Towards the Evolution of Multicellularity: A Computational Artificial Life Approach

by Moritz Buck

Submitted to the University of Hertfordshire
in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

under schedule A

July 2010

“There is grandeur in this view of life, with its several powers, having been originally breathed into a few forms or into one; and that, whilst this planet has gone cycling on according to the fixed law of gravity, from so simple a beginning endless forms most beautiful and most wonderful have been, and are being, evolved.”

Charles Darwin — *On the Origin of Species*

Acknowledgements

I first want to thank my family for always being there.
I want to thank all my friends.
I want to thank Chrystopher Nehaniv, my supervisor.
I want to thank everyone from the Science and Technology Research Institute of the University of Hertfordshire.
All my house-mates.
Barkeepers.
Coffee-makers.

Moritz Buck

University of Hertfordshire
July 2010

Abstract

Technology, nowadays, has given us huge computational potential, but computer sciences have major problems tapping into this pool of resources. One of the main issues is how to program and design distributed systems.

Biology has solved this issue about half a billion years ago, during the Cambrian explosion: the evolution of multicellularity. The evolution of multicellularity allowed cells to differentiate and so divide different tasks to different groups of cells; this combined with evolution gives us a very good example of how massively parallel distributed computational system can function and be "programmed".

However, the evolution of multicellularity is not very well understood, and most traditional methodologies used in evolutionary theory are not apt to address and model the whole transition to multicellularity.

In this thesis I develop and argue for new computational artificial life methodologies for the study of the evolution of multicellularity that are able to address the whole transition, give new insights, and complement existing methods. I argue that these methodologies should have three main characteristics: accessible across scientific disciplines, have potentiality for complex behaviour, and be easy to analyse.

To design models, which possess those characteristics, I developed a model of genetic regulatory networks (GRNs) that control artificial cells, which I have used in multiple evolutionary experiments. The first experiment was designed to present some of the engineering problems of evolving multicelled systems (applied to graph-colouring), and to perfect my artificial cell model. The two subsequent experiments demonstrate the characteristics listed above: one model based on a genetic algorithm with an explicit two-level fitness function to evolve multicelled cooperative patterning, and one with freely evolving artificial cells that have evolved some multicelled cooperation as evidenced by novel measures, and has the potential to evolve multicellularity. These experiments show how artificial life models of evolution can discover and investigate new hypotheses and behaviours that traditional methods cannot.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

> "I have no doubt that in reality the future will be vastly more surprising than anything I can imagine. Now my own suspicion is that the Universe is not only queerer than we suppose, but queerer than we *can* suppose."
>
> J. B. S. Haldane

If you look around you in the world, you can find complexity and amazement, from the eyes of a fly to the blue jewel that is our planet and further! And when it comes to our planet, much credit is due to evolution. From what once was a fiery Hades some 4 billion years ago has become an oasis of life. In these aeons past, during the pre-Cambrian our unseen microscopic relatives working hard to transform our planet, paving the way, without knowing it, for us, had already evolved most of the tricks of the trades that we think are the gifts of our higher taxa of the tree of life.

Bacteria and Archaea live in complex societies, different strains cohabiting in biofilms, each having their role in their micro-ecologies, their own miniature cities. These societies still exist nowadays, in every puddle, in every spadeful of mud, in your own gut. Microorganisms are not little cell-sized loners going for the kill; many of them are much closer, in a sense, to us. Of course they have no brain, no free will, they are just little cells containing an extremely complicated set of chemical computations. But even being some of the less complex creations of evolution, they can cooperate.

Cooperation is one of the cornerstones of evolution. At the origin of life, those 4 billion years ago, some molecules of some sort worked together to create a work of ever evolving chemistry and interaction. I might give too much agency to all those parts (to be honest any agency is too much), but one cannot resist the lure of romanticizing Nature some times. Cooperation in one form or the other is omni-present in this world, ecosystems survive only through specialized species performing their role in the system, as does any multicellular organism, where every cell performs its task. In a genome as "selfish" as the genes might be, they still form a part of a cooperative unit, if one goes rogue the survival of the whole is at stake. Kropotkin in his book, *Mutual Aid* Kropotkin (1904), was one of the first to state the importance of it, as idealistic as he was (and critical of the idea of "survival of the fittest"), he looked around and he saw a well oiled system where in every interaction, between species and inside species, survival was never the work of an organism alone, everything influenced everything. Mutual aid, not necessarily conscious, drove evolution, not mutual destruction.

Multicellularity, symbiosis, ecologies, societies, genomes, swarms, herds, all are in a sense cooperation, mutual aid. The main question, as a scientist, for me, is: "Is there an underlying, general, principle of cooperation?". This is of course a big question, and not one I can answer.

Cooperation under all its forms isn't without its problems. Any kind of cooperative system, has what is often called in the biology literature, "cheater". A "cheater" is an element in a cooperative system that will try to get the benefits of the system, without having to pay the cost. In biology, most cooperative systems have evolved safeguard mechanisms, self-policing technology, to limit the damage such cheaters can do (Michod, 2003). At every possible level one can find some: From the "police cells" of our immune system, to pleiotropy (one gene with multiple functions) tricks in the genetic regulatory network of cells Foster et al. (2004). But as much as cooperation is visible and quite well understood the flip-side of it, the self-policing, tends to be difficult to detect and is little studied as such.

Again as for cooperation there is more to the understanding of self-policing, than just the biology of it. In modern sciences and technologies we are moving more and more towards our first real von Neumann replicators being robots on Mars or the moon, nano-bots in our bloodstream, or even very advanced self-replicating software agents. Most of those machines, will be programmed to do some specific task, but being self-replicating those machines will have the possibility to evolve and, depending on the system we are speaking about, this evolution could be extremely fast. As soon as a system is enmeshed in the process of evolution it will run the risk of being hijacked by it, turned away from its purposes by the pressures of evolution. Understanding self-policing mechanisms, understanding the safeguards life on earth evolved to protect itself from rogue elements is vital.

The research in this thesis, will of course, not aim to solve all these problems, but I hope that it will, in a first place raise some awareness on some of those issues, and secondly try to present some work I think goes towards answering them.

A major trend in computer sciences in the last couple of years and probably in the years to come, is toward massively parallel and/or distributed systems. Of which probably the most well-known and established one is the internet.

What is meant by massively parallel and distributed system is a systems where a huge number of more or less simple processing units are connected to each other (in a structured or non-structured manner) to perform some computations. The computations those processors are performing do not have to be directly connected (they do not have to perform a common task), but computations performed on one processing unit can influence the rest of the network.

These kinds of system are becoming more and more common: the internet, Beowulf clusters, robot swarms, and so forth. These systems are very interesting because they have a very high potential processing power, yet taping efficiently into that power is very difficult because it is very difficult to program such systems.

So one of the starting points of this work was to develop new methods to design (possibly through evolution) what I will call a *multicelled computational system*: massively parallel distributed computing systems based on organisational principles like those of multicellular life. But what became apparent early on during this time was to evolve this kind of system, it had to be considered differently to a "single-celled" computational system. The first concept would have been to go towards the concept of developmental systems. Developmental systems are systems based on the early cellular development in multicellular (mostly eucaryotic) organisms, the idea being to start of with one cell, which after a number of divisions, gets a complex structure and organisation. The problem being that the evolution of development[1] in biology (and in theory) is still quite badly understood. Which brought me to push this idea one step further: the modelling of the evolution of multicellularity and one of the main hypothesis of this work: "to evolve good multicellular systems, one needs first to evolve multicellularity"[2]. Hence the focus from this work slowly shifted from the idea of developmental multicelled computer systems to understanding the evolution of multicellularity.

The purpose was and still is, in part, to develop computer systems, but I firmly believe that to use evolution efficiently for this purpose, much more about the major transitions in evolution, cooperation, conflict mediation, and multicellularity needs to be understood.

1.1 Thesis Statement

The work I will present in this thesis will first present an artificial life cell for the study of evolution of multicelled systems. The goal being to design models that could evolve multicellularity from single cells, without explicit fitness. These kind of models would be complementary to more traditional models of evolutionary theory and could help the computer science community to design and build biology-inspired massively parallel computational systems.

1.2 Contribution

The main contributions to knowledge presented in this thesis are:

1. Presentation of an artificial cell model usable for a wide variety of models and problems (Chapter 4)

2. Development of a graph colouring method using this artificial cell as the base of a multicelled environment (Chapter 5)

3. A fitness-based model for the evolution of multicelled cooperation in a "colony"-type environment, which shows the link between fixation of a complex adaptation (such as multicelled cooperation) depending on the environment (Chapter 6).

[1] Studied in the new-ish field of evo-devo (Evolution of Development)

[2] And then one can go to development, and to a good (whatever good is) system.

4. Showed how the models I present can have behaviours that are not trivial with mathematical modelling methods (Chapter 6).

5. Presentation of a cell-based model allowing free evolution of multicelled cooperation (and potentially multicellularity) without explicit fitness.

1.3 Outline

For the rest of this thesis I will first introduce in Chapter 2 the state of research in evolutionary theory, open questions, and how I hope my work can contribute to this field. In the second part of my literature review (Chapter 3), I will present the computer science part of the background, mostly centred on artificial life and more specifically multicelled artificial life (Section 3.4), and evolutionary algorithms which I used for my experiments 4.2.

The rest of this thesis will be about the actual research I have done. In Chapter 4, I presented my novel artificial life cell model, and some other algorithms I have used throughout this work. In Chapter 5, I will present an experiment in which I evolved a simple multicelled system to colour graphs. This experiment shows how one can evolve a multicelled computing system for classical optimization problems, and will explain and illustrate its problems and shortfalls. In Chapter 6, I present a first simple model where an explicit fitness model is present to model evolution of multicelled cooperation. This experiment helps me to understand and clarify some concepts about complex fitness landscape for multi-level systems. It shows as well how such a model can show behaviours that would not have been possible to have in more classical models of evolution. And in the last experiment (Chapter 7), I present an implicit fitness model of bacterial-like evolution that evolved some multicelled cooperation and has the potential for multicellularity.

Chapter 2

Evolutionary Theory

> "Any competent biologist is aware of a multitude of problems yet unresolved and of questions yet unanswered."
>
> Dobzhansky (1973)

2.1 Evolution Theory: Introduction

The field of evolutionary theory has gone a long way ever since Darwin's voyage on the Beagle in the 1830s and the publication of *The Origin of Species* in 1859. The insights of August Weismann leading to the demise of Lamarckism, the addition of population dynamics and Mendelian genetics to create the modern evolutionary synthesis in the first half of the 20^{th} century, the extension of it with modern molecular biology after the discovery of structure of DNA, all the way to the resurgence of Lamarckism in the form of epigenetic inheritance have shaped and improved our understanding of the process which have made earth what it is nowadays. But still a large part of the process is badly and even misunderstood, and not only by the layman.

Evolution is to biology what history is to political sciences: one cannot understand the latter without understanding what brought it into being. But where for history its effect on politics is probably weakening with time and so mostly the latest part (of which we have the completest records) is important. With evolution and biology it is mostly the reverse: all life comes from a common origin and that origin is very far away, and those origins and the evolution during the earlier periods of life on earth are still extremely important yet very difficult to study.

Actually every part of evolution is difficult to study. The only accurate and precise record we have is the snapshot of the life on earth right now. Even though we carry a lot of evolutionary baggage in our genome (see section 2.3.1), it is a bit like studying history by only looking at the state of current political affairs; you could study history, but it is greatly difficult, with a high risk of inaccuracies, and chances of missing big events completely. The other solutions are to study fossil records, but again this is not without problems. Most importantly, the whole fossil record is greatly biased: mostly hard-shelled and bony animals, and plants are represented; the bacterial and protist worlds are almost not represented, and when they are, they are very difficult to study for their lack of evident physiological characteristics. Hence the species found in the fossil record represent only a tiny part of the history of life, and only this part can be reasonably interpreted. Also in the cases where there are fossil records, they are often incomplete, making an exact account near impossible.

But one can still study evolution and get a better understanding of its underlying processes, in my opinion mostly through modelling methods.

This is due to one of the main features of the evolutionary process: it is not only a practical process, evolution can be theoretically and mathematically characterized. Any system showing a certain set of characteristics can evolve. This principle has been extensively accepted since the founding of the modern evolutionary synthesis and the use of mathematical modelling has since been one of the main tools to study different processes and problems. Although it has to be acknowledged that theoretical approaches to the study of evolution are complementary to the experimental approaches, the theoretical approaches are still very important because so much experimental evidence is open to interpretation.

Evolution is first and foremost a purely theoretically identified process, not necessarily a biological one. Evolution will happen in any system exhibiting certain properties. The exact formulation of these properties vary slightly depending on whom you read but basically all are variants and precisions on the same theme, the first time it was explicitly identified was in (Lewontin, 1970) as:

- phenotypic variation
- differential fitness

- heritability

Any system exhibiting these properties will have the mean of its individual fitnesses increase over time (in most circumstances).

Let me try to clarify this kind of system more in depth. An evolutionary system is composed of a population ϵ of entities . Hereditary reproduction means that each entity of ϵ can make a "copy" of itself into ϵ, and with variation is meant that that copy can be different to some extent from its original. Yet there are differences in fitness between every entity in ϵ (differences in reproduction rate or survival, or others), but as the offspring ("copy") of any entity is correlated to its parent, their fitness will be too. So to simplify again: entities with a higher fitness might have more offspring of better quality, hence they will have more chances to survive and reproduce themselves, and so forth.

Evolution can be confusing in a certain sense by these simple required properties: even though it is a theoretical framework applicable to any kind of system possessing certain properties, the implementation details of evolution in nature are discussed by human observers describing what they think they see (what is the fitness, who is the individual, what is the population, is a group structure relevant...). The details will often depend on the type of problems and systems that are addressed or observed, and the way they will be studied by said human scientist.

I will, in the next section, first describe the gene-centred view of evolution, the most used framework in which evolution is studied, then describe the main methodologies with which, and in a last section describe more specifically issues and frameworks linked to my specific field of interest: evolution of cooperation and multicellularity.

2.2 Dawkins and the Gene-centred view of Evolution

The gene-centred view of evolution is based on the works of Hamilton (1964a,b), Williams (1996), and later popularised by Dawkins (1976). This "view" of evolution is more or less directly the result of the modern-evolutionary synthesis, meeting the newly discovered central dogma of molecular biology (Crick, 1958). The premise of the gene-centred view is that for selection to be able to act upon some entity, this entity needs to be persistent through generations in the same form. If not it would not be possible for heritable changes on that entity to accumulate. With the newly discovered structure of DNA in 1953, the candidate to be such an entity was closer than ever. The cell or the organism is not constant enough, it does not persist in time, but the nucleotide sequences of the DNA included in every cell (or organism) are kept from generation to generation and can have a cumulative effect on the survival of the cell (or organism).

Richard Dawkins in his book *The Selfish Gene*(Dawkins, 1976) defines his genes on which selection is happening in an almost tautological way. His statement starts with the fact that in sexual organisms crossovers break and recombine the chromosomes of the mother and the father, hence the chromosomes are not persistent enough to be the units of selection (nor is the whole genome either, for that matter, as chromosomes are picked randomly from each of the parents). But you can imagine any stretch of nucleotides small enough not to be broken by cross-over all that often (so that it can persist in a lineage), such a stretch of DNA he calls "gene". This stretch of DNA can be arbitrarily short but if it is too small becomes uninformative so for the purposes of modelling, they need to be of a "reasonable length".

This "gene" should not be confused with the biological gene. In biology, a gene is roughly defined as a stretch of DNA that codes for a protein sequence (and a certain stretch around with promoter, and regulatory sites). Dawkins' "gene" can be any stretch of genome, with any number of biological genes or none at all. The only thing counting is for it to have the capacity to be persistent in the lineage. To avoid any confusion I will use the term genetic replicators, introduced by Dawkins in *The Extended Phenotype*, when I speak about Dawkinsian "genes".

To clarify a bit, a (sexual diploidic) organism's genome is composed of two sets of equivalent genetic replicators; by equivalent I mean each genetic replicator in one set has an alternative form in the other set (with some exceptions of course, like the X and Y chromosomes, and mitochondrial DNA) . Each set got inherited from one of the organism's parents, the organism's brother and sister share half of your genetic replicators, and so on. So genetic replicators are kept alive throughout generations (by definition, remember a genetic replicator is a stretch of DNA *not* destroyed by cross-over). So if a certain genetic replicator at a specific place in a genome is "better" (by, for example, replicating more) then another genetic replicator at the same spot, this genetic replicator's proportion in a population[1] of organism will increase. So any adaptation can be seen as ultimately benefiting the replicator.

2.3 Methodology and Modelling in Evolutionary Sciences

The study of evolution is to a certain degree a historical science, we want to understand the history of life itself. But evolution is also a process: a process that acts on any system implementing Lewontin's three rules. Hence evolutionary models can often be viewed through two lenses: one focussing on understanding the history of life and explaining biology through evolution; and the other on understanding the evolutionary process and discovering the main principles underlying it. Some models and methods might be only useful for one approach, and some might seem to be relevant for both but should not, hence caution and critique is to be had when looking at specific models as to which part of the study of evolution it is relevant to.

2.3.1 Phylogeny and Comparative Molecular Biology

The discovery of the structure of DNA and the advances in sequencing technology has allowed huge leaps of knowledge about evolution, of which phylogeny has probably contributed to most. Phylogeny is the part of biology that concerns itself with building "trees of life". It was probably the first way of studying evolution and modelling its instance with life on earth. The first real phylogenies were based on purely physiological characteristics[2], and this is still done nowadays with increasing precision. But the arrival of genomic and proteomic data has increased the possibilities in this fields by orders of magnitude.

One of the difficulties of physiological phylogeny is that one needs comparable characteristics for the organisms we want to build a tree for: it is very difficult to compare a bush with a bat. Genomic data gave us the necessary similarity measures. For example, all genomes that we know of on earth share

[1] The notion of population can be problematic, and is often anthropocentrically defined.

[2] One could for example build a phylogeny of the evolution of birds by comparing forms of beaks, and to each pair of beaks give a similarity score, and out of the matrix of "beak similarity" draw a tree.

genes for different tRNAs[3], these are strings of bases which we can then compare automatically (which is also an advantage compared to the "old school methods") across all species (for which we have sequenced the tRNAs), and so build a phylogeny of "all" life on earth.

