristics of the *French flag* genome. These characteristics are:

1. its genome is composed of `4` genes
2. the `1st` gene in the genome is the same as the `4th` gene
3. if `n` is the number of times which is repeated the `1st` gene in the genome (called *multiplicity* of the gene), the `2nd` gene has therefore a *multiplicity* of `2xn` and `n` for the `3rd` and `4th` genes.

Thus, all genomes are constructed in the same way (see `Table 1`).

Table 1. *French Flag* Genome Characteristics

Genes	Gene 1	Gene 2	Gene 3	Gene 1
Multiplicity	`n`	`2xn`	`n`	`n`

3.3 INITIALIZATION

During initialization, we define the match criteria to be achieved by simulated genomes in the base. When simulating *French flag* genome, we chose `3` as the first gene order of *multiplicity*. We note that the multiplicity doesn't affect the shape but just its size. The bigger it is, the larger is the shape. And `3` is an optimal value for an exploration of such a large base.

In `Table 2`, we have some informations from different simulations performed with *French flag* genome. For a given *multiplicity* of the *French flag* genome (`row 1`), we have the values of the criteria recorded during the simulation. The *number of cells* that make it up (`row 2`) and the *age* at which it reaches its final form (`row 3`). The current time is determined during the simulation by the scheduler and is expressed as a double.

So, during the exploration, to assess that a given genome has achieved the *French flag*, we mainly refer to these values of these criteria.

Table 2. *French Flag* Genome Simulation Results

Multiplicity	Number of Cells	Age
3	70	1.16
4	117	1.224
5	176	1.288
6	247	1.352

3.4 DESCRIPTION of the ALGORITHM

The goal of the algorithm is to explore all genomes. Each one is simulated in order to check if it is able to reach the *French flag*. In other terms, if it is possible to achieves the same number of cells as the *French flag* in an identical time on a genome constructed in the same way as the *French flag*'s one but with different parameters. If a such genome is found, the algorithm lists it as well as the order of *multiplicity* with which it has converged.

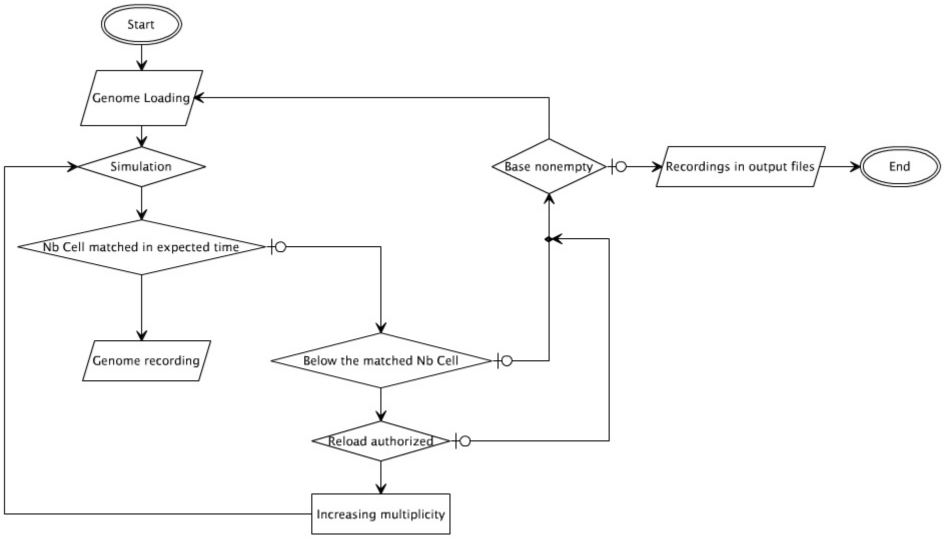


Fig. 2. The Algorithm's flowchart

Apart from a match, there may be two cases (see Fig. 2) :

1. the number of cells exceeds that corresponding to the current choice of *multiplicity*. So we definitely assume that we can't reach the *French flag* with this genome. In this case, the algorithm continues exploring and pick its follower in the base.
2. the number of cells doesn't reach that of the current choice of *multiplicity*. So we will check the correspondence of the same genome led to a higher order of *multiplicity* (i.e from 3 to 4). The algorithm then reload the same genome