These methodologies more than anything (should) have dissipated last doubts about the theory of evolution. Also these methods of comparing genomic and proteomic data across species and organisms, called comparative biology, are the main methodology to study evolution the experimental way.

These methods have allowed us to have a much better picture of evolution, and how it happened on earth. It allowed us to separate Bacteria and Archaea into two different domains, confirmed the endosymbiotic theory[4],or showed us just how close we are to our closest animal relatives, the apes. These methods can have a very surprising precision: lately one was able to track early human migrations (around 100 thousand years ago), by building phylogenies with data from secluded hunter-gatherer societies.

There are some limitations with these experimental methodologies. It is in a sense a purely historical science, it only can tell the story from a result point of view. It is very hard (if not impossible) to get any kind of mechanistic or theoretical result from these. One can build a "who is the cousin of whom", but we cannot really say why and how. This is the realm of different approaches to the study of evolution.

2.3.2 Mathematical Modelling

In the first half of the 20th century, a new way of understanding and studying evolution emerged. This new way of looking at evolution, led by Fisher, Wright, and Haldane (and many others), would become known by the name of modern evolutionary synthesis. It combines evolutionary theory, with Medelian genetics, and population dynamics. The modern evolutionary synthesis paved the way for most of what has been discovered about evolution since then, and

[3] tRNAs are part of the essential transcription machinery, they do the "translation" from RNA to protein.

[4] The endosymbiotic theory (Margulis, 1981) says, that the mitochondria in our cells were the descendants of actual bacteria that lived in symbiosis with other bacteria (around 1 billion years ago, during the late pre-Cambrian), that actually got internalised and now cannot live alone any more (neither the host, by the way). This was proven when doing the phylogenies of the little genetic material left in the mitochondria.

led to new methodologies.

One of these methodologies is to study evolution with mathematics. The modern synthesis itself is partly born out of mathematics, when in the early 20th R. A. Fisher proved mathematically that the sum of the effects of Mendelian genes, if there are many of them with small effects, can give a continuous spectrum of variety. This proved that Mendel's genetics are consistent with the theory of evolution.

Ever since then many diverse mathematical models have been in the centre of the study of evolution. There exists a plethora of models, most of them very problem specific. I will describe shortly how these are usually structured.

2.3.2.1 Classical Population Dynamic-like Models

Most of the mathematical models of evolution are directly derived from the gene-centred view of evolution, even its "opposition" start usually from there, but add other effects on top of it. So most models are sets of equations describing the spread of different genetic replicators situated on the same locus of the genome of individuals in a population, there are multiple alleles for this replicator in the population so their is one equation describing the spread of this allele depending of its own frequency, the frequency of the other alleles and a set of parameters describing the interaction of the different alleles at that locus. These models mostly originate from population dynamics, and use similar methodology. So in a case of a generation based model with t different phenotypes for generation $n+1$ one would have a proportion of phenotype i described by some predefined function f_i:

$$p_i^{n+1} = f_i(p_0^n, p_1^n, ..., p_t^n)$$

. For the case of a differential equation based model one would get the rate of change of the proportion of allele i in the population defined as:

$$\dot{p}_i = f_i(p_0, p_1, ..., p_t)$$

. These functions can be defined as any possible function, but traditionally they are polynomial function, because of the great difficulty of analysis of any other differential equation systems. Often also the degree of the polynomials (hence interactions) does not exceed two. So traditionally for a system with two phenotypes for example, one would have a system of equations of this type:

$$\begin{cases} \dot{p}_0 = a_{00} \cdot p_0^2 + a_{11} \cdot p_1^2 + a_{01} \cdot p_0 p_1 + a_0 \cdot p_0 + a_1 \cdot p_1 + a \\ \dot{p}_1 = b_{00} \cdot p_0^2 + b_{11} \cdot p_1^2 + b_{01} \cdot p_0 p_1 + b_0 \cdot p_0 + b_1 \cdot p_1 + b \end{cases}$$

, where the a and b parameters are, in a broad sense similar to the parameters in the pay-off matrices for the game theoretical models (Section 2.3.2.3). They represent the effect of the interactions (or lack thereof) they are linked too.

These equation systems can then be studied with classical dynamical systems methods to find stable points and analyse their stability. They can be use for a wide variety of studies, which range from host-parasite evolution (Mode, 1958) to plant-resistances.

2.3.2.2 Price's Equation

Price's Equation is another approach often used for mathematical models of evolution. It has been developed by Price (1972), it is an algebraic result that describes the evolution of a population from on generation to the next. In a population of entities each parent entity i possesses a measurable phenotypic character z_i (the character in which we are interested), the average of all z's is $\bar{z}$, also the fitness of entity i is w_i and the average of all of those is $\bar{\mathrm{w}}$. From those quantities one can derive that

$$\bar{\mathrm{w}}\Delta\bar{z} = \mathrm{Cov}(\mathrm{w}_i, z_i) + \mathrm{E}(\mathrm{w}_i \Delta z_i)$$

, where $\Delta\bar{z}$ is the change of average character from one generation, $\mathrm{Cov}(\mathrm{w}_i, z_i)$ the covariance of fitness and character, $\mathrm{E}(\mathrm{w}_i \Delta z_i)$ the expected value, and Δz_i is the difference of character between the entity i and its offspring(s). If we introduce relative fitness as $\omega_i = \frac{\mathrm{w}_i}{\bar{\mathrm{w}}}$, and if we divide both sides of this equation with $\bar{\mathrm{w}}$ we get:

$$\Delta\bar{z} = \mathrm{Cov}(\omega, z) + \mathrm{E}_{\mathrm{w}}(\Delta z)$$

, where $\mathrm{Cov}(\omega, z)$ is the covariance between z_i and ω_i, and $\mathrm{E}_{\mathrm{w}}(\Delta z)$ is the fitness-weighted Expected value of Δz_i.

I will not go in depth into the interpretation of this equation but, just show how it differs from the type of modelling described in the previous section.

The second form of the equation is reasonably easy to visualize. The LHS represents the change of a characteristic over time, and this change can be decomposed into two quantities, $\mathrm{Cov}(\omega, z)$ which represents the correlation between the fitness and the characteristic (basically the idea of natural selection: if tall animals are fitter, size will increase), and $\mathrm{E}_{\mathrm{w}}(\Delta z)$ represents the transmission-bias of the characteristic, the copying fidelity so to say.

So basically this equation gives a decomposition (purely mathematical) of the change of a characteristic over time, into two quantities that seam biologically relevant. This decomposition has been used mostly in multi-level selection theory (Okasha, 2006), mostly because the covariance can be decomposed over and over again, and each decomposition can represent a different level of selection.

It is to note that this decomposition is purely algebraic, so even though the quantities of the decomposition show resemblance to biological quantities, the decomposition of $\Delta\bar{z}$ is not unique, and there are other ways to do this kind of decomposition (for example, contextual analysis (Goodnight, 2005)).

2.3.2.3 Game theory and Evolution

During the second half of the 20th century, R.C. Lewontin (Lewontin, 1961) developed a new type of models to study evolution, these new models came from the world of economics, and are known as game theoretical models (Axelrod, 1985; Smith, 1982). These kinds of models consider every entity of an interaction as a player in a game, in which usually the player needs to win. The classical game theory models consist of iterative games where two or more players interact one after the other. So if they are only two players, they play a first round of the game, check who wins, then another round, and so forth (iterative games).

In this kind of setup usually the games are modelled by a “pay-off” matrix. This matrix will have as many dimensions as players, and lines for every possible strategy for every player. Each field of the matrix will contain the “pay-off” for each of the players according to their chosen strategy. The “pay-off”, in the case of economy, would be monetary, and for evolutionary models it would often be fitness, or resources.

In Table 2.1, I present a pay-off matrix for the hawk and dove game which is often used as example in the biological literature, and which was used by Smith and Price (1973) in their seminal paper. In this game two animals fight over a resource, each animal has two (possibly genetically determined) possible behaviours: Hawk and Dove. In the case of the hawk behaviour, the animal will fight until defeat (or the enemy retreats), and for the dove

	Hawk	Dove
Hawk	−10/−10	20/0
Dove	0/20	10/10

Table 2.1: A Pay-off matrix for the hawk and dove game

behaviour, the animal will always retreat if he meets a hawk, and if two doves meet one random one will. Fighting has a cost (the animal gets hurt). So we can see, in the pay-off matrix, that if a hawk meets a dove, the hawk gets all the resources (20 units), if two doves meet they share[5] the resource. In the case of two hawks meeting, both hawks get hurt (cost of 20 resource units) but one wins the resources randomly (so ends up even), so on average they both lose 10 units of resources $((0 - 20)/2)$.

This example has been used by John Maynard Smith and Price to present the concept of Evolutionarily Stable Strategy (ESS), which is one of the main reasons game theory has become so popular in evolutionary theory. The concept of ESS is related to the concept of Nash equilibria (Nash, 1950). An ESS is a strategy that cannot be invaded by any other strategy adopted by a small proportion of the population.

In the example of the hawk and dove game the dove strategy, neither "all dove", nor "all hawk"[6] are ESSs, because each of them can be invaded by the other behaviour. There is however an ESS, which is mixed (there is a ratio of the population of doves and hawks that is stable, e.g. can not be invaded).

The concept of ESS is probably responsible for the big success of evolutionary game theory, it gives a simple framework to model evolution of behaviours, and a criteria for stability.

[5] Actually, the game supposes that the resource is indivisible so, the winner is taken randomly, meaning that half of the time one dove will get the resource, the other half of the time the other one.

[6] All dove and all hawk meaning that the whole population behaves according to one or the other strategy

2.4 Theories and Concepts for the Evolution of Cooperation and Multicellularity

As Kropotkin and other 19^{th} century naturalists recognized already, cooperation (Kropotkin used the term *mutual aid*) of some degree or another is extremely omnipresent throughout the animal kingdom, and if one think of cooperation in a wider sense (including symbiosis for example) you can find cooperation in all kingdoms of life: association of plants and fungi (for nitrogen fixation), social bacteria "hunting in packs", diverse species of birds flocking together to migrate... The number of examples of cooperation in Nature is uncountable. Hence the need to understand cooperation at an evolutionary level. A number of theories have been proposed to explain diverse degree of cooperation and Multicellularity.

2.4.1 Notes on Vocabulary

Before discussing further on I will first clarify three terms I will use consistently throughout this thesis: cooperation, multicelled interaction, and multicellularity.

Cooperation: I use in this thesis the term cooperation with a very wide definition that will incorporate the two other concepts I define here, as well as others. I will call cooperation any kind of interaction between two or more entities that is benefic to at least one of them (at the replicator OR vehicle level). This is purposefully a very inclusive definition including almost any kind of interaction between individuals of the same species or different species: altruism, parasitism, symbiosis, pack behaviour, commensalism, societies, even feeding. The notion of "benefic" in this definition is also kept broad so as to allow for any definition of fitness at any level of selection and organisation, or indirect benefit. For example, an interaction without which one of the interacting entities is penalized is also considered "cooperation" with this definition.

Multicelled cooperation: I will call multicelled interaction any kind of cooperation between very related cells. This could be multicellularity (as defined here) or pack behaviour, bacterial biofilms (with only one type of bacteria), sponges, algae ... I will use this term as a somewhat lesser form of early multicellularity, where differentiation has not evolved yet, or is not obvious. I will also use similarly multicelled system, any system that shows multicelled cooperation.

Multicellularity: I define multicellularity as a type of multicelled cooperation. The main characteristic of it being that certain cells of the set are differentiated, e.g. performing other tasks, different physiologically ... Multicellularity as defined here includes, for example, mammals, trees, slime moulds during their budding phase, social insects, and many more.

I will use those terms throughout this thesis as defined here. The way cooperation is defined here means that cooperation does not necessarily actively evolves, a cow and beetles cooperate because the beetle uses nutrients of the cow dung, yet this cooperation has not evolved, it probably has driven somehow the evolution of the beetle but the interaction itself has not evolved. However certain types of interaction such as multicellular interaction and multicellularity have evolved actively.

2.4.2 Kinship Selection

Kinship selection is a concept first formalized by Hamilton in (Hamilton, 1964a,b), but was already hinted to by Fisher and Haldane. Kinship selection is very widely used to explain and model evolution of altruism[7] and eusocial behaviour in the animal kingdom, and is widely accepted as the main drive for the evolution of these.

This concept, deeply linked to the gene-centred view of evolution, states that a genetic replicator will be able to spread in a population if its fitness gain for the carrier and all its related vehicle (weighted by their *relatedness*, this is called *inclusive fitness*) is higher than the fitness cost of that trait to the carrier. Formally Hamilton expressed it in the well-known Hamilton rule:

$$rB > C \tag{2.1}$$

where r is the *relatedness* between the recipient of the genetic replicator and the carrier, B is the benefit to the recipient, and C the cost to the carrier.

[7] Altruism is a type of cooperation where one or more entities benefit *at the expense* of the others.

Relatedness often refers to the probability of the recipient and the carrier share this specific genetic replicator. So for animal, for example, two siblings (or parents and offspring) share half of their genetic replicators, hence have a relatedness $r = 0.5$, cousins will share an eighth of their genetic replicators, hence have a relatedness of $r = 0.125$[8].

For example, to take an example attributed to Haldane, who said he would happily give up his life to save three of his brothers. As we saw before, each brother shares half of his genetic replicators with him, so a genetic replicator which would "push you" to save three of your brothers, would on average save 1.5 copies of itself. So sacrificing one copy to save 1.5 copies is a 'fair deal'. So basically, Haldane would sacrifice himself for three of his brothers (but not two). In terms of Hamilton's equation, we would have the following genetic replicator for this behaviour, $r = 0.5$, $B = 3$ (three siblings live), $C = 1$ (Haldane sacrificing himself), hence $0.5 * 3 > 1$, which is true, so this genetic replicator could theoretically spread.

One of the main issues with kinship selection is that the carrier of the genetic replicator (let's say an "altruistic" genetic replicator) needs to "know" how related (what the relatedness r is) he is to the recipient. I will just mention here two mechanisms: "green beard genes" and spatial effects.

The green-beard effect, first mentioned by Hamilton (1964a,b) and named by Dawkins (1976), is linked to a genetic replicator who has an effect on three specific phenotypic aspects:

- a signalling phenotype
- an effect recognizing that specific signal
- a special treatment of that phenotype

The carriers of that genetic replicator will hence be able to recognize other carriers of the same replicator (which will mean, that they are probably highly related, at least for that replicator), and cooperate altruistically with it. This will ensure that the relatedness in equation (2.1) is always high and hence allow the genetic replicator of altruism (the "green-beard gene") to spread.

Another way to achieve high relatedness, is through simple spatial effects. This is mostly connected to the concept of *viscosity*: the distance and

[8] This is for the case of diploid sexual animals.

speed of dispersal of offspring from their parents. Being it because of behavioural reasons or simply because the physiology of the organism limits their movement, if offspring stay close to their parents the probability of any altruistic behaviour being beneficial to a closely related individual will be logically high (if the local ecosystem and environment is 'right'), since related individuals are nearby.

Even though kinship selection is widely recognized as a main driving force for the evolution of altruism and certain other cooperative behaviours, it has to be realized that it only explains these certain kind of specific cases. Kinship selection will be hard pressed to explain the multitude of non-kin cooperation like symbiosis or human society for example. Another issue is instability, in many situations kin-selected cooperation can easily be destroyed by invading *cheaters*, cheaters being genetic replicators trying to benefit from the cooperators without 'paying'. Hence, often the evolution of some kind of self-policing is necessary [9]. For most of the theoretical and field work this problem can be safely ignored, especially when the studied population that are already stable and highly evolved, but as mentioned in Section 2.4.4, if one works on the transitions from one vehicle to another, this issue becomes major.

For kinship to explain altruism, it is vital to know which vehicle will likely benefit from the cooperative behaviour; different genetic replicators might try to favour different vehicles, and hence conflict on who is the recipient vehicles (or levels). For stable organisms, usually all (at least most) of the genetic replicators will favour the same vehicle, but this is an already evolved characteristic, it probably was not the case at the beginning of its evolutionary trajectory; the majority of genetic replicators probably favoured the lower-level vehicle. To explain the evolution of this "union" of genetic replicators to favour the same vehicle is one of the main challenges of evolutionary theory which will be addressed more in depth in Section 2.4.3.

2.4.3 Major Transitions in Evolution

Whatever school of thought you adopt as an evolutionary theorist (and there is almost as many as people), it is almost universally recognized that different

[9] The green-beard effect can be seen as such a system, as the same genetic replicator is responsible for the cooperation and the recognition.

levels of organization exist in biology. Every animal is composed of cells, every eucaryotic cell is composed of organelles, every ecology is composed of different organisms, and so on. Not only are their different levels of organization but also different evolutionary events linked with certain levels of organization. During the history of life on earth some levels of organization evolved, they are themselves results of evolution.

One of the most famous descriptions of these processes is John Maynard Smith and Eörs Szathmáry's book *Major Transitions in Evolution*. They confine their major transitions to the ones where the way information is transmitted from generation to generation change. They list them as follows:

evolution of the protocell: from the replicating molecule to a population of molecules in compartment

evolution of the genome: from independent genetic replicators to joined replicators

evolution of the genetic code: from the RNA world to the DNA/protein world

evolution of the eucaryotic cell: from a procaryotic cell to an endosymbiotic compartmentalized cell

evolution of sex: from the asexual clonal reproduction to a sexual reproduction

evolution of differentiation: from the protists to the animals, plants and fungi

evolution of non-reproductive castes: from solitary individuals to colonies

evolution of language: from primates to human societies

I will here not detail most of those transitions here but will concentrate on the one I am particularly interested for this work. For more information I greatly advise to read John Maynard Smith and Eörs Szathmáry's book.

The main focus of this thesis will be evolution of multicellularity (and its lesser form: multicelled interactions). I will not stick to the evolution of the higher phyla from the protists, but I will consider the general idea of evolution of multicellularity more generally. There is strong evidence for some degree of multicellularity in numerous species of Eubacteria and Archaea (West et al., 2007), Protists, and obviously the higher Eucaryotic phyla. One could also consider evolution of insect societies as an analogue of this kind of transition. Basically this work is motivated by the study of evolution of division of labour. My models will be mostly motivated by microbiological systems, but hopefully some of the conclusions and methods could be readily applied to the other aforementioned processes and possibly wider.

John Maynard Smith and Eörs Szathmáry are (or at least were) very much in the genes-eyed view of evolution, so why be interested in these transitions, if every selection boils down to genetic replicators? First because even if selection acts on genetic replicators, this selection acts *through* their vehicles, and understanding how the vehicles evolved can help understand the "hidden" level. Second, is the problem of the individual (Buss, 1987). The notion of individual in most evolutionary problems is a premise (e.g. the cell is the individual, or the bear is the individual, or the group is the individual), what the individual is is not questioned. The evolutionary game (of the genetic replicators, or whatever) is played through those specified individuals. But (mostly during transition of evolution) those games are rarely played at only one level at the time.

One of the biggest challenges of modern medicine, cancer, is exactly that. Some cells in your body for one reason or another (usually oxygen mutating some gene), tries to make it all alone without the help of the rest of the body, and in most cases loses all.

Between different levels of organisation evolution will have different "goals", and often those goals will be conflictual, and understanding the evolution of those conflicts and their policing is important for biology, medicine, as well as computer sciences.

The understanding of the evolution of these different levels of organisation, as well as the dynamics of these, is one of the main goals of the models of multi-level selection theory.

2.4.4 Multi-Level Selection Theory

Multi-level selection theory is more a set of open questions in evolutionary theory than an actual theory. The intrinsically hierarchical nature of biology has long fascinated scientists, and the different models and frameworks of multi-level selection theory are interested in the evolution and the workings of evolution in such a hierarchical world.

One of the main interests of this is how selection acts at different levels of hierarchy, and how the selections processes at different levels interact. Often there is conflict between selection at different hierarchical levels (think cancer cells in a multicellular entity). These questions do not necessarily contradict the gene-centred view of evolution. Even if selection is ultimately to the benefit of the genes, different genes can be favoured at different hierarchical levels, so in this view it is often about the conflict between different genes that get their fitness from different levels of selection.

2.4.4.1 Conflict Mediation and Policing

One of the main issues in a hierarchical system is the issue of conflict between the different levels. What is "good" for one level does not have to be good for the other level. In Eörs Szathmáry and John Maynard Smith ' *Major Transitions of Evolution*, they are more interested in the definition and possible evolution of the different levels, where traditionally the issue of conflict is more of an issue of persistence: if evolution of a new level of organisation has happened, how can it survive? But nowadays it becomes more and more obvious that the issue of conflict is an integral part at every step of the evolution of a new level of organisation. As much as cooperation is the driving force of the transitions, conflict is the main aspect that holds it back. For this reason during the course of the transition, conflict (also often called defection, mostly in the context of game theoretical models) needs to be mediated and methods of policing need to evolve.

In *Darwinian Dynamics*, (Michod, 1999) presents multiple models (using the types of mathematics described in Sections 2.3.2.1 and 2.3.2.2) showing how conflict mediation is sufficient to increase heritability of fitness, which is necessary for the evolution of a new level of organisation. Different types of cooperation, as he states in his work, trades fitness at a lower level of organisation (cost) for an increase in fitness at a higher level (benefit), conflict mediation ensures, by increasing the heritability of fitness, that the lower level entities share the costs. Without conflict mediation or policing a group of cooperating entities could be invaded by any new defecting mutation in the population that would destroy the newly acquired cooperation (or any higher level of organization).

In biology the examples of policing are numerous, from the immune system protecting the animal body from internal and external threats, to the separation of soma and germ lines to reduce the effect of any defection in the soma to be transmitted, over pleiotropy that tries to render any mutation to a vital part of the genetics of cooperation deadly to the cell (Foster et al., 2004). These are just examples that show that conflict mediation and policing is found not only in the equations of Michod, but also in every biological system we know of, from the smallest of the units and their conflict mediation mechanism: genes and pleiotropy (Foster et al., 2007) to the largest organisms and the immune system, or if we stretch it societies and police forces.

2.5 Contribution of my Research

In this thesis I will present a novel model of artificial cell that has been used to study some of the questions presented in this chapter. This artificial cell has one linear genome that encodes a genetic regulatory network. This artificial cell, if put into an (computational) environment in which it can reproduce with mutation, and some differential reproductive success is ensured, will be able to evolve. Depending on the problem to be studied the evolutionary environment can be changed and adapted. This kind of model allows for a much wider variety of behaviours than more classical approaches. Different selective pressures can have very more complex impacts on the evolution of the system.

In chapter 7, for example, I present an experiment in which artificial cells evolve freely in an environment allowing communication, there is no explicit fitness, and given the cell and its genetic system, evolution is solely driven by the design of the environment. This kind of system might allow one to study the open-ended evolution of different phenomena (in this example evolution of multicelled interactions). The precise implementation of the cooperation is not predefined, if multicelled interaction evolves it has not been defined in the system to start with (the environment has been set-up in a way for it to be possible though). This could allow us to “see” adaptations evolve *in silico*, and once they evolved the artificial life system can be examined at every possible level of its hierarchy (expression patterns, communications, network evolution, phylogenies, population dynamics...). These multi-level analyses would allow one to observe much more in detail what has driven certain adaptations, and hopefully to find new hypothesis to unsolved questions.

Many scientists have studied these much more open-ended models of evolution in the field of Artificial Life. I will present some of these in the next chapter.

My studies present an abstract framework to study low-level evolution (as opposed to high-level behavioural evolution), yet this abstract framework will serve to be a good metaphor of biological life so to ease transfer of knowledge from the artificial-life field to biology (and the inverse), and will have a complex internal structure to allow the evolution of complex adaptations (like multicelled interaction and multicellularity).

Chapter 3

Artificial Life and Bio-inspired Computing

> “Life is a process which can be abstracted away from any particular medium.”
>
> John von Neumann

3.1 Biology and Computer Sciences

Ever since engineers existed they have been fascinated by the power of biology, and tried to imitate its prowess. During the 19th century clockwork builders built automata looking like animals, and tinkerers have tried to build flying machines inspired by birds and bats. This is true even more so since the discovery of evolution and the invention of computers. Computer scientists have designed a plethora of algorithms inspired by biological systems: Genetic Algorithms (GAs), Neural Networks (NNs), swarm systems, Artificial Genetic Regulatory Networks (AGRNs)... Biology fuelled by the ingenuity of evolution has produced remarkably complex, reliable, robust, and efficient systems, that any engineer would have only dreamed of designing.

Turing (1950) saw this already in the 50s, when he invented the so-called “Turing test”, when he had the forethought that computers one day would be indiscernible from humans. We are, of course, not there (except in very specific fields), yet. But we are getting closer and closer, with newer and better technology appearing every year.

The advances in computing power allow the design of huge networks of processors (being for computation or robot control, for example). These will need new ways of programming, our standard programming paradigms being mostly not suitable for this kind of systems.

Yet, we don't have yet a very good understanding of complex dynamical systems, neither on the design side, nor on the control.

Biology has been showing us the way into the right direction, and the start is in evolution. Evolution has created an amazing array of complex dynamical system, including the control mechanism, and engineers that have used and/or designed bio-inspired systems, would probably agree with me.

In this section I will present bio-inspired and artificial life systems that I have used and/or have informed my experiments. First I will present Evolutionary algorithms, a method to "harness" evolution to solve problems and create designs. In a second part I will present Artificial Genetic Regulatory Networks (AGRNs). These are the algorithmic backbone of my systems. I will also present other artificial life models that have looked at the evolution of multicelled systems and multicellularity.

3.2 Evolutionary Computation

Early on in the history of computer sciences, engineers saw the advantages of evolution for difficult optimization problem. During the sixties independently, Lawrence Fogel, with Evolutionary Programming, John Holland, with Genetic Algorithms, and Hans-Paul Schwefel, with Evolutionary Strategies, developed methods of using Darwinian evolution for engineering problems. These three methods are now all subsets of what is known as Evolutionary Computing. The general principle of these methods is to use the Darwinian principles to "evolve" solutions to problems.

The idea is that a population of solutions to a specified problem are encoded in computational genomes, the quality of each genome is then measured by a fitness (or objective) function specific to the problem to be solved. Then a selection routine is run that makes copies of some of the solutions in certain proportions according to their fitness value. And each of those is modified to some extent. For more details to my exact implementation see Section 4.2.

This kind of metaheuristics is now widely used in high-dimensional optimization problems. They are easy to adapt to a wide variety of problems, easy to implement, and the need for in-depth knowledge about the problem (such as needed for specialized optimization algorithms) is reduced. They main structure of any GA (Figure 3.1) is directly based on the three main properties necessary for the evolutionary process to happen (see Section 2.1):

- hereditary reproduction
- variation
- differences in fitness

Those properties are of course implemented differently than in nature, and in a much more controlled environment.

To start with the problem to be optimised by a GA needs to be encoded into a representation on which the algorithm evolution can than act. Traditionally this representation will be some artificial genome or a tree structure, but could be any structure encoding the solution to a problem that one can copy, mutate and execute (with an appropriate interpreter). Once the problem has been encoded one can run a GA using it. A population of representation will be chosen for the start, then each of the representations will be executed and the quality of the solutions will be measured (the so-called fitness function). The next step is to 'reproduce' the fittest representations. A number of reproduction routines are used, but the general idea is that the representations with the highest fitness will get copied most, also each copy can be modified (so-as to have variation). Once the reproduction routine is finished, one starts again by evaluating the quality of the new representation and so on.

3.3 Artificial Genetic Regulatory Networks

3.3.1 GRNs in Biology

In every cell of a living organism is the genetic material needed to build the whole, yet cells tend to use only a small subset of the information available to them in their genome. Cells need to have mechanisms to regulate the expression of their genetic material: a Genetic Regulatory Network.

The genome of every living organism is composed of elements called genes, each gene normally controls the production of one protein, and proteins

```
Genetic Algorithm
  for $i \leftarrow 0$ to $length[Pop]$
        do $Pop[i] \leftarrow$ random genome
  repeat
            for $i \leftarrow 0$ to $length[Pop]$
                  do $Fitness[i] \leftarrow$ Fitness($Pop[i]$)
            $Pop \leftarrow$ Reproduce($Pop, Fitness$)
            Mutate($Pop$)
     until termination condition
```

Figure 3.1: Pseudo-code for a Genetic Algorithm

are the functional molecules that compose most of the cell's chemical reaction network. Each gene contains a DNA code that can be transcribed into the protein sequence it produces (the coding sequence). The GRN of a cell regulate this transcription process.

The transcription process is started when the RNA-polymerase enzyme binds to a promoter sequence upstream of the coding sequence. However, the binding of the RNA-polymerase is usually not simple, some specific enzymes (transcription factors) need to bind to the polymerase and to other sites (cis-regulatory sites) upstream so that the polymerase can bind to the promoter sequence. Also these enzymes that bind to cis-regulatory sites have to be produced by other genes (or the environment) and these genes are regulated as well, often by proteins that they regulate, thus creating a network of transcription regulation (Figure 3.2).

These networks can become very complex, the enzymes binding to cis-regulatory sites can be complexes of multiple proteins that need to be all present for that site to have an effect on transcription, multiple cis-sites can interact or negate their effects, or a single simple enzyme can stop the transcription all together.

Cis-sites in the regulatory region of genes are composed by enzyme-specific binding sites, that bind only one specific enzyme or type of enzyme, and multiple binding sites interact. Each cis-site is usually categorized as either activatory if they have the tendency to help the binding of the polymerase (so increasing the production of the protein), or inhibitory if the have the tendency

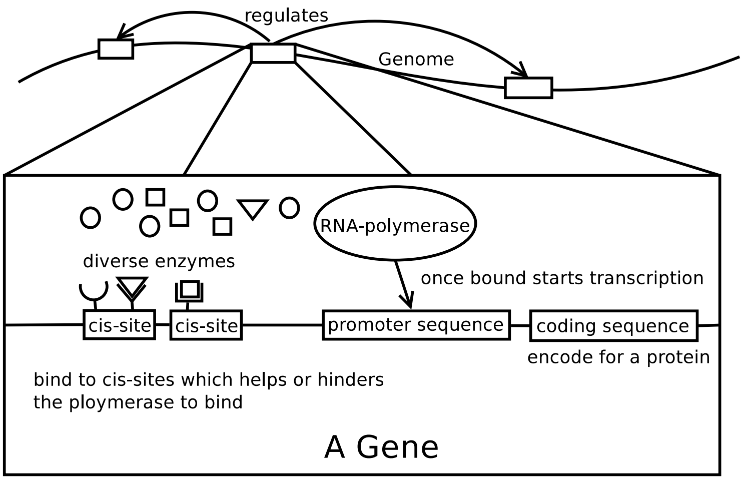


Figure 3.2: Simplified diagram of a gene

to reduce the binding characteristics of it (and so decreasing the production of the protein).

This great variety, and the omnipresence of GRNs in biology have inspired computer scientists to use them for many applications.

3.3.2 Modelling GRNs

GRNs are modelled in biology through a plethora of methodologies, using very different levels of abstraction and realism depending on intended purpose. I will here present a few approaches, for a good overview of GRN modelling methodologies see (de Jong, 2002). Two great approaches to the study of GRNs can be differentiated: quantitative approaches that try to design models that can predict biological gene expressions, and qualitative ones that desire to model more system wide qualities of regulation networks.

The first models of GRNs where sets of Ordinary Differential Equations (ODEs), these are mathematical models widely used in biology to model reaction kinetics (Jacob and Monod, 1961), as well as population dynamics, and many more. These methods mostly model biological GRNs quantitatively. In ODE-based models each equation of a set describes the variation of a certain chemical (protein normally in the case of GRNs) as a function of a number of other molecules. The solution to the set of equation describes the behaviour of the chemical concentrations in the cell. The parameters of these models usually are the kinetic characteristics of the chemicals involved, these can to some extent be acquired experimentally, and theoretically the models can directly be fitted against real chemical reactions.

These models have been used to model wide varieties of biochemical reactions (Voit, 2000). In the context of GRNs, they have been used to accurately model some of the best known regulatory systems in biology like the *cro-cl* switch of phage λ in *E. coli* (McAdams and Arkin, 1998), the expression of the *lac* operon in *E. coli* (Wong et al., 1997), the expression of HIV (Hammond, 1993), or the modelling of circadian rhythms (Liu et al., 2007). Even though this kind of models has been used since the 40s, there has been a lack of adequate data to fit the models in the early years of the study of GRNs. The arrival of DNA chips and other massively automated biological data retrieving systems allows increased use of this kind of systems. An other issue with ODE-based systems is the complexity of the mathematical tools need for their analysis, rarely can analytical solutions be found to solve the equation systems, so numerical simulations have to be used, and these often limit the number of equations that can be used, hence the study of the characteristics of large networks is difficult.

To alleviate the need of difficult-to-get biological data, and to be able to study the qualities and characteristics of large regulatory network qualitative models have been used from early on. The earliest, and probably most well known of these models is Kauffman's (1969) Random Boolean Networks (RBNs). This formalism is a very simple abstraction of biological GRNs, it abstracts all protein concentrations to Boolean states; either a protein is present, or absent. Each node of a RBN represents one of n genes, each of those genes has k input connection, corresponding to the output of other (or the same) genes. The k inputs are used to compute the output of that gene (which represents a protein). The functions used to compute the outputs are any Boolean function with k inputs and one output. Each node (gene) can have a different k and a different computation. A RBN is traditionally run synchronously, meaning that at each time step the outputs of all genes are computed by using the outputs of previous time step. This process is deterministic and finite, so the trajectory of the protein will settle into an attractor (possibly cyclic). The attractor is fully defined by the initial condition. The simplicity of the RBN formalism allows in depth study of the networks, their characteristics and their dynamics, this allows the study of global properties of GRNs. To achieve this, random Boolean networks are generated with fixed local properties (for example fixing k and n) and studying global properties of the network (number of attractors, for example). One of the most famous results Kauffman presented in his seminal paper (Kauffman, 1969) is that with a low k and certain choices of Boolean functions for the inputs, the number of attractors was empirically found to be about $\sqrt{n}$. The number of attractors of the network can be interpreted as the number of cell types in a multicellular organism, Kauffman argues that this is in accordance to observations in biology where the number of cell-types seem proportional to the square roots of the number of genes as well.

RBNs are still widely used due to their simplicity, a comprehensive overview is presented in Kauffman's book ((1993)). The artificial cell my model I present in chapter 4 will use a GRN model largely based on the Boolean network framework. A notable part of the most recent RBN experiments have been using evolutionary algorithms to study the evolvability of this kind of networks (Kauffman and Smith, 1986; Iguchi et al., 2005). Also Boolean networks have been more and more used to actual modelling of real biological systems (Bornholdt, 2008).

The simplicity of the model gives many advantages to the RBN formalism, such as speed of computation and simplicity of analysis, but many applications and research topics need more granularity of the simulation, hence many other models of GRNs have been developed over the last 40 years. An extension of Kauffman's model is the generalized logical method from Thomas and colleagues (Thomas, 1991), which allows variables with more then two levels and state transitions that are asynchronous. For even more granularity, many formalisms inspired by and related to artificial neural networks models have been developed (Mjolsness et al., 1991; Vohradsky, 2001). The GRN models here are recurrent networks with continuous variables, where each gene has a transfer function between the inputs from the regulator proteins and the output. These kind of models have been used extensively to model biological phenomena, but not only, they have been used as well for more abstract studies (Reil, 1999; Banzhaf, 2004; Knabe et al., 2006), and as controller for diverse software and hardware agents (Eggenberger-Hotz, 1997; Bongard, 2002; Quick et al., 2003; Kumar, 2005).

3.4 Artificial Life

Artificial Life (ALife) is the area of computer sciences purposed to implement computer systems that exhibit some characteristic of life, and study them. As opposed to biology (or computational biology), which study life-as-it-is, Artificial Life (Langton, 1995) studies life-as-it-could-be (Swan, 2009). It tries to extract from a diversity of models fundamental principles of life and its most important processes. Some of the most studied processes being self-replication, autopoiesis, and of course evolution. It also tries to study processes that are impossible or at least very difficult to study in the 'life-as-it-is" realm, such as the origin of life (Takeuchi and Hogeweg (2008), for example).