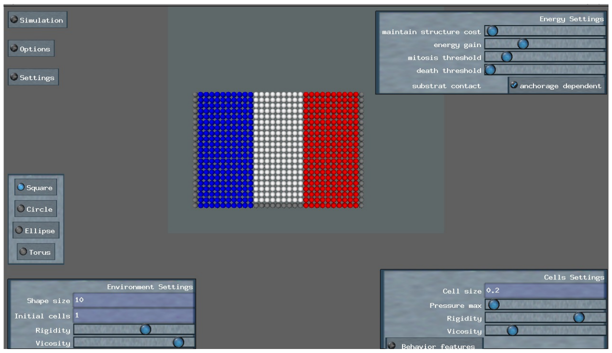
by changing the settings of these genes multiplicity. In the 2nd simulation of the genome, if the number of cells corresponding to this new choice of *multiplicity* is met, then it is listed. Else, if the number of cells exceeds, we execute the same treatment as in **(1)**. Else if during this second simulation of the genome, the corresponding number of cells is still not reached, back to **(2)** : increase *multiplicity* from 4 to 5 in the 3rd simulation of the genome.

While the number of cells keeps being lower than that expected, we redefine genes parameters of *multiplicity* in the genome a maximum number of times allowed. In the algorithm, it is 3 times. So the last choice of *multiplicity* is 6 with an expected number of cell 247 in 1.352 time units. In this step, the genome is no longer reloaded, considered definitely not able to achieve the *French flag*. Then, the algorithm continues exploring and pick its follower in the base.
When the algorithm has finished simulating all genomes, the program stops and we retrieve the results in an output file.

4 EVALUATION

4.1 TEST CONDITIONS

In the tests we have achieved, the targeted form was, as shown in `Fig. 3`, the *French flag* introduced by L. Wolpert in 1969 [17].



substrat contact ☑ anchorage dependent

Shape size 10

Initial cells 1

Cell size 0.2

Fig. 3. Form targeted by genomes in the base

The genome which allows to generate the *French flag* is as follows (details in `Table 3`) : [**1 2 3 1**].

Table 3. Informations carried by the *French flag* genome

Genes	Colour coded	direction indicated in the route choice
1	Blue	Right
2	White	Forward
3	Red	Left

For the 64 genes we have, we obtained 262,114 possible combinations of genomes, making the exploration space pretty huge. Indeed, for the simulation of the entire base, it took 5 days. Simulations were done on a computer with the following characteristics :

- System : Linux 32 bits
- Processor : Intel Core 2 CPU 6300 1.86GHz x 2
- RAM : 2 Go
- GPU : ATI Readon X1300/X1550 Series (256MB)

4.2 OUTPUT RESULTS

5 genomes were identified as generating forms with characteristics identical to those of the *French flag* (see Fig. 4) : [**1 2 4 1**]-[**1 4 2 1**]-[**3 2 1 3**]-[**4 1 2 4**]-[**4 3 2 4**].

They have all achieved it with the first order of multiplicity (i.e. 3). Consequently, they are each composed of 70 cells with a maturation age of 1.16 time units (Ref. Table 2).

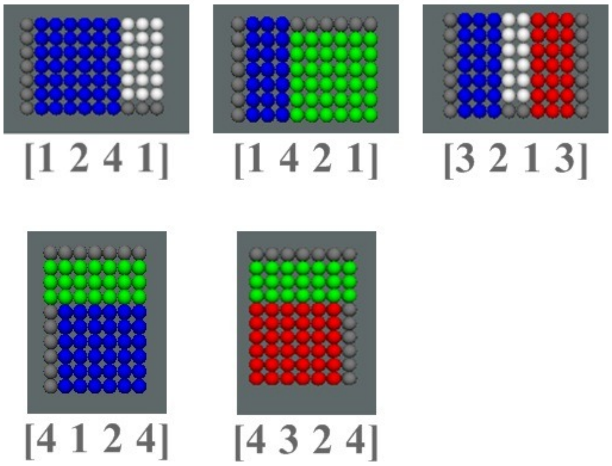
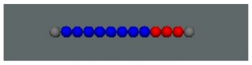