Artificial life was born almost at the same time as the computer sciences themselves. John von Neumann, through his universal constructor, showed high interests in self-replicating systems (von Neumann, 1953), and Alan Turing worked on morphogenesis (Turing, 1952). They could be called the fathers of the modern Artificial Life field, as well as fathers of modern computer sciences.

ALife is a very broad field of research, almost any area of Biology has been looked at to some extent by ALife researchers, from the origin of life to theory of mind, from biochemistry to ecosystems. The main idea is to design a model that shares some characteristics with the phenomena or system one wants to study, and run that model *in silico* multiple time while studying the effect of certain parameters, and invariants and characteristics.

This field of research has not only helped biology but also furthered engineering sciences. Many bio-inspired algorithms such as genetic algorithms, neural networks, and swarm optimization have sprung out of an effort to understand their biological counterpart through ALife.

A classical example of an artificial life model is Tierra, developed in the 90s by a biologist, Thomas Ray. Its purpose was to study evolutionary and ecological dynamics in an artificial environment as unconstrained as possible (Ray, 1991). Ray's organisms are based on computers and programs rather then on biology *per se*. So organisms in Tierra are pieces of code on a shared memory, and all organisms have their own virtual central processing unit (CPU).

One of Ray's goals was to create a model with implicit (or ecological) fitness[1] that could evolve freely, in which he could study diverse ecological and evolutionary phenomena. He was able to discover that his model evolved diverse levels of parasitism, and he was also able to study some characteristics of punctuated equilibria.

Similar models of computer models have been used (like Avida (Adami and Brown, 1994) and Physis (Egri-Nagy and Nehaniv, 2003)).

3.4.1 Multicellularity and Artificial Life

All of life on earth is composed by cells, which can be considered as small extremely complex computational units. They can interact in a huge multitude of ways, to perform just as many tasks. It would be very helpful if somehow we could use some more of the lessons from nature and apply them to engineering.

Many multicelled and multicellular systems have been studied in artificial life. Cooperation in a broad sense is pervasive in many ALife modes, one can for example consider the evolved parasites in Tierra as cooperating (certainly with my definition of cooperation), or boid flocks (Reynolds, 1987). More explicit studies of cooperation have been done, with a large body of research on the prisoner's dilemma (Hoffmann, 2000), for example. Also more biologically realist models of cooperation have been studied. For example, Takeuchi and Hogeweg (2008) in this model studies evolution of cooperating RNA-like agents replication cycles. Each agent in this simulation is a string, which represents an RNA molecule, each string has a secondary structure computed from the string with actual biological modelling tools. An RNA strand can bind with a neighbouring strand depending on how well the dangling ends of both strands match, and if they are bound one can replicate the other if its secondary structure matches a certain pre-defined (biologically relevant) pattern. Added to this, each RNA strand has a probability of decay and can move randomly to an empty adjacent square. All these are not yet models of multicellularity but represent somehow the first step to the evolution of multicellularity: the evolution of multicelled cooperation.

Another aspect widely studied in artificial life related to multicellularity is the study of developmental systems (Stanley and Miikkulainen, 2003). This

[1] Implicit fitness means that one could calculate a fitness *a posteriori*, but the amount of offspring does not depend directly from the fitness, as opposed to GAs for example, with its *à priori* defined explicit fitness function

is the study of systems that develop a body plan and/or differentiate similarly to multicellular organism. For this kind of systems one usually starts from a seed cell and one lets the system "grow" into a certain shape, or functionality.

One of the simplest and most famous developmental system is Lyndenmeyer's L-system (Lindenmayer, 1968a,b). For this system a lexical rule is used to build a fractal shape starting from a root word. This systems and variations of it have been used extensively in computer graphics to generate complex plant-like structure, and to study the morphogenesis of plants (Rozenberg and Salomaa, 1992). These kinds of model traditionally use grammatical rewriting rules to generate patterns. Another type of models called *cell chemistry approaches* in Stanley and Miikkulainen (2003) are more biological realistic. The earliest model of this type is presented in (Turing, 1952), in which he uses reaction-diffusion dynamics to model development. Cell chemistry models, as the name implies, model the developmental chemistry of biological cells to build systems that develop similarly to eucaryotic multicellular organisms. For this, traditionally, are used genetic regulatory network models, these being used in biology to control development.

Developmental systems are used mainly for two purposes: engineering multi-celled systems, and studying characteristic of differentiation. Engineering approaches of this type have mostly concentrated on evolving artificial creatures/robots, in simulation (Sims, 1994; Hornby and Pollack, 2001), and also physical environment (Lipson and Pollack, 2000), using biological development as a direct inspiration for "robot development". Also other, less direct, applications of developmental systems have been studied such as "development" of buildings (Kicinger, 2006). Some other artificial life model that have made heavy use of artificial cells similar to mine are the models presented in Marée (2000), and Hogeweg, P. (2000). They have used artificial cells and simple adhesion physics to build models of multicellular slime-molds.

Most models of multicellularity and multicelled systems have been used to model development or distributed problems, but very few have been used to study the actual evolution of multicelled systems. Noticeably an extension (Ray, 2000) of the Tierra model presented shortly in section 3.4, has been used to study the evolution tissue differentiation, in this system the environment was seeded with a handmade ancestor that is already differentiated (two CPU-"cells" each copying half of their tierra code). The number of cells of the multicelled tierra-organism can evolve and so change the level of parallelism (and efficiency) of that organism.

3.5 Contribution of my Research

In this chapter I have discussed some models inspired by biology used in computer sciences. I have presented the computer science approaches to multi-celled systems and multicellularity and the evolution of those. The goal of my work is to further the understanding of the evolution of multicellularity for the design of computational systems. I will present a computational artificial cell that will have some characteristics that will share some characteristics from the GRN and RBN based artificial cells presented here. I will then use this cell model to study explicitly the evolution of multicelled systems. In the next chapters I will present models that will address the issues discussed here and the previous chapter, computational models designed to study the evolution of multicellularity.

Chapter 4

Methods

“You can only predict things after they have happened.”

Eugene Ionesco

The main purpose of this chapter is to present a set of tools and methods to study varied questions on evolutionary theory, more specifically, evolution of multicelled cooperation and multicellularity. All the experiments presented later on will use the same algorithmic backbone: a GRN-controlled artificial cell.

In a diverse set of experiments, networks of artificial cells will be evolved for different goals and research questions.

In this section we will detail the working of the artificial cell model we use, as well as the GRN controlling it, and the genetic algorithm that will be also used across all the experiments to evolve the cells.

4.1 Artificial Cell

The core part of all this thesis' experiments is an artificial cell model. Four characteristics were important in the choice of the model's design: complexity, flexibility, efficiency and clarity. We wanted the cell to have potentially very complex behaviours and evolutionary dynamics, and to be usable in very different contexts. It also had to be computationally simple enough for it to run *in silico* efficiently, and easy enough to be understood without going into the details of the implementation.

These artificial cells are intended to expand the possible complexity of behaviour (number of possible behaviours), so as to be able to observe effects that cannot be observed easily in traditional models of evolution. To achieve this, the cells implement a complex genotype-phenotype mapping, and the control network (the GRN, encoded by the genotype) will have a wide range of possible behaviours.

Compromises have to be made in the search of complexity. One of them, issues of computational power, is purely practical. The more one makes a model complicated, the slower it will run on any given computer. Since the experiments will be involving evolutionary time scales, the cell has to run fast, or the experiments will take too long. This is one of the reasons we chose a Boolean valued GRN implementation rather than a slower continuous-valued one (Section 3.3.2). A second consideration, which is especially important in interdisciplinary research, is an issue of clarity. Many artificial life models studying evolution have a tendency towards the abstract, but in a field like this it is also important to make models with which biologists can relate without going through implementation details. We have tried to design an artificial life model, which abstracts the most relevant aspects from real biology as possible (a linear genome, a regulatory network, cells and communication proteins). The principle behind the experiments can be discussed in order to improve cooperation and mutual acceptance across disciplines.

The general setup of our artificial life model (see pseudo-code in Figure 4.1) is reasonably simple and composed of two main elements: a genome and a genetic regulatory network, the genome encoding the network. The genome will encode for the GRN with bio-inspired fashion, and the GRN will be updated at every time step of a simulation to express different ‘proteins’. In addition to this, certain proteins of the GRN will be used for different purposes (depending on the context), mainly for communication, but also for specific tasks.

4.1.1 Genetic Regulatory Network

The GRNs used for all the experiments operate as Boolean control networks. The same model has been used in (Buck and Nehaniv, 2006a, 2007), and is similar to Kauffman’s random Boolean networks (Kauffman, 1993). Our networks interact continually with their ambient environment (cf. (Quick et al.,

```
GENERIC SIMULATION
    for $i \leftarrow 0$ to $length(Cells)$
        do
            $GRN[i] \leftarrow$ BUILD GRN$(Genome[i])$
    repeat
            for $i \leftarrow 0$ to $length(Cells)$
                do HANDLE ENVIRONMENT$(Cells[i], Environment, Cells)$
            for $i \leftarrow 0$ to $length(Cells)$
                do UPDATE GRN$(GRN[i])$
            for $i \leftarrow 0$ to $length(Cells)$
                do
                    for $j \leftarrow 0$ to $length(Neighbourhood[i])$
                        do COMMUNICATION$(GRN[i], GRN[Neighbour[i]])$
        until TERMINATION CONDITION
```

Figure 4.1: Pseudo-code for any generic simulation: *Cells* is an array of all the cells in the system; the *Handle Environment* procedure takes information from the system and passes it on to the cells, it can also add or remove cells; the *Update GRN* procedure updates the protein levels of the GRN according to the system described in Section 4.1.1; the *Communication* procedure takes the communication information from neighbouring cells and updates the GRN of the cell accordingly.

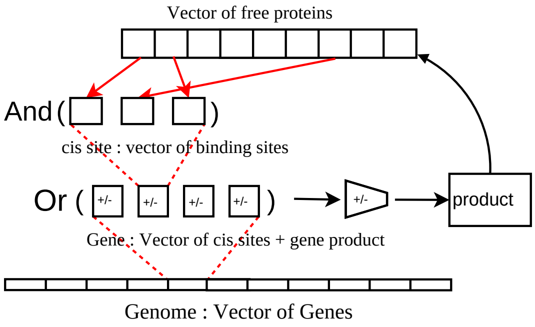


Figure 4.2: Schematic of the Boolean genetic regulatory network model

2003; West-Eberhard, 2003)), and the GRN-controlled cells interact with each other in a manner similar to that in Bull and Alonso-Sanz (2008) and many others. The GRN model is inspired by the models presented in Section 3.3.2, the layered input function we will present here are inspired by the formation of protein complexes as we describe in Section 3.3.1.

The structure of a single genome is shown in Figure 4.2. Inside a cell there are n different proteins, the level of each protein is modelled by a Boolean value reflecting its presence (*true*) or absence (*false*). The network structure is derived from the genome as described in section 4.1.2. The cell's genome consists of a string of genes, with each gene composed of a regulatory part and a part specifying its protein product as in nature (Watson et al., 2003; Davidson, 2001b).We use a two-level genetic regulatory structure (see Schilstra and Nehaniv (2008) for other models of genetic control logic). The regulatory part represents the inbound connections of the gene in the network whereas the product part represents the outbound. The inbound part (regulatory part) is structured in so-called *cis-sites*, which themselves each consist of a number of binding sites. A binding site returns a Boolean value depending on the presence in the cell of the protein it is supposed to bind. The values returned by all the binding sites of a cis-site are joined by an *AND* operator. The obtained value is then negated if the cis-site is an *inhibitory* one. Then all the values returned by the cis-sites of a gene are joined by an *OR* operator. This value is then finally negated if the gene is *default on*, if the final value of this operation is *true* then the protein encoded by the gene will be produced, i.e. the value indicating the presence of this protein in the cell will be set to *true*. If more than one gene can produce the same protein, to set the value for that protein to true for the cell, any one of them suffices. The system has a one-timestep 'memory'; at every simulation time step it takes the protein state vector of the cell in the previous step and creates a new protein state vector for the next time step using the genetic regulatory network.

Formally, for each gene of a cell's genome, we have for each protein-binding site i, potentially binding some protein p_ℓ, the present binding value b_i,

$$b_i = \begin{cases} \textit{true} & \text{if binding protein } p_\ell \text{ is present} \\ \textit{false} & \text{if binding protein } p_\ell \text{ is not present.} \end{cases}$$

The expression value c_j of a cis-site j,

$$c_j = \begin{cases} \bigwedge\limits_{\text{all i}} b_i & \text{if j is activatory} \\ \neg \bigwedge\limits_{\text{all i}} b_i & \text{if j is inhibitory} \end{cases}$$

where the logical AND-operation is taken over all binding sites b_i of the given cis-site c_j. The final protein production p_k of the gene k is

$$p_k = \begin{cases} \bigvee\limits_{\text{all j}} c_j & \text{if k is default off} \\ \neg \bigvee\limits_{\text{all j}} c_j & \text{if k is default on} \end{cases}$$

where the logical OR-operation is taken over all cis-sites c_j of gene k. The new value of p_k *for the cell* will be *true* if and only if at least one gene produces p_k. It can be shown that this system is complete in the sense of combinatorial logic: given a Boolean vector of size n (the vector of the n proteins of the cell) there always exists at least one network computing every one of the $(2^n)^{(2^n)}$ possible Boolean functions. (This can be easily seen by writing the logical function to determine the presence or absence of each protein in conjunctive normal form as function of the activation levels of all proteins in the cell, and translating this form into a genome with n default-on genes.).

4.1.2 Encoding

The encoding we chose for the networks is a highly simplified version of the encoding of GRNs in real biology (Hawkins, 1996; Davidson, 2001a). We wanted to keep a certain number of characteristics of the double-stranded DNA helix, which encodes the regulatory networks of all living organisms on earth. Our genome as in biology is composed by a very small alphabet: in nature the four nucleotides: adenine, thymine, guanine and cytosine; in our genome only two bases, 0 and 1. Our genome is sectioned as in biology by different tags that are recognised by the cellular machinery: certain combinations of bases have a certain specific meaning for the genome. There are some main differences between the encoding we use and the natural one. First our encoding is deterministic. In biology different parts of the genome can be used differently at different moments during its lifetime whereas our genome always represents the same network. The second is the fact that the biological genome is situated in a three dimensions, which can bring a high amount of modulation into the expression patterns. Another point to notice that our genome is of the single stranded sort. Of course there are many more differences but these are some structural differences which actually could be addressed in later research.

The genome is sectioned in genes. A gene is tagged by a so-called *gene tag* a pattern composed by four ones (‘`1111`’). This tag is followed by one bit to set the type of gene (‘`1`’ for *default on*, ‘`0`’ for *default off* gene) and a certain amount of bits to define the produced protein (in our experiment we used a 64 protein system so six bits are necessary to encode the binary representation for each protein). Preceding a *gene tag* is the regulatory region of that gene, that region is separated into cis-sites each one of those starting with a *cis-site start pattern* consisting of a triple zero (‘`000`’) followed by a bit for the type (*inhibitory* or *activatory*) and a certain number of binding sites (each of six bits to characterise the protein to bind at the site). Using a certain set of predetermined rules (see the pseudo-code in Figure 4.3) we can give to each bit of the genome a certain unequivocal function (even if this is merely to identify the bit as uninterpretable other than as “junk”) so as to build the GRN represented by that genome. This structure allows a genetic regulatory network to be unambiguously constructed from the genome.

The encoding is illustrated in Figure 4.4, which shows the encoding of a single gene. A genome consists of a string of such genes. The number and lengths of genes may vary between genomes in the evolving population. In the present model a gene encodes only one protein product (which can be linked to specific functions of the cell).

4.2 The Genetic Algorithm

In all three of the experiments a genetic algorithm (GA) will hold a central part; in the two first experiments (Chapter 5 and 6, it will actually be the evolutionary ‘engine’ of the experiments, and in the last experiment (Chapter 7) it will be used to design an artificial cell used as seed-cell for the main part of the experiment.

The genetic algorithm we used to evolve the single cell is a relatively standard one. It is generational, in the sense that no individuals from one time-step of the GA are carried forward to the next. The population is com-

```
TAGGING GENOME($Bitstring$)
  repeat
        $positionGene \leftarrow$ FIND NEXT($GeneTagTemplate$)
        TAG BITS($positionGene, LengthOfGeneTag$, "$GeneTag$")
        TAG BITS($positionGene, 1$, "$DefaultActivity$")
        TAG BITS($positionGene, LengthOfGene$, "$Product$")
        $positionCis \leftarrow$ FIND NEXT($CisTagTemplate$)
        while $positionCis \leq positionGene$
           do
              TAG BITS($positionCis, LengthOfCisTag$, "$CisTag$")
              TAG BITS($positionCis, 1$, "$DefaultActivation''$)
              $positionNxtCis \leftarrow$ FIND NEXT($CisTagTemplate$)
              TAG BINDINGS($positionCis, positionNxtCis, LengthOfGene$)
              $positionCis \leftarrow positionNxtCis$

    until END OF GENOME
```

Figure 4.3: How to tag the bits of the genome: The FIND NEXT(`Bitstring template`) procedure returns the position of the next untagged appearance of `template` (the gene tag or the cis-start pattern); the TAG BITS(`int pos, int size, type`) procedure tags the next untagged block of size `size` after position `pos` with the tag `type`; the TAG BINDINGS(`int positionStart, int positionEnd`) procedure tags the sub-string between `positionStart` and `positionEnd` with the type "BindingSite" in multiples of the length of genes and the left-over bits, if any, with the type "Junk".

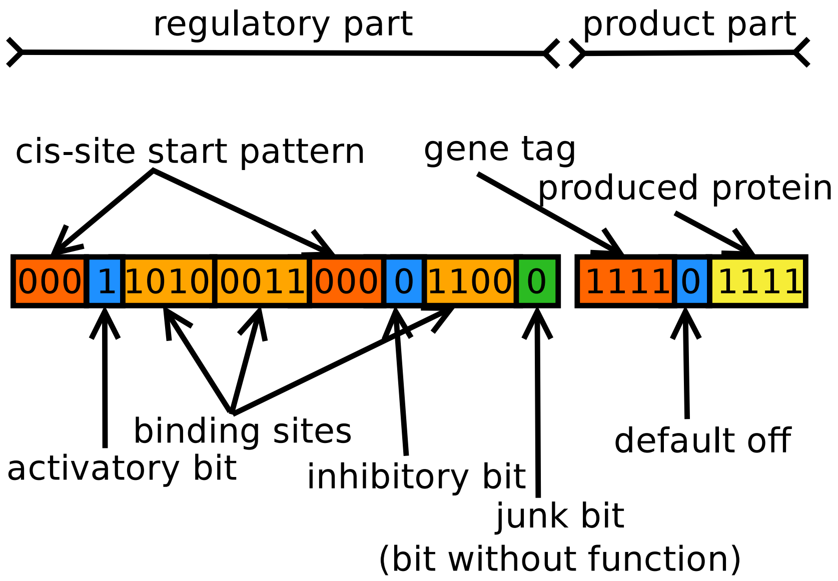


Figure 4.4: Example of the gene structure. This gene encodes a product protein 1111 and has a two cis-site regulatory region. The first cis-site is activatory and comprised of two binding sites, while the second is inhibitory and has a single binding site. The gene is off by default. Genomes are concatenations of such genes. The logical function computed by this example is $p_{15}^{t+1} = (p_1 0^t \cap p_3^t) \cup \neg(p_{12}^t)$, where p_i^t is the Boolean value attributed to the protein i at time step t.

posed by binary genomes representing GRNs. The fitness of each of these GRNs is measured as described later. A tournament-based selection is used throughout this experiment (for each pair of cells in the new generation, n individuals of the old generation are randomly selected and the best two are chosen to reproduce). For variability we use a bit-flip mutation with a constant probability (each bit of a genome has a probability p of being flipped) and for certain experiments a two-point crossover.

Various fitness functions will be used across all the experimental setups to evaluate the quality of individual genomes. We will describe them in detail in the various sections, we will here only describe the main principle of them.

In each experiment the genome will represent the artificial cell used. For the fitness functions we will always run the artificial cell or rather a network of cells with the same genome for a number of time steps. Some measures will be taken during those runs and used as fitness function. The actual measure (or measures for that matter) will depend on what the network of cells is supposed to achieve: in chapter 5 we will use an explicit fitness function fitting the quality of a colouring, in chapter 6 a two-level fitness function modelling two levels of selection, and in chapter 7 we use a an explicit fitness function to evolve “viable” cells that then will evolve freely in an environment, only constrained by implicit fitness.

4.3 Conclusion

The algorithms presented here will be used as a toolbox to be assemble different experiments for the specific problems we will address.

The two main tools in this toolbox are the artificial cell and the GA. Each is an approximation of one of the two biological components of this research.

The cell (the GRN and the genome it is mapped to) is an approximation of the a biological cell and its metabolism. The GRN as presented here will be our only approximation of the cell we use in this research, but one could have used a number of other models (see Section 3.3.2), such as neural networks approaches, or differential equation-based GRN models. We chose a Boolean GRN for multiple reasons. First, because of its intuitive, easily understood structure. A second reason being a reasonable speed of computation. The GRN model we use shares many characteristics with RBNs but its very low-level encoding allows evolution to manipulate the network more diversely then traditional RBN encodings.

The second tool, the GA, will be used in all the experiments too. But not for the same purpose in each experiment. In the first experiment (Chapter 5), it will be used purely as an optimisation method: to optimise the GRN for a specific multicelled task. In the second experiment (Chapter 6), it will represent a gross approximation of natural evolution, it will be directed by a multi-dimensional fitness function to study the fitness landscape and genotype-phenotype mapping of a simple setup for the study of evolution of multicellularity. In the third and last setup (Chapter 7); A GA will be used as a design tool to evolve a working cell to be used in an implicit fitness driven system.

In the three next chapters we will explore in detail the implementation, and results of these three experiments using the techniques described in this section.

Chapter 5

Graph Colouring in a Multicelled Environment

> "Colour, if I may say so, is biological. Colour is alive and colour alone makes things come alive..."
>
> Paul Cézanne

This work is an exploratory investigation to study the computational power of my multicelled artificial cell model, and will also be used to study different inter-cellular communication protocols. Another aspect studied is how scalable our artificial cell model is in respect to the number of proteins controlling the GRN.

We wanted to get an idea of what cells could achieve in a multicelled setup using classic optimization problems to create problem specific cells. In this setup we will use an optimised cell at each node of the network we want to colour. Each cell will have the same genome (hence GRN). Then the simulation is run and certain proteins of the GRNs will represent the colour of the cell.

The experiment is NOT designed to find very good solutions nor to challenge any other algorithm performing similar computation. It will be used as *a proof of concept: to prove that in a quite general kind of setup we can evolve multicelled behaviours*. The opportunity is also used to study communication protocols, as all our experiments are in a multicelled environment, cells need to communicate. There are multiple possible choices for how this communication is implemented: problem specific or general protocols, addressed or not.

Wanting to have a cell model usable in diverse setups, with variable connectivity (see Section 4.1), we did not study addressed protocols. Even though we did consider one problem specific protocol, so we could compare it with the two general purpose protocols we were actually interested in.

All three communication protocols will be studied in setups with different number of proteins in the GRN. If our cells are to be used in further studies, one needs to know how the number of proteins controlling the GRN impacts on the evolvability of the cells. An increase in the number of protein gives potentially a higher computational power to the cell, but it can also reduce dramatically the ease in which this computational power can be used and evolved. So this experiment is also used to get an idea of the behaviour of the system with increased number of proteins.

In Schwefel and Kursawe (1998), the authors present some of the advantages for using multicellularity for optimization. They do however use self-adaptation of mutation rates. We have not used this due to the great plasticity of our genome. One bit does not represent, in our model, a fixed attribute (like a single gene) and does not directly link to the fitness such as in Schwefel's model. It would be however interesting in the future to devise a complex self-adaptation system for the mutation rate, that could for example reduce the mutation rates of important regulatory hubs in the network.

Once an idea of the computational power, scalability and some reasonably satisfying communication protocols have been established further experiments on the main questions of this thesis can be studied in the following chapters.

5.1 Extension of the Methods

5.1.1 The Graph Colouring Problem

The graph colouring problem is a very well known combinatorial NP-complete problem. To colour a graph each node of the graph has a colour assigned to it and none of its immediate neighbours is allowed to have the same colour. To decide whether there is a way to colour an arbitrary graph using k colours is NP-complete for $k \geq 3$. It is in fact one of the 21 NP-complete problems described by Karp (Karp, 1972). This problem is important for numerous real life applications including: map colouring, radio frequency allocation, register allocation in compilers (Mueller, 1993) or scheduling (Marx, 2004). This problem has been approached with numerous different types of algorithms (Biggs, 1990; Costa and hertz, 1997; Prestwich, 1998; Shawe-Taylor and Zerovnik, 1995), mostly heuristic local optimisation algorithms, with most of these methods being far more effective than our approach to it. But the goal of our work is not to make an especially effective method to solve this problem; rather, we will use the graph colouring as a first test bed for a new cell based computational model and study the evolvability of different cell-to-cell communication protocols.

The graph colouring problem is very adapted to our investigations in being simple to understand but still combinatorially complex. Also its inherent local nature is very suited to the structure of a multicellular distributed approach.

5.1.2 Simulation

As benchmark graph colouring problem instances, the simulation environment can use any graph, we will use the *myciel7.col* and *miles250.col* graphs (191 nodes, 2360 links; and 128 nodes, 774 links, respectively) which we retrieved from the COLOR04 website[1] and which both can be perfectly coloured with a minimum of 8 different colours. Each node (artificial cell) of a graph is controlled by the same GRN; all the cells are initialized to the same state (protein levels all set to 0 (false/absent) in our case). The simulation has its own time frame independent from the evolutionary time frame. It is updated randomly: every cell of the simulation space is updated once per time step but at each time step in a different random order. This is the only stochastic part of the graph colouring simulation therefore very important for the setting up of patterns. With all the cells being initialised to the same state at the start of a simulation run, we need some randomness for them to achieve colour differentiation. Each cell is updated in the same way following a specific order of events :

Update of the GRN: the GRN in each cell is updated by one time step and new values for the protein levels in that cell are computed following the system dynamics described in section 4.1.1

[1] http://mat.gsia.cmu.edu/COLOR04/

Handle communication: the levels of communication proteins are checked and associated protein levels accessible to neighbours are updated (see section 5.1.2.1)

Update of states: the colour of the cell is updated (the state of a certain number of proteins taken together encode the number representing the colour)

Each of those steps will be repeated for each cell in the simulation space for the number of time steps required by the simulation.

5.1.2.1 Communication

For this set of experiments we will use three different types of communication protocols between our artificial cells.

The *OR*-unconstrained protocol: Each cell has a set of m different *emitting* proteins and m corresponding *receptor* proteins. If one of the emitting proteins is set to *true* then the receptor protein value of all its direct neighbours is set to *true*, if a receptor protein of the neighbours is set to *true* already it stays in that state. Thus the new value of receptor protein of the number is the logical *OR* of its current value and the emitter's value for that protein.

With this communication protocol the system can evolve relatively freely the way it uses a certain number of predetermined communication proteins. It is also totally problem independent, so if the same system would be used in other environments this communication protocol can still be used. It is inspired by the close range diffusion and receptor systems existing in biology.

The *XOR*-unconstrained protocol: This protocol is very similar to the previous one. It is virtually the same except for the last step: if a neighbour's receptor protein is already set to *true* it is switched to *false*. Thus the new value of receptor protein of the number is the exclusive *XOR* of its current value and the emitter's value for that protein. According to information theory this protocol should have a higher information transmission capacity[2], and hence perhaps able to evolve more complex communication.

Constrained protocol: This communication protocol is problem specific and the cell already transmits encoded information. The cells here have only eight receptor proteins. A cell transmits to its neighbour directly its colour i.e. switches to *true* the receptor protein of all its neighbouring cells corresponding to the emitting cell's colour.

We have made a conscious choice of not using any addressing protocols. With the number of neighbours not being the same for all the cells, the system would otherwise lose too much of its generality.

5.1.3 Genetic Algorithm

The genetic algorithm used is a relatively standard one. The population is composed by binary genomes representing GRNs (section 4.1.2). The fitness of each of these GRNs is measured (section 6.1.1.1) by running colonies of GRN-controlled cells in the simulation environment described in section 5.1.2. A tournament-based selection is used throughout this experiment (for each pair of the new generation 5 individuals of the old generation are randomly selected and the best two are chosen to reproduce). For variability we use a bit-flip mutation with a constant probability (each bit of a genome has a probability of 0.002 of being flipped) and a two-point crossover (section 5.1.3.1).

5.1.3.1 Crossover

The crossover operator (providing recombination of genetic material from two genomes) is always a subject of discussion in evolutionary computation (Jansen and Wegener, 2005). The difficulty is to give to the crossover a sense, where this sense usually has to do with structure in the genome. In an unstructured genome, where each bit of the genome has no functional relationship to the neighbouring bit, swapping stretches of genetic information from one individual to another makes little sense as exchanges are likely to disrupt any possibility for evolved structure in the genetic encoding. On the other hand, if

[2] The *XOR* loses less information then the *OR*.

the genome is structured and that the stretches of information passed are structured meaningfully, then it can be useful to recombine meaningful stretches of genetic information.

Our genome being structured we chose to include a crossover operator. We chose to use a two-point crossover, meaning that we choose two points at random in the genomes of the two parents and switch the piece of genetic information between those points between the two parents to create two offspring. Actually the points chosen are not totally random: to make the pieces of information sensible the points have to be either a *gene tag* or a *cis-site start tag*. We chose a two-point crossover as the way it has been implemented it enables one to emulate a variety of biologically plausible scenarios like gene duplication, gene transfer or one-point crossover, which are important in evolution (Ohno, 1970). Moreover the way our GRN model is implemented, a gene duplication is not deleterious (which it may well be in other GRN models, especially in the continuous ones cited earlier), therefore it seems possible that it could help facilitate interesting courses of evolution.

5.1.3.2 Fitness

The fitness of a specific GRN is measured with the run of a certain number of simulation steps on the network graph. Every cell of the graph is endowed with that GRN and then run for a certain number of time steps. At each time step, each cell gets a cell-fitness of *one* if all of its neighbours are in a different colour than itself, otherwise it gets a value of *zero*. A network-fitness is then computed by averaging all the cell-fitnesses and represents the fraction of cells whose colour differs from all of their neighbours. The best network-fitness over all the time steps, representing the best colouring achieved by the genome for the graph over the run, is the fitness of the genome for that run. To lower slightly the effect of the randomness the fitness of five simulation runs are averaged to finally compute the fitness of that GRN at this generation of the evolution.

5.2 Experiments

We will in this set of experiments study the reaction of our system to the different communication protocols as well as to different number of proteins on the two graphs we chose (*myciel7* and *miles250*, both of which can be coloured with 8 colours). The number of proteins contributes to determining the size of the search space of different possible behaviours the GRNs can have; the more proteins in the system the more complicated the behaviour of the GRN can be; this influences also the way a multicelled colony can use the information gathered by the (local) communication system.

The base experimental conditions for the genetic algorithm are: population size of 50, tournament size of 5, 1000 evolutionary generations, 250 simulation time steps, mutation rate of 0.002 and crossover rate of 0.3.

We have 9 (times two) experimental set-ups, for each set-up we have 9 evolutionary run, two for each of the 3 communication protocols (8 emitting and receiving proteins, only 8 receiving for the constrained protocol), each with 32, 64, 128 and 512 different proteins in each cell, less then 32 proteins is not possible as already 21 proteins are reserved (8 for emitting, 8 for receiving, 3 for the colour).

We run t-tests to quantify the effect of the increase of protein, so the p-values in Tables 5.1 and 5.2 represent the significance of the increase of proteins from the previous column (if the p-value is smaller then 5% the increase of number of proteins since the previous column is significant).

A test run for each parameter set has also been run. The test runs are similar in construction to the normal runs, but the GA has been replaced by a random search algorithm (in our case the GA with a mutation rate of 0.5, e.g. randomising the new genomes entirely).

5.3 Results

Table 5.1: Fitnesses achieved for the *miles250.col* graph

Communication protocol	*OR*-unconstrained				*XOR*-unconstrained				constrained			
Number of proteins	32	64	128	512	32	64	128	512	32	64	128	512
Maximum fitness	0.83	0.82	0.65	0.25	0.68	0.68	0.67	0.23	0.74	0.63	0.63	0.14
Average fitness	0.73	0.71	0.51	0.24	0.67	0.67	0.62	0.22	0.69	0.61	0.52	0.09
Standard deviation	0.09	0.09	0.11	0.01	0.01	0.01	0.08	0.01	0.07	0.03	0.09	0.01
p-value	N/A	64%	0.6%	<0.1%	N/A	99%	8%	<0.1%	N/A	0.6%	1%	<0.1%
Random search	0.45	0.39	0.32	0.12	0.58	0.49	0.16	0.13	0.49	0.48	0.35	0.10

Table 5.2: Fitnesses achieved for the *myciel7.col* graph

Communication protocol	*OR*-unconstrained				*XOR*-unconstrained				constrained			
Number of proteins	32	64	128	512	32	64	128	512	32	64	128	512
Maximum fitness	0.84	0.82	0.75	0.14	0.48	0.48	0.40	0.09	0.88	0.88	0.86	0.15
Average fitness	0.77	0.77	0.65	0.12	0.43	0.41	0.38	0.08	0.81	0.82	0.70	0.13
Standard deviation	0.04	0.03	0.09	0.01	0.03	0.04	0.02	0.01	0.07	0.03	0.15	0.01
p-value	N/A	99%	0.2%	<0.1%	N/A	24%	6%	<0.1%	N/A	70%	3%	<0.1%
Random search	0.53	0.51	0.33	0.11	0.25	0.22	0.19	0.05	0.66	0.62	0.49	0.12

The results of the experiments are compiled in figures 5.1 to 5.2. The values for each experimental setup are the average of the best fitness achieved by each 9 evolutionary runs. The *max* line values are the best fitness achieved over all the 9 evolutionary runs in a given experimental condition.

Overall the *OR*-unconstrained and the constrained protocols have qualitatively similar results and showing similar up-scaling issues. The *OR*-unconstrained protocol being slightly better for the *miles250* graph and the constrained one in the *myciel7* graph. These differences might be an issue in a pure optimisation approach but not in our problem. These differences are largely outweighed by the greater usability and versatility of the *OR*-unconstrained protocol.

The *XOR*-unconstrained protocol performs very well on the *miles250* graph, but its performance for the *myciel7* graph was poor, even with its higher informational content. This protocol can be very good in specific graph topologies, or very bad in others, which is not a desirable characteristic for the desired versatility we are looking for.

For all cases, except the constrained protocol in *miles250.col*, there is no significant difference between the use of 32 and 64 proteins. Also in all cases, except the *XOR*-unconstrained, there are significant differences between 64 and 128 proteins, and in all cases between 128 and 512.

5.4 Conclusion

In this study we have used the graph colouring problem to study the evolvability of different communication protocols and scalability of my artificial cell in a multicellular computational environment.

The results show that, at least in this setting, the differences between the constrained and the OR-unconstrained communication protocol are small enough so as to be outweighed but the gain in generality. This allows us to use the same (*OR*-unconstrained) communication protocol for all subsequent experiments. Hence not restraining the cells with problem specific protocols.

Also the GRN model is quite robust to the increase in number of proteins from 32 to 64, but there is a tendency of decreased quality on average when the number increases to higher number.

A note of caution has to be stated here though. As the results with *XOR*-unconstrained protocol show, only a change of topology of the network (of the graph to be coloured) is enough for the results to drop dramatically.