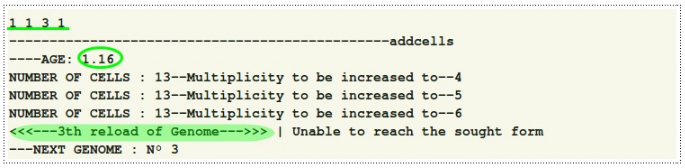
Fig. 4. Forms identified by the algorithm that have met matching criteria of the *French flag*

We note that among these 5 forms retained, there are only 2 of which exactly the three genes of the *French flag* are activated : [**4 3 2 4**]-[**3 2 1 3**]. And only 1 of these 2 is same as *French flag* : [**3 2 1 3**].

So, from two different genomes, we have reached the same form.

Two generated forms discarded by the algorithm are presented in Fig. 5 and 6. The first one present the case where removal is due to the fact that the algorithm has reloaded the 3 times authorized the same genome [**1 1 3 1**] without getting result. Based on the output result of the simulation, we can point out that for these 3 times, the number of cells remain 13. Then, the algorithm continues to the follower genome.

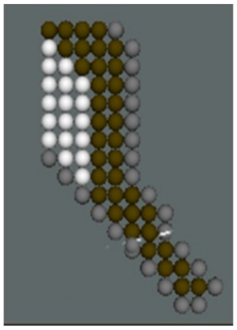


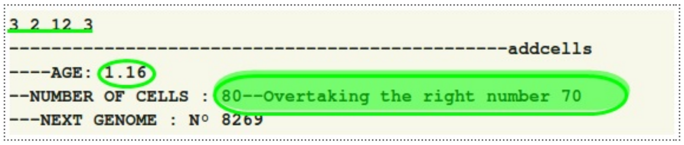


```
1 1 3 1
-----------------------------------------------addcells
----AGE: 1.16
NUMBER OF CELLS : 13--Multiplicity to be increased to--4
NUMBER OF CELLS : 13--Multiplicity to be increased to--5
NUMBER OF CELLS : 13--Multiplicity to be increased to--6
<<<---3th reload of Genome--->>> | Unable to reach the sought form
---NEXT GENOME : N° 3
```

Fig. 5. Example of a discarded form after three attempts of reaching the *French flag*

The second one present the case where removal is due to the exceed number of cells in the generating form relative to that of the sought form with regard to the defined *multiplicity* order. Here too, output results show that the algorithm aborts the simulation of the genome [**3 2 12 3**] and proceeds to the next one in the base.





```
3 2 12 3
-----------------------------------------------addcells
----AGE: 1.16
--NUMBER OF CELLS : 80--Overtaking the right number 70
---NEXT GENOME : N° 8269
```

Fig. 6. Example of a discarded form with a number of cells exceeded that targeted

5 CONCLUSION

Inspired by the concept of *epigenetic landscape*, we have shown the importance of the developmental process constraints on the final form. Starting from 2 different genomes, we have achieved the same form thanks to the choice of feedback mechanisms to deal with stress throughout development. Theses mechanisms are, in the one hand, cellular differentiation which allows the activated gene to encode the colors of the form. In the other hand, they imply successive changes of direction that allow to generate (to design) the shape as it goes along.

5.1 RELEVANCE to BIOLOGICAL ISSUES

The model is tested in French flag pattern because it is a widely known issue in studying the factors that put into play in cell differentiation. It constitutes the first steps towards the comprehension of biological mechanisms. Nevertheless, in future works, it might be also possible to test the model in real case. For example, in genetics, *single nucleotide polymorphism* (SNP) is the variation of a single base pair of the genome between individuals of same species. They are used for recognition of individuals or reconstruction of family trees. In our algorithm, we could build the genome base so that they are all identical except the last gene in each of them. This gene, different from one genome to another, will constitute our SNP. And then we can explore this base to study the influence of this small genome variation by comparing their generated forms.
We can also consider applying the algorithm to look for genomes that can generate more complex forms. Indeed, from the cell lineage of an organism, we can know its number of cells and its different cell types. We can take the case of Caenorhabditis Elegans (*C. Elegans*). This is an eukaryote[1], which means that it shares cell structures, molecular structures and control channels with organisms of higher species. Thus, its biological information (embryogenesis, morphogenesis, growth, etc.) could be directly applicable to more complex organisms, such as humans. In addition, it has a fixed number of cells. The adult hermaphrodite is composed of `959` somatic nuclei and the adult male of `1031` while the young consists of `1090`. The algorithm can be used to search in the base a genome which can generate a form with the same number of activated genes than cellular types in *C. Elegans* and having an identical number of cells.