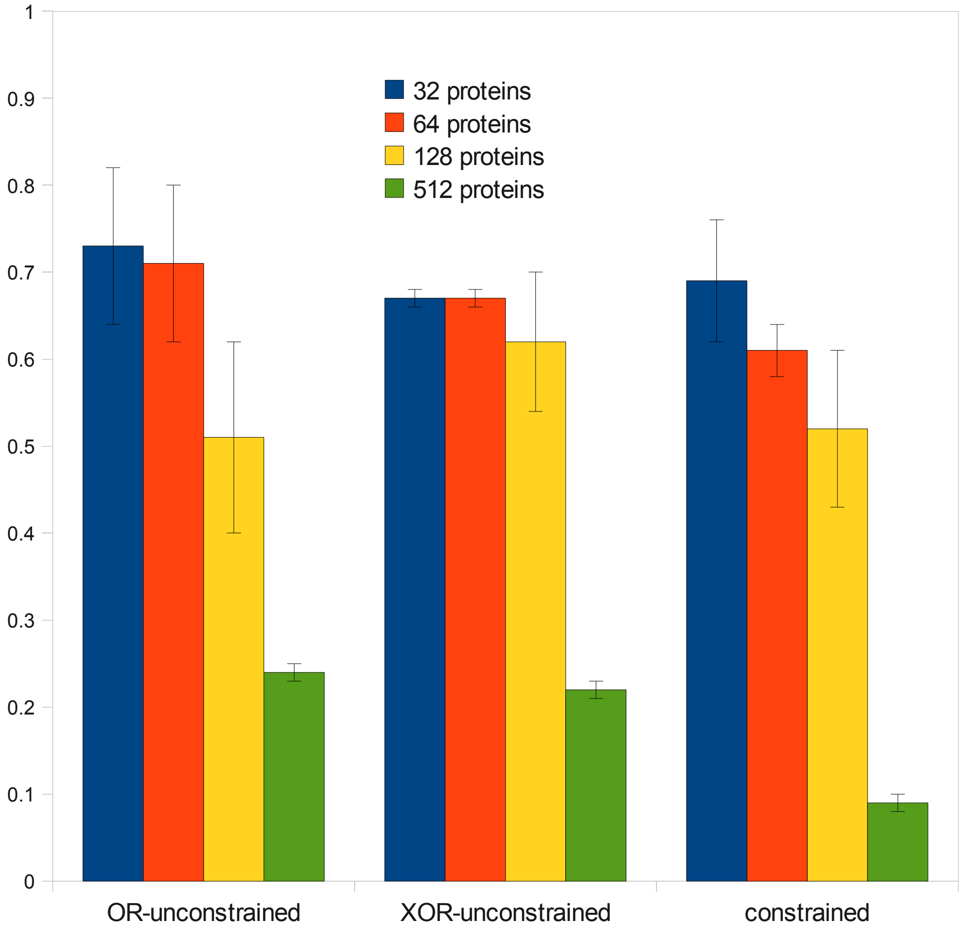


Figure 5.1: A graphical representation of the results of Table 5.1 (*miles250.col* graph)

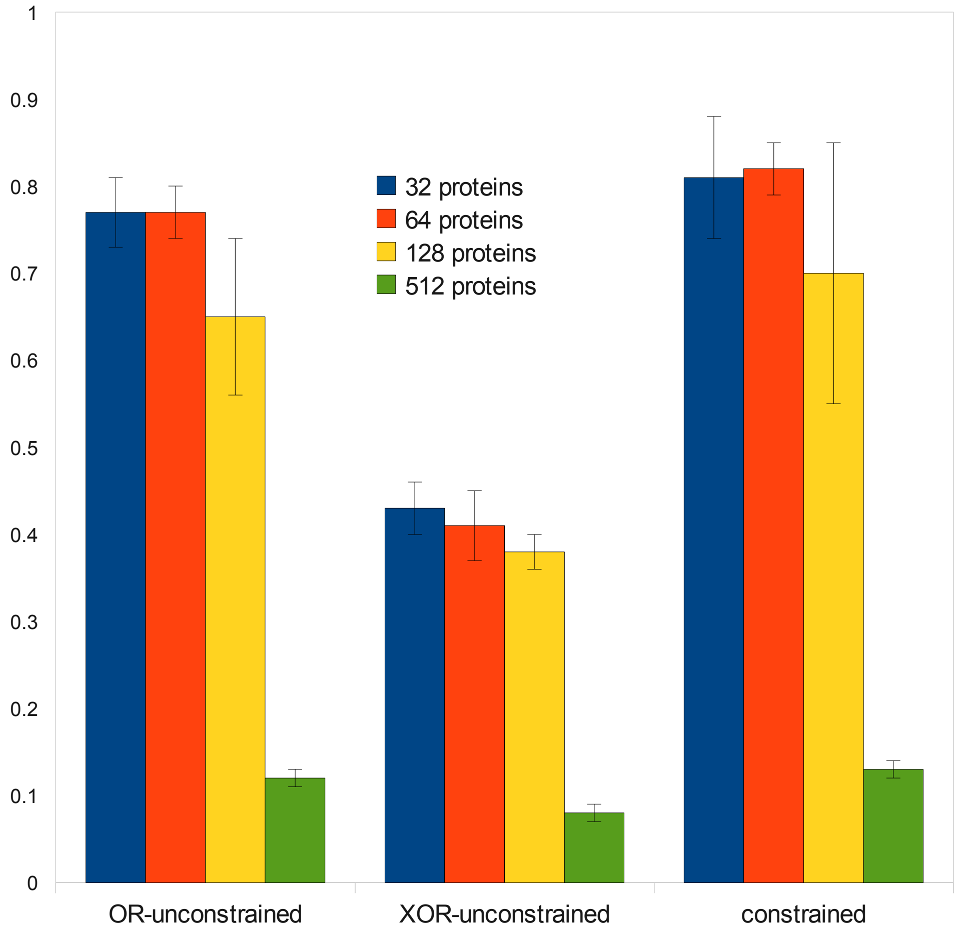


Figure 5.2: A graphical representation of the results of Table 5.2 (*myciel7.col* graph)

This shows that even more general protocols can be in a certain sense highly problem specific.

The fact that the communication protocols efficiency is dependent on the topology of the problem highlights one of the major problems of these classical optimization algorithms: adaptive. Even though GAs and other metaheuristics, are adaptative in the sense that they can be easily adapted to a variety of problems by just changing the fitness function (or its equivalent), the solution of one evolutionary run for a specific problem will not work for any other problem, or even the same problem with a different setup.

One of the main hopes in the development of highly distributed systems is to create systems that can adapt to a variety of problems and topologies, one solution would be of course to use a system similar to the one described in this chapter and design a fitness function that combines the whole variety of problems the system might encounter. For this, of course, we would need to be able to predict what the system will encounter, this might not always be the case.

The solution we interested in for the rest of this thesis is the idea of having distributed systems similar to the ones find in biology: multicelled entities. A cooperating colony of bacteria for example can adapt to its environment through evolution, the problem then becomes how to evolve/design such an artificial multicelled (or even multicellular) system? This has many problems some of them being discussed in chapter 2, and these problems have driven the rest of the research presented in this thesis.

Chapter 6

Checkerboard Colony

> "In the animal world we have seen that the vast majority of species live in societies, and that they find in association the best arms for the struggle for life: understood, of course, in its wide Darwinian sense — not as a struggle for the sheer means of existence, but as a struggle against all natural conditions unfavourable to the species."
>
> Peter Kropotkin

After having evolved a cooperating multicelled system and noted that classical optimization methods are not perfect for the development of the kind of systems we are interested in, we wanted to work more in detail on the question of evolution of multicellularity and multicelled systems. And before starting a fully-fledged open evolutionary system, we wanted to experiment first in more constrained environments.

By more constrained environments we mean an explicit fitness driven systems, like the GA in the previous chapter, but a GA tailored to the study of the evolution of multicelled entities. These systems have one major advantage: they are quantifiable. Due to the explicit nature of the fitness, runs can be "judged" and compared and the effect of every parameter change can be studied easily. Also if a GA is used, another benefit is its speed of optimization, they tend to be faster and more effective than ecological evolution.

We will hence use in this experiment a GA to evolve some sort of multicelled interaction starting from individualistic cells. We will also use this experiment to explore a bit further the behaviour of our setup. We will use it to study the nature of the genotype-phenotype mapping of our artificial cell model.

As a first approximation the nature of the evolution of multicellularity is a classical problem of optimization (Kursawe, 1993; Schwefel and Kursawe, 1998). It can be represented as a fitness landscapes with two main fitness peaks: one of individualistic cell behaviour, and one of cooperative cell behaviour (where cells need to interact to be fitter). The main question being: "how to get from one peak to another?". This depends a lot on the shape of the fitness landscape, and the shape of the genotype-phenotype mapping. The dependency on the fitness landscape is quite trivial, but the importance of genotype-phenotype mapping might need some explanation.

What is meant by genotype-phenotype mapping? In our model the genotype is a string of Booleans, and the phenotype is the protein levels of a GRN over time. I will use this to illustrate the notion of phenotype-genotype mapping. If one mutates Booleans in the genotype, it can have an impact on the network, but not every mutation will have the same impact, and also the way the genotype maps the phenotype influences the impact of mutations. The effects can be of various amplitudes, changing the dynamics of the network gradually or directly. Also their can be an imbalances in the effect of mutation: the effects of mutations can be similar for every Boolean of the genotype, or very different for certain positions. As the encoding we use for the artificial cells encode also for the number of proteins and connection a single mutation (breaking a gene-tag, for example) can have drastic changes.

The ease with which one can go from one fitness peak to the next one depends directly on the shape of the fitness landscape and the genotype-phenotype mapping. In this experiment, we implemented two levels of fitness, one requiring a higher level of organization requiring inter-cellular cooperation (the formation of a checkerboard pattern), and an individualistic behaviour. The peak for *individual behaviour* will be very flat and lower (or equal, the height of this peak will be a parameter) than the *cooperative behaviour* peak, which is narrower and higher. In this situation if the genotype-phenotype mapping is too "soft" (effect of mutations are small) evolution might never leave the flat peak of individuality, whereas if it is to "rugged" (mutations have dramatic effects), the risk is that evolution finds the peak but loses it again before stabilizing correctly[1]. So we hope that our genotype-phenotype

[1] We have not put any elitism into the system, in nature the fittest individual is not kept

mapping have some elements of both types: the capacity of moving around across the fitness landscape with mutations that have a big effect on the phenotype, yet not every mutation should have these big effects so that the GA can explore the area around interesting phenotypes without risking to lose the peak.

To do this study, we will use a GA to evolve cells in a similar fashion to the experiment in Chapter 5. To measure the fitness of a genome it will control the GRN of cells in a grid. We will then use a two-level fitness function to measure the quality of that genome, each of the two levels representing one of the peaks in the fitness landscape. We will vary the height of the peaks and the population size of the GA. With this setup we can find out under which circumstances evolution will go towards the higher peak of cooperation and when not, and also find out some useful information about the topology of the genotype-mapping.

6.1 Extensions of the Methods

6.1.1 Simulation

The simulations takes place in a 2D toroidal grid, were each cell only considers his four direct neighbours. Each position of the grid is occupied by an agent (cell) controlled by a GRN. All of the cells in the grid have the same controlling GRN. We use in this experiment GRNs with 32 different proteins, therefore each cell can be in one of 2^{32} different states, but not all of these proteins have an actual effect on the environment most of them are internal states used to control the cells.

This architecture gives the cells the potential to communicate. The communications is the *OR*-unconstrained protocol from the previous section with $m = 4$.

The cell can be in three possible "visual" states, two of them being "cooperative" and one "individualistic" state. One protein controls the "individualistic" state, if it present in the cell this cell is in that state, if it is not it is in one of the "cooperative" states. Those states are controlled by another protein, if it is present the state will be "red" else "green" (these are both two cooperative states). Those different states are independent of the

alive artificially either.

communication, an “individualistic” cell can still communicate and receive communication, the “visual” states are used during the computation of the fitness function.

The regulation networks, the communication and the “visual” states are updated in a random synchronistic way. The cells are updated in a random order but each cell one and only one time during each time step. This is the only non-deterministic component of the simulation. Each simulation will have a finite fixed number of time steps.

6.1.1.1 Fitness

Developing an environment with a natural (implicit) fitness is not easy and usually needs many parameters. Therefore we chose to work with an explicit fitness function. This fitness here has the particularity to be actually two fitness functions representing two levels of selection, one trying to reach a high level goal needing multicelled interaction and one representing a low level single cell goal, both goals being exclusive (computed independently), so both goals are in competition.

The lower level fitness is simply to stay as long as possible in the “individualistic” state. We check for each cell in the grid which cell has stayed longest in that state and normalise that time to 1. If $t_{\text{ind}}(i)$ is the time cell i has spend in the “individualistic” state, the “individualistic” fitness F_{ind} of a GRN in a certain simulation is

$$F_{\text{ind}} = \frac{\max\limits_{\text{all cells i}} t_{\text{ind}}(i)}{t_{\text{sim}}},$$

where t_{sim} is the length of a simulation.

The higher-level goal is to create a checkerboard with the “red” and “green” cells. At each time step of a simulation, for each cell of the grid in a “cooperative” state we check the neighbourhood, for each of the neighbouring cell which is in a different state but not individualistic that cell gets a score of 0.25 (remark : 0.25 is 1 divided by the number of neighbours 4). So at each time step each cell can get a score between 0 and 1. Those scores are then summed for each time step over all cells and normalized to 1. If $n_i(j,t)$ is equal to 0.25 if the j^{th} neighbour of cell i is in the same state than cell i but not the individual one at time t, else 0, $f_{\text{check}}(i)$ the fitness of cell i is

$$f_{\text{check}}(i) = \begin{cases} \frac{1}{t_{\text{sim}}}\sum_{t=1}^{t_{\text{sim}}} \sum_{j=1}^{\text{neighbours}} n_i(j,t) & \text{if i cooperative} \\ 0 & \text{if i individualistic} \end{cases},$$

hence the higher level fitness of the GRN after a simulation F_{group} is the average of f_{group} over the colony

$$F_{\text{check}} = \frac{1}{n_{\text{cells}}} \sum_{\text{all cells i}} f_{\text{check}}(i),$$

where n_{cells} is the total number of cells in the grid.

The final fitness of a GRN will be the maximum (For complete separation of the two level, we did not want a weighted system to limit the possible of intermediate solutions) between the higher level and the lower level fitness weighted by $\alpha \in [0, 1]$, a parameter weighting the advantage/disadvantage of being individualistic. So the fitness F of a GRN lies in the interval $[0, 1]$ and is

$$F = \max(F_{\text{check}}, \alpha \cdot F_{\text{ind}}).$$

6.2 Experimental Investigation

We have for this experiment run a 10 GAs (mutation rate: 0.002, cross-over rate: 0.5, starting genome size: 1000 bits, size of tournament: 25, size of the grid: 6 × 6, length of simulation: 30). The values of α studied were between 0 and 1 included in steps of 0.1, and the population sizes 125, 250, 500, and 1000. An α parameters set to zero meaning that there is no contribution of to fitness from the individualistic fitness, the evolution is only driven by the high level fitness. We have done the same experiment for three different length of GA, 200 and 1000 generations, and an experiment with 200000 fitness evaluations (which is equivalent to 1600 generations for population size 125, 800 for population size 250, 400 for population size 500, and 200 generations for a population size of 1000).

The smaller α, the higher is the incentive for the cellular colonies to evolve cooperation because the reward of cooperation is so much greater then simple non-cooperation.

Also we have used the OR-unconstrained communication protocol described in Section 5.1.2.1.

In this experiment we are not directly interested in the actual fitness achieved, rather we are interested in the local optimum in which an evolutionary run stabilizes. There are, as mentioned earlier, two local optima, one for individual behaviour (shallow peak), and one for cooperative behaviour (steep peak), the steep peak being always higher or equal to the shallow one. The shallow peak's height is characterized by the parameter α, so any GA run that has stabilized on a fitness value above α has certainly achieved some degree of multicellular cooperation. So for each set of 10 GA-runs we have computed the proportion of runs that have achieved this, we will call this the proportion of multicellularity, and this is the value plotted on the graphs of Figure 6.1 to 6.6. This proportion of multicellularity is an approximation of the probability that an evolutionary run with a set population size will stabilize on multicellular behaviour in a set number of generations.

6.3 Results

Figures 6.1 to 6.6 are the results of this experimental setup, and figure 6.7 shows some picture of resulting behaviours.

The first remark is that for most of the plots one can notice a phase transition. Only for the plots with a population size of 125 it is not obvious. This signifies that there is a tipping point at which the behaviour of the evolutionary algorithm changes. Before that point evolution has a very high probability of reaching a multicellularity and then, for a very small increase of α this probability tends to zero. The dependence of the tipping point on the population size is slightly unclear, in figures 6.1, 6.2, and 6.3, one can see that the tipping points for population sizes 250 and 500 are very close, yet for population sizes 125 and 1000 they are respectively lower and higher. One has to be slightly careful, with the analysis of figures 6.1 and 6.2, because as the number of generations is fixed and the population size in not the same for every line, the number of fitness evaluations for each line of the plots are different. Naturally a GA with a smaller population size will take more time (generation-wise) to explore the fitness landscape. For this purpose we have included the results of figure 6.3, where all the GAs could take the same amount of sample points in the fitness landscape (the same number of fitness evaluations), but one can see that the resulting plot is qualitatively similar to the two previous ones.

In figures 6.4 and 6.5, we have presented some of the same results but with a fixed population size, and varying number of generations. We can see that qualitatively the lines are the same, hence the number of generations does not matter for the phase transition, or at least for the explored parameter space. This means that the minimum number of generations we have picked (200) is enough for the GA to get to a stable point.

This result allows us to compute figure 6.6, which is a combination of the previous graphs. We recomputed every point of the graph using the data from figures 6.1 to 6.3, without considering the number of generations (basically, supposing that all the GA-runs had been stopped at the same number of generation, or at stabilization). This allows figure 6.6 to have a better definition on the vertical axis.

We can still, in figure 6.6, notice the phase transition, the two curves for population sizes 250 and 500 that are very close, the line for a population of 1000, that drops a bit later, and the one for a population of 125 that starts to drop already for small values of α.

6.4 Conclusion

What can we conclude from these results? The results from this experiment are a bit mitigated, but still can conclude a certain number of points.

First, there tends to be a clear phase transition for population sizes above 125[2], meaning that there is a non-linear shift of the evolutionary behaviours of the GAs. Both evolutionary attractors (individuality and cooperation) have clearly defined domains of attraction depending on α. We are supposing that a colony's fitness can always be higher if cooperating, than if not, this transition shows, that even though the higher fitness would always push towards cooperation, due to the combination of a complex fitness landscape and genotype-phenotype mapping, this high fitness is not always achieved. Even more the behaviour on which the evolutionary runs stabilize seem to be in an almost deterministic way depending on a set of parameters. One could consider α an environmental parameter defining the difficulty

[2] This is probably due to the fact that we have used a constant size for the tournament of the selection procedure, this changes the selection pressure for small population sizes.