5.2 FUTURE WORKS

To consider the description of all forms of dynamics, we plan to study a generalization of our mathematical model. In the current model, when a cell initiates mitosis, we just create next to it another one. The mother cell does not implement any other mechanism. Therefore, we will consider a mathematical model where the mother cell can in each time step:

- remain quiescent as it is in the current model,
- migrate to another point or,
- die.

This formalization will allow to reach all possible forms. The development of a such model raises new challenges in mathematical and numerical perspectives. Because describing all possible dynamics for each cell at each time step offers many options to consider. In computing, *parallel programming* on GPU[2] could overcome scalability and duration of simulations.

[1] A eukaryote is an organism whose cells contain complex structures enclosed within membranes.

[2] Graphics Processing Unit

References

1. J.-P. Aubin. *Viability theory.* Birkhauser, 1991.
2. P. Chevaillier, S. Bonneaud, G. Desmeulles, and P. Redou. Experimental study of agent population models with a specific attention to the discretization biases. In *In Proceedings of the European Simulation and Modelling Conference (ESM'09), Leicester, UK.*, pages 323–331, 2009.
3. G. Desmeulles, G. Querrec, P. Redou, L. Misery, V. Rodin, and J. Tisseau. The virtual reality applied to the biology understanding: the in virtuo experimentation. *Expert Systems with Applications*, 30(1):82–92, 2006.
4. R. Doursat. *Organically Grown Architectures: Creating Decentralized, Autonomous Systems by Embryomorphic Engineering.* Springer-Verlag, 2008. Organic computing.
5. A. Fronville, F. Harrouet, A. Desilles, and P. D. Loor. Simulation tool for morphological analysis. In *In Proceedings of the European Simulation and Modelling Conference (ESM'2010), Hasselt, Belgium.*, pages 127–132, 2010.
6. A. Fronville, A. Sarr, P. Ballet, and V.Rodin. Mutational analysis-inspired algorithms for cells self-organization towards a dynamic under viability constraints. In *SASO 2012, 6th IEEE International Conference on Self-Adaptive and Self-Organizing Systems, Lyon, France.*, pages 181–186, 2012.
7. A. D. Goldberg, C. D. Allis, and E. Bernstein. Epigenetics: A landscape takes shape. *Cell*, 128:635–638, 2007.
8. J. Henderson and D. Carter. Mechanical induction in limb morphogenesis: the role of growth-generated strains and pressures. *Bone*, 31(6):645–653, 2002.
9. S. Kauffman. *The Origins of Order : self-Organization and Selection in Evolution.* Oxford University Press, USA, 1993.
10. B. Lawson and S.Park. Asynchronous time evolution in an artificial society mode. *Journal of Artificial Society and Social Simulation*, 3(1), 2000.
11. C. Melani, N. Peyriéras, K. Mikula, C. Zanella, M. Campana, B. Rizzi, F. Veronesi, A. Sarti, B. Lombardot, and P. Bourgine. Cells tracking in the live zebrafish embryo. In *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, volume 1, pages 1631–1634, 2007.
12. G. Müller and S. Newman. *Origination of organismal form : beyond the gene in developmental and evolutionary biology.* MIT Press, 2003.
13. N. Peyriéras. *Morphogénèse : L'origine des formes*, chapter Morphogenèse animale, pages 179–201. Belin, 2006.
14. P. Reignier, F. Harrouet, S. Morvan, J. Tisseau, and T. Duval. AReVi: A virtual reality multi-agent platform. *Lecture Notes in Computer Science*, 1434:229–240, 1998.
15. F. Varela. *Principles of biological autonomy.* 1979.
16. C. Waddington. *Organisers and Genes.* 1940.
17. L. Wolpert. Positional information and the spatial pattern of cellular differentiation. *Journal of Theoretical Biology*, 25:1–47, 1969.