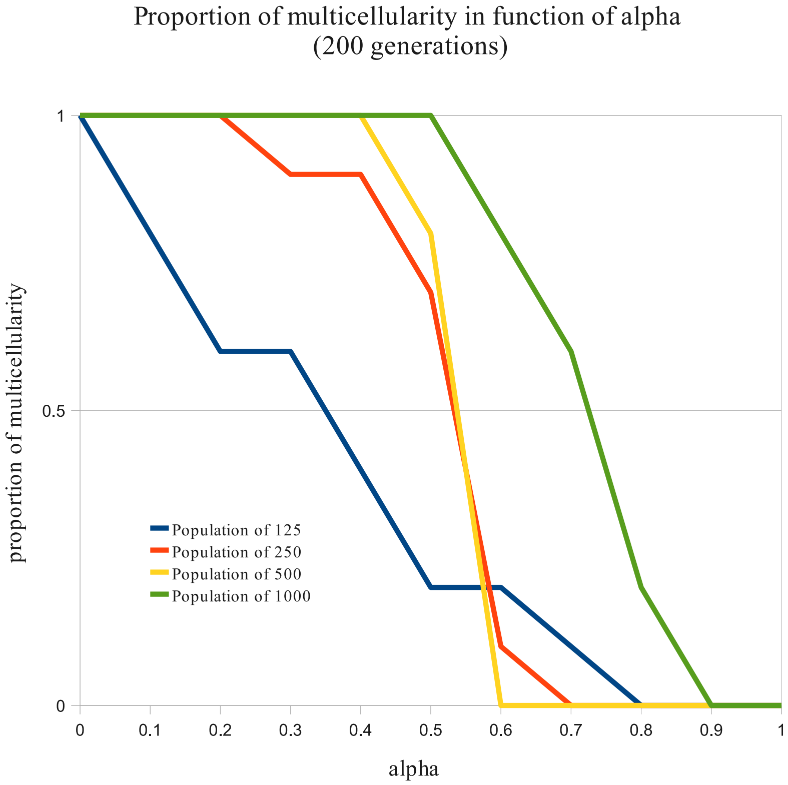


Figure 6.1: Proportion of evolutionary runs that have stabilized on the multicellular state after 200 generations, for different values of α.

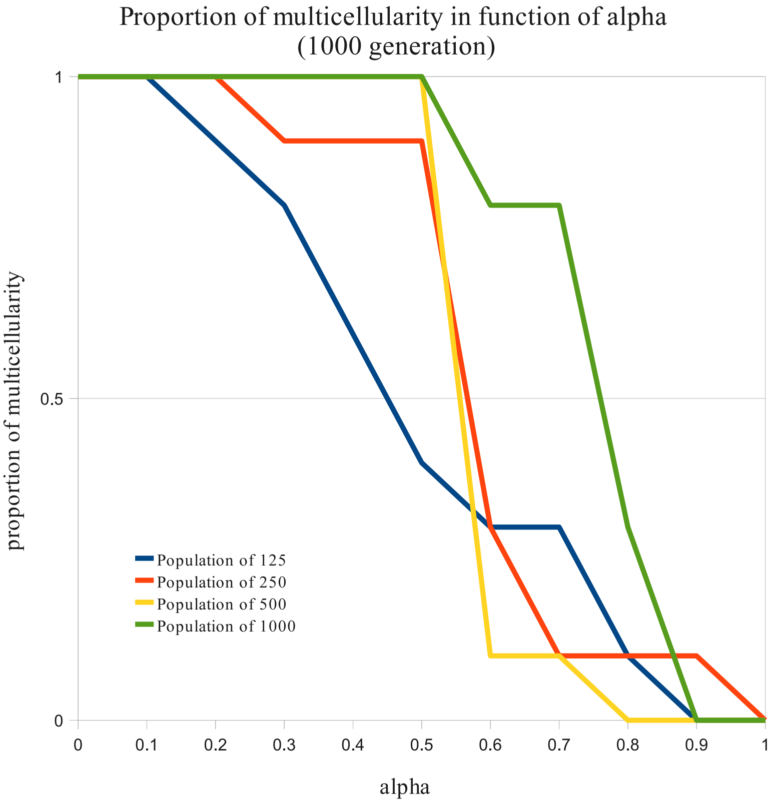


Figure 6.2: Proportion of evolutionary runs that have stabilized on the multicellular state after 1000 generations, for different values of α.

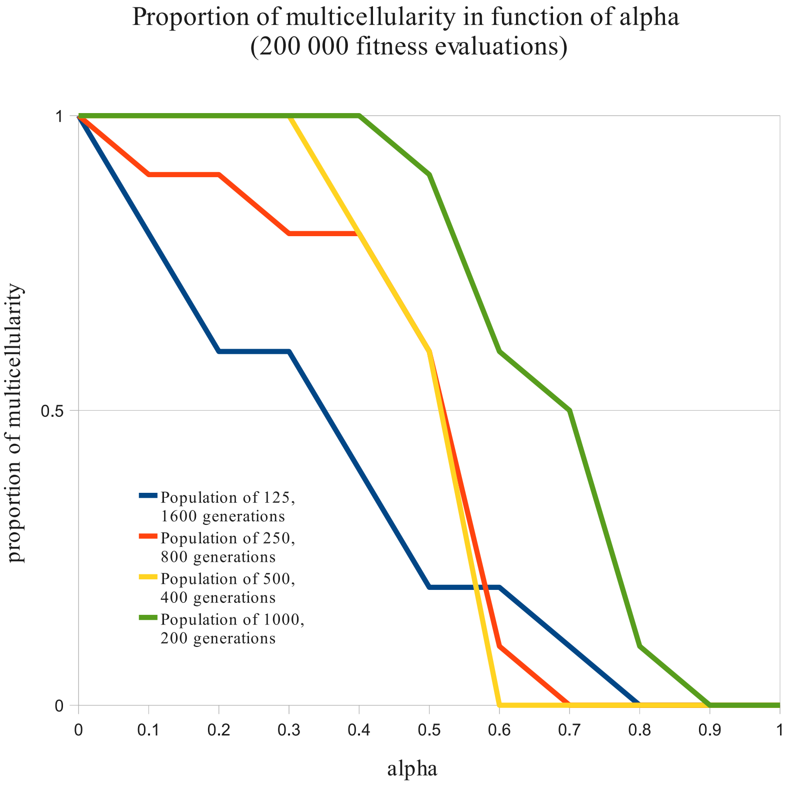


Figure 6.3: Proportion of evolutionary runs that have stabilized on the multicellular state after 200000 fitness evaluations, for different values of α.

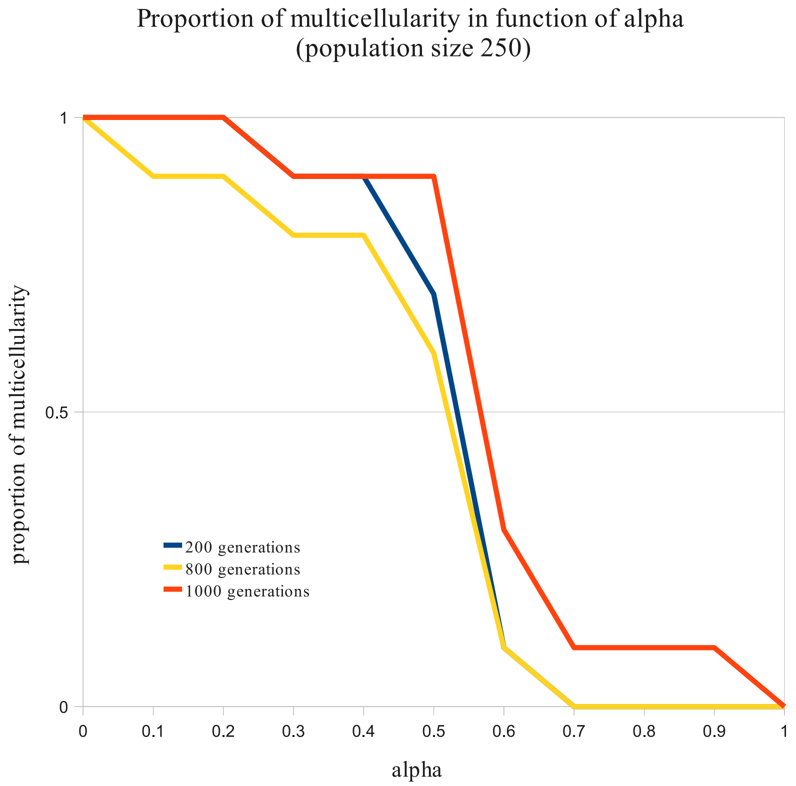


Figure 6.4: Proportion of evolutionary runs that have stabilized on the multicellular state for varying number of generations, for different values of α, for a population size of 250.

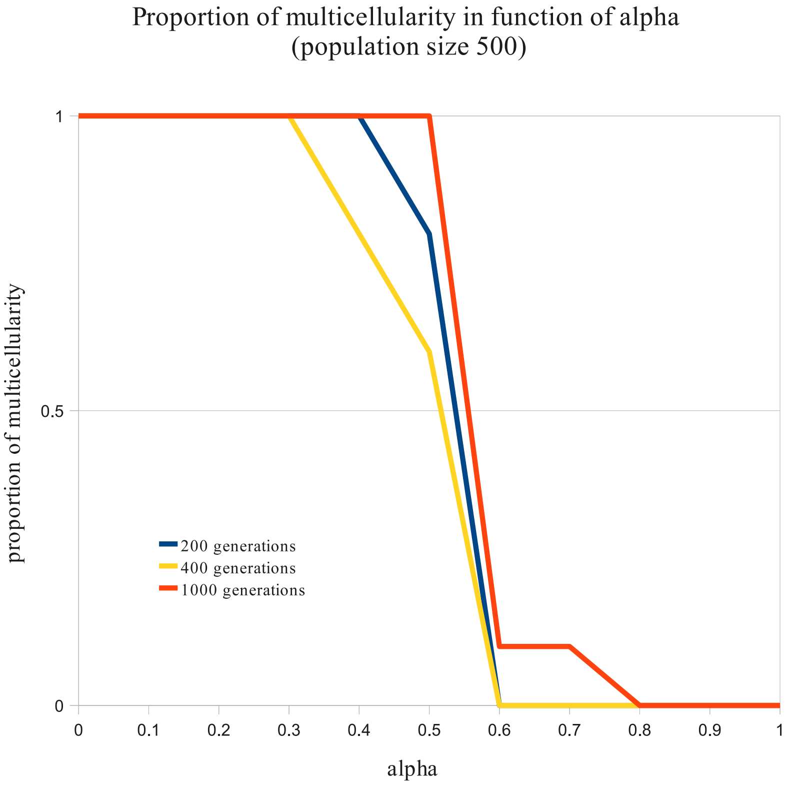


Figure 6.5: Proportion of evolutionary runs that have stabilized on the multicellular state for varying number of generations, for different values of α, for a population size of 500.

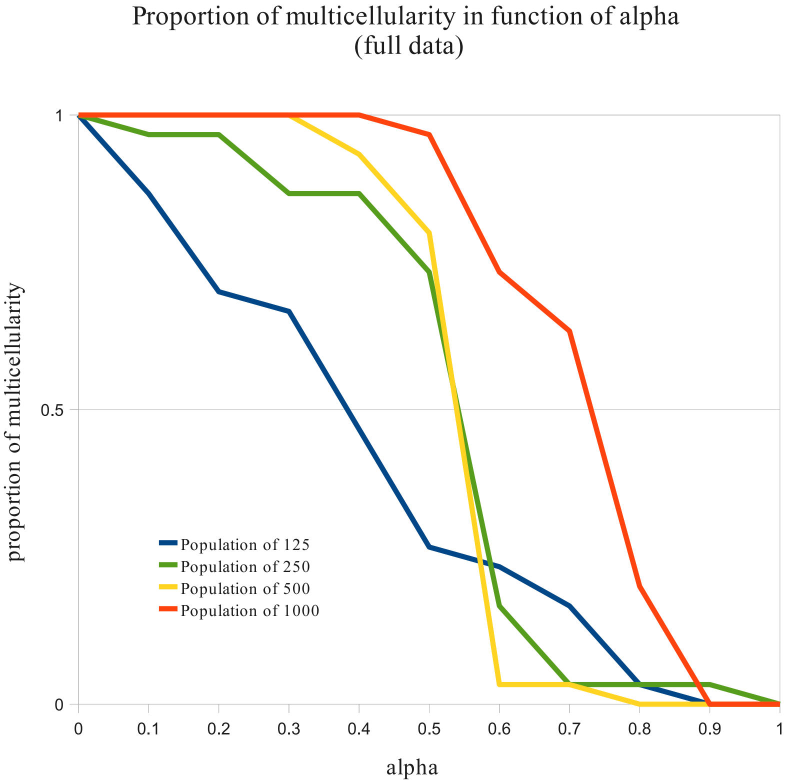
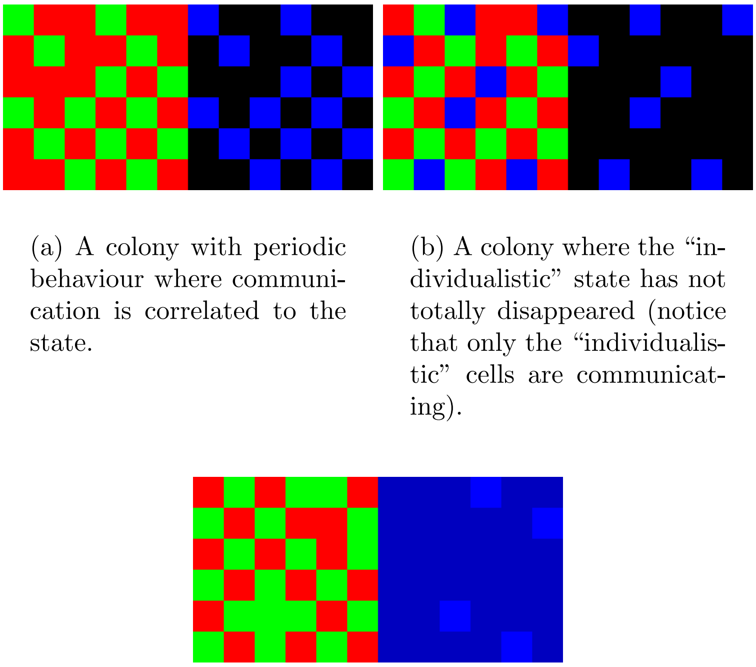


Figure 6.6: Proportion of evolutionary runs that have stabilized on the multicellular state for varying number of generations, for different values of α (full data).



(a) A colony with periodic behaviour where communication is correlated to the state.

(b) A colony where the "individualistic" state has not totally disappeared (notice that only the "individualistic" cells are communicating).

(c) A stable colony, with only transient differentiated communication.

Figure 6.7: Snapshots of some simulations. Left in each pair (a, b & c): the statuses of the cells, here the cells are either in "red" or "green" state or in blue for the "individualistic" state and the whole multicellular organism is rewarded for building a checker-board-pattern. Right in each pair (a, b & c): cells which are "communicating" with their neighbours are coloured in blue depending on the "amount" of communication.

of cooperation in that environment (or the fitness gain of being a cooperative colony). In that case one could say that evolution is not only quantified by absolute fitness, but also by the computational complexity of the way of achieving this fitness.

These kinds of results are not easily discovered through classical models of evolution. The problem the cells have to solve in this setup is similar to the iterated prisoners dilemma, which is a much-studied model in game theory Graham Kendall (2007). However, by bringing into the game the complex genotype-phenotype mapping of the GRN controlling the artificial cell some new conclusions have been possible. In most mathematical or game theoretical approaches the system will always stabilize at the stable point of highest pay-off, which in the case of this model design would have been the multicellular peak. Of course one could design a model to take into account a parameter representing computational complexity and complexity of the genotype-phenotype mapping, but as for the purpose of identification of new hypothesis normal models would not have been able to show this kind of behaviour. Also, in mathematical or game theoretical approaches, the cooperative or individualistic behaviours are fixed by the genotype, in the model I presented in this chapter, they are partially determined by the genotype but through a complex genotype-phenotype mapping, hence the cells can switch their behaviour during their lifetime. This is very important to study, and cannot easily be done with more classical models.

The fact that evolution is not only driven by fitness but also by the computational complexity is of course not new. Gould has been battling with Dawkins for a long time about the importance of what he calls "developmental constraints", the idea that a goat might be "fitter" with some eyes behind his head, but the complexity of achieve this new adaptation is far too big to ever happen. I think everybody agrees that this is the case, the main question here is how important in our evolutionary history has this been, and I think the kind of models I presented in this chapter and the next can help towards this.

Still have we come with this experiment any closer to understanding the evolution of multicellularity or the design of multicellular computational systems? We have been able to show that in this setup complexity of multicellularity has a role, but many things are still un-answered and I don't think we got any closer to developing multicellular computational system.

Explicit fitness driven systems often have one major problem, one can only get what one puts in. For example, in this system there is no conflict at the lower level, simply because even though it is a multi-layered system, there is no evolution at the cell level during simulation time, the conflict is only at the level of competing colonies. A model is defined by its assumptions and in fitness driven systems many assumptions will be taken when the fitness function is designed, the system will not evolve policing if there is no lower level conflict, this can be useful if you want to study a very specific point of a problem, so to minimize side effects, but I want to, in a first time, get a global idea of the workings of the evolution of multicellularity, for this the best (in my humble opinion) is an implicit fitness driven system as I will present in the next chapter.

Chapter 7

Filament World Experiment

> "If you try and take a cat apart to see how it works, the first thing you have on your hands is a non-working cat."
>
> Douglas Adams

After having used an explicit fitness function driven system in the previous chapter, in this chapter we will propose a framework allowing the study of evolution of multicelled interaction in an implicit (or natural) fitness driven system.

One of the major problems of looking at evolution with defined fitnesses is the huge amount of assumptions taken. The results of a study will be greatly biased by the fitness function and landscape. When the prisoner's dilemma is used in evolutionary game theory to study the evolution of cooperation, the values in the reward matrix and the operators will fully define the dynamics of the system. We make assumptions about what the players want to "optimize" and the behaviour the system will have will depend on that. The problem with making assumptions about fitness is that in a natural environment it is extremely difficult to actually know what fitness is, and what actually gets optimized.

One way to remedy this problem is not to assume a specific fitness function, but let the environment in which the entities that can evolve drive evolution: a so-called *natural*, *ecological* or *implicit* fitness driven system (I will use those terms interchangeably). There still is a great amount of assumptions to be clarified, but the assumptions in this case will not be about what the cells will “optimize”, but about the environment they evolve in, these are usually much easier to visualize and understand.

In such a system individuals will “be born”, “reproduce” (sometimes), and “die” (more or less young). A population of such individuals will interact between each other, and with the environment. This will influence how “well” they live, how often they reproduce, how young they die, or any other parameter of their lives, and, if the individuals fit to what is needed for natural selection to occur, the population will evolve. In this kind of system very little is assumed about the evolution itself (the main assumptions about evolution will be on the details of reproduction), so any conclusion about evolutionary dynamics will be independent of a specific evolutionary theory paradigm. The results will still be dependent on the assumptions made about the environment though.

One downside of these kinds of models is a certain difficulty of study. In an explicit-fitness driven system, the fitness itself is the most important quantity to study and contains usually the information one was looking for. Yet in an implicit-fitness driven system it can be very difficult to understand what is happening in the system, and to understand why some things are happening.

We will try to minimize the problem of the inherent complexity of an implicit-fitness driven system, by designing a setup that is as simple as possible at the top level (cells interact with cells in a filament with no movement allowed, they live, reproduce, and die), yet to have enough complexity of behaviour and possibilities for evolution a complex layer behind the simple one (each cell will be controlled by a GRN). This kind of setup allows to encapsulate and unpack different levels of study when it is needed, e.g. study at a population level with simple population statistics, but if interesting phenomena appear, the possibility of in depth study (genomic studies, or expression analysis for example) is still there.

To further help, we have designed a set of easily understood measures to detect the appearance of multicelled interactions and hopefully multicellularity.

7.1 The Setup

The base scenario is a reasonably simple abstraction of evolution of cell colonies: The experiment has two main phases: the evolution of a single-celled organism, and the evolution of the descendants of that evolved single cell in an environment allowing interaction between cells. The single cell is evolved to perform two tasks in parallel: one task representing metabolism, one representing a reproductive cycle. The metabolism task comprises processing some information the cell gets from the outside world in an appropriate fashion (in our case, to determine which of two 4-bit numbers coming from the environment is greater). The reproductive cycle is modelled as a simple a sequence of protein activations the cell has to perform in a certain order. We evolve a cell performing these tasks with the help of a simple Genetic Algorithm. Once a reasonably good single cell has been evolved, we use it as a seed in a one-dimensional cellular array, and we let it reproduce freely, with mutation. It still has to perform the metabolism task or get penalised, and each time a reproductive cycle is performed a new cell, with an inherited genome describing its GRN (that has possible mutated), is inserted next to the mother cell. The capacity to communicate is given to the cells of this growing cellular filament. We hypothesise that under certain conditions multicelled cooperation (and multicellularity) will be easily detectable with some very instinctive measures.

7.2 Extension of the Methods

7.2.1 Extension of the GRN Model

7.2.1.1 Metabolism Task

Our artificial cells have to perform two distinct tasks, one representing a reproductive cycle (see section 7.2.1.2) and a task representing the general processing of environmental information a cell does. This second task we will call metabolic task, we call it so because it will represent the 'life maintaining' process of the cell, not because it produces any 'biomass'. The 'maintaining task' in a biological organism is highly complex so for our purpose we will have to simplify and abstract away. We will see the metabolic task simply as a non-trivial computation the cell has to perform using some information it gets from the environment. We arbitrarily chose a comparison of two random numbers for this computation, but any number of different computations could have been considered.

In our implementation, the information the environment gives to the cell is modelled as two new 4-bit random numbers at each time step. Those 8 bits of information are transmitted to the cell directly as proteins; 8 predefined proteins of the cell are switched to the state of the binary values of the 8 environmental bits. And the task the cell has to perform is to recognise when the first value of the two 4-bit binary numbers encoded in those 8 bits is strictly larger then the second. One predetermined protein of the cell is the output protein and is checked after one update of the GRN (for example if the two numbers are 13 and 7, e.g. '`1101`' and '`0111`' in binary, at a time step i, the output protein would have to be '1' at time step $i+1$, because 13 is larger than 7). This task, done at each time step, represents the ever-ongoing adaptations and computations a cell has to perform during its lifetime (West-Eberhard, 2003). If the cell does not perform well enough at this task it will get either a bad fitness in the case of the GA evolution or will get a drastic energy penalty if in the filament environment. We chose this specific task rather than a more "biological" task mostly for sake of simplicity (no need for parametrization) and abstraction. Again, one could have chosen any kind of non-trivial information processing task, with information input from the outside world (here the two 4-bit numbers) and expected behaviour depending on it.

7.2.1.2 Reproductive Cycle

In any biological cell, life is directed by a reproductive cycle, which generally ends with mitosis, the dividing of the cell into two daughters (each of which inherits its genetic traits from the parent cell). Inspired by this, our cells will also have a simplified abstracted reproductive cycle. These kinds of cycles have been evolved successfully in GRN controlled models.

This cycle is controlled by 5 proteins[1]. The cell has to cycle through a (arbitrarily chosen) designed pattern of expression of these proteins to be able to reproduce. So to start the reproductive cycle the cell has to have those five

[1] A number selected empirically after test experiments have shown that longer cycles take too long to evolve.

Table 7.1: The Reproductive Cycle

1^{st} step	'11100'
2^{nd} step	'01110'
3^{rd} step	'00111'
4^{th} step	'10011'
5^{th} step	'11001'

proteins set to '11100', and to continue it has to cycle onwards to '01110' and finish with after the fifth step in '11001' (Table 7.1).

7.2.2 Fitness for the GA

To compute the fitness of each individual in the GA, the GRN of each individual is run for 100 time steps during which three different values are recorded. The first value recorded, f_1, is how far each cell advanced into its reproductive cycle. So if the cell achieved only step 1 followed immediately by step 2 of the cycle during a run, its fitness $f_1 = 2$; the cell has to start with step 1, else its fitness is $f_1 = 0$, f_1 becomes maximal when all five steps have been done in order. The second component of the fitness, f_2, is the accuracy of the computation of the metabolism task, if the cell got the right answer 78 times, its fitness $f_2 = 0.78$. The third fitness is the number of partial reproductive cycles of the maximum size the cell achieved, so if that cell with $f_1 = 2$ did 10 partial cycles of size 2, it would have $f_3 = 10$.

To know which of two individuals has the higher fitness we compare successively the three fitnesses; so f_1 is the most important component, if f_1 is higher for one of the individuals, that individual's global fitness is the better one. So the global fitness prioritises complete reproductive cycling over the accuracy of the metabolism and prioritises metabolism over the number of reproductive cycles.

7.2.3 The Cell Filament

For the second part of the experimental setup the evolutionary environment, we let cells reproduce and evolve freely in a one-dimensional filamentous cell array, growing from a selected single cell of the GA, which will be the initial leftmost cell. The cells and environment will be updated synchronously. At

Table 7.2: Energies

E_{max}	maximum/birth energy
E_{meta}	cost of wrong metabolism computation
E_{GRN}	running cost of the GRN
E^{n}_{pop}	population-dependent energy penalty

each time step all the cells of the filament will get the same random 8 bits of environmental information for their metabolism task.

This arrangement could potentially allow for multicelled interaction and ultimatively multicellularity to emerge.

7.2.3.1 Reproduction

Each time any cell completes two reproductive cycles a new cell is created and placed directly to the right of the mother cell in the filament. This daughter cell is a copy of the mother cell but may be mutated similarly as in the GA. No crossover is applied.

7.2.3.2 Energies

To get an abstraction of a living colony of simple cells, the cells need to die. For this purpose the fitness of the GA has been replaced by a set of energy consumptions (Table 7.2). Each cell is born with a fixed amount of energy E_{max}. Each time a cell gets the metabolism computation wrong it will lose a certain fixed amount of energy E_{meta}. Added to that each bit of change (from 0 to 1 or 1 to 0) in the proteins of a cell costs one single unit of energy, so the operating energy cost of the GRN E_{GRN} is the Hamming distance between the protein expression levels at time t and $t + 1$. At every time step the energy level of each cell gets updated, and if a cell runs out of energy it is removed from the environment.

7.2.3.3 Communication

If we want the cells to show cooperation and differentiation we presumably have to give them means of communication. For the purpose, we use a communication protocol similar to a one-step diffusion. In particular, we use the *OR*-unconstrained communication protocol studied in (Buck and Nehaniv, 2008a).

Each cell has four emitting proteins and four respective receiving proteins. If one of the cell's emitting proteins is active the respective receiving protein of its neighbouring cells gets activated.

7.2.3.4 Limiting Growth

A first series of exploratory experiments showed that growth had to be limited since the population dynamics of the system as described are exponential. Therefore the population either exponentially grows or dies out, and, as the memory of the computer systems running the simulations is not infinite, the growth of the population had to be limited, reflecting finiteness of resources as in biological evolution.

To limit the growth of the population of cells, two systems have been implemented. The first one is a hard cap on the population size, if the population increases above a certain cap, random cells are decimated until the population is below the cap again. This strategy is implemented so that the simulation does not run out of memory. The second system is inspired from population dynamics. We added to the energy calculation a population-size dependent energy penalty (Table 7.2). At every time step each cell incurs an energy penalty $E^n_{\mathrm{pop}} = n/\kappa$, where n is the size of the population at that time step, and κ, an empirically set parameter which determines a certain maximum population size dependent on the actual implicit fitness of the population. This, in effect, is a model of logistic growth in population dynamics (Roughgarden, 1979). The parameter κ is set so that this maximum population size is reasonably stable (no risk of extinction) but leaving enough space for the population to become more efficient without reaching the hard capped maximum size.

7.3 Measures and Results

7.3.1 The Single Cell

To evolve the single cell we use the GA as described earlier with a population size of 1000, random initial genomes 10000 bits long and a mutation rate of 0.0001. The evolutionary process achieves reasonably good individuals: $f_1 = 5$, exhibiting the 5-step full reproductive cycle; $f_2 \simeq 0.7$, so that the metabolism task is performed correctly most of the time; f_3 varying between 1 and 24 (25 being the possible maximum for $f_1 = 5$). Evolution reliably yields genomes with cells able to complete the full reproductive cycle (at least once) and completing the metabolic task reasonably well. These evolved individuals serve the purpose of providing seed cells to study the emergence of cooperation and differentiation in the multicellular filamentous setting.

7.3.2 The Filament World

7.3.2.1 Experimental Setup

In this step of the experiment we inject a single pre-evolved seed cell (Section 7.3.1 into the filament world, and let it evolve freely in this context as described earlier. Not all the genomes are adequate for this evolution: when we add communication the GRNs of some pre-evolved cells are disrupted and are not able to reproduce any more, and therefore the colonies are not viable. So we select viable single cells, and study free evolution in the resulting filamentous colonies.

We will concentrate in this analysis only on the most interesting run discovered, similar runs where available with less strong effects. The results we present here are set in an environment with: $E_{\mathrm{max}} = 5000$, $E_{\mathrm{meta}} = 500$, $\kappa = 15$, and a mutation rate of 0.0001 (no cross-over). We study two different filamentous colonies, one with communication (experimental condition) and one without communication (control experiment), starting with the same initial cell. The experiments ended when all the cells of the filament were dead (extinction) or ran until a maximal time limit is reached (in this case around 5.10^5 time steps).

7.3.2.2 Effects of a Multicelled Environment

Our model can be analysed with many different approaches. One could study genome evolution and phylogenies, expression patterns of the GRN, or population dynamics for example. Many methods used in biology can be applied with little modification, and the added benefit being that the data used would be complete. For this first study instead of such detailed analysis that would only be justified for particularly interesting evolutionary runs, we will use and present here only population statistics using our new measures. With this kind of statistics one can recover information about the evolution of a filament rapidly, and evidence of multicelled interaction can be easily spotted.

7.3.2.3 Measures

To study the evolution of this system a number of possible population statistics could be considered: number of cells over time, life expectancy, average number of offspring, efficiency of the metabolic computation, etc. In this study we look more precisely at three measures: life expectancy, average number of offspring, and proportion of reproducers, where 4500-time-steps windows are used for these rolling averages and the data for all the cells which died in each period are used. For the two first of those measures (life expectancy and number of offspring) we will also compute derivate measures representing the effect of the colony on the individual. To compute those measures, each time a cell dies we rerun it in isolation (i.e. without any communication) from its neighbours (we still remove the energy lost from the population dependent penalty term though) and then measure the dead cell's life expectancy and number of offspring. We will call these measures *individual potential life expectancy* and *individual potential number of offspring*. These potentials can be smaller or greater than the actual measure in the multicellular environment. If they are smaller they would mean that the cells live longer and/or have more offspring if they get some information from neighbouring cells the can potentially interact with and conversely. We also could have used already existing measures for artificial life systems such as the ones presented in Mark et al. (1992), but the adaptation of these onto a complex model like ours needs a great number of simplification which could make the data difficult to analyse.

The life expectancy measure gives us an idea of how well the metabolic task is performed and how efficiently the GRN is used (the E_{GRN} term), if we compare it to the individual potential life expectancy we could detect any kind of cooperative computation and metabolism. The average number of offspring should be varying around one due to the population limiting term, but by comparing to the individual potential number of offspring one can detect the presence of population and growth control organised at a population or local level. The last statistic studied is the proportion of cells having one or more offspring during their life time; this should help us to detect whether any kind

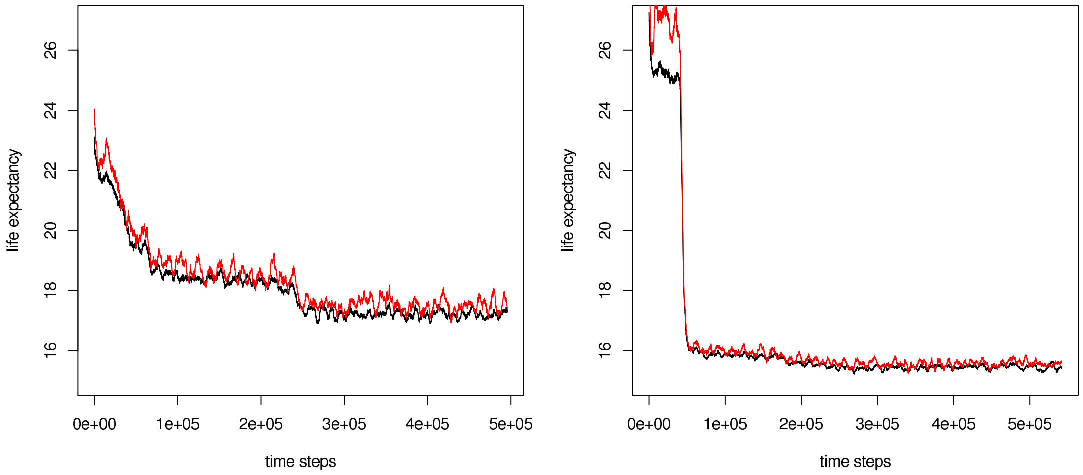


Figure 7.1: Evolution of average cell lifetime (measured over rolling 4500 time-step windows): Multicellular filament with (left) and without inter-cellular communicative ability (right). The red lines being the individual potential life expectancy as described in section 7.3.2.2

of germ/soma differentiation could be happening[2].

We apply these statistics for both kinds of colonies: communicating and non-communicating.

7.3.2.4 Results and Analysis

At first sight there is no evolution of multicellularity, there is no major shift after stabilization in Figure 7.3. The use of information coming from neighbouring cells is not trivial so a colony of cells might well discard or reduce as much as possible the effect of communication so as not to disrupt the operation of its individual cells. But examining the individual potentials we have presented in section 7.3.2 can refine this analysis. We can notice that indeed there is no effect of communication on life expectancy (figure 7.1), as both curves are very close, but there are some spikes for the individual potential number of offspring. For the control setting individual potential number of offspring never reaches below the actual number of offspring, yet in the colony

[2] If this measure lowers significantly it would mean that a certain proportion of the population is not reproducing, and possibly doing the metabolic task for the reproducing (germ) cells.

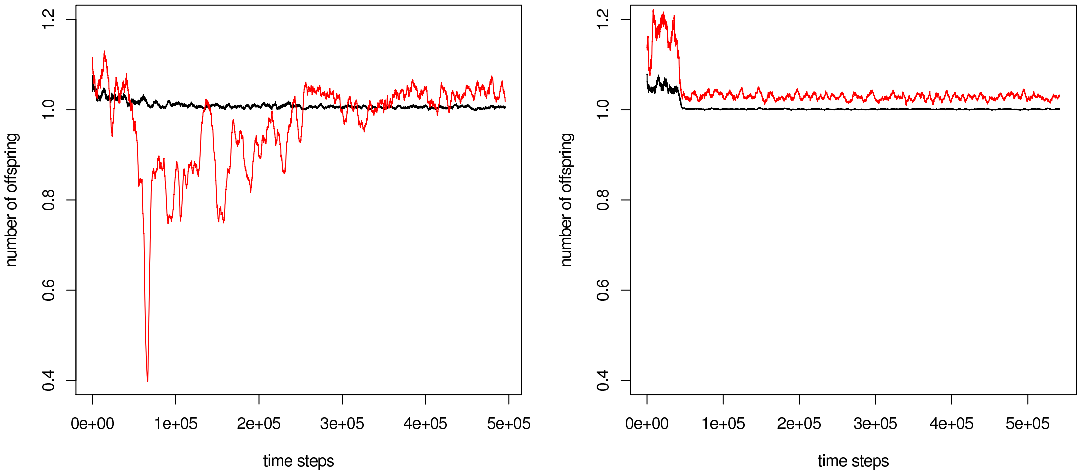


Figure 7.2: Evolution of average number of offspring per cell (measured over rolling 4500 time-step windows): Multicellular filament with (left) and without inter cellular communicative ability (right). The red lines being the individual potential number of offspring as described in section 7.3.2.2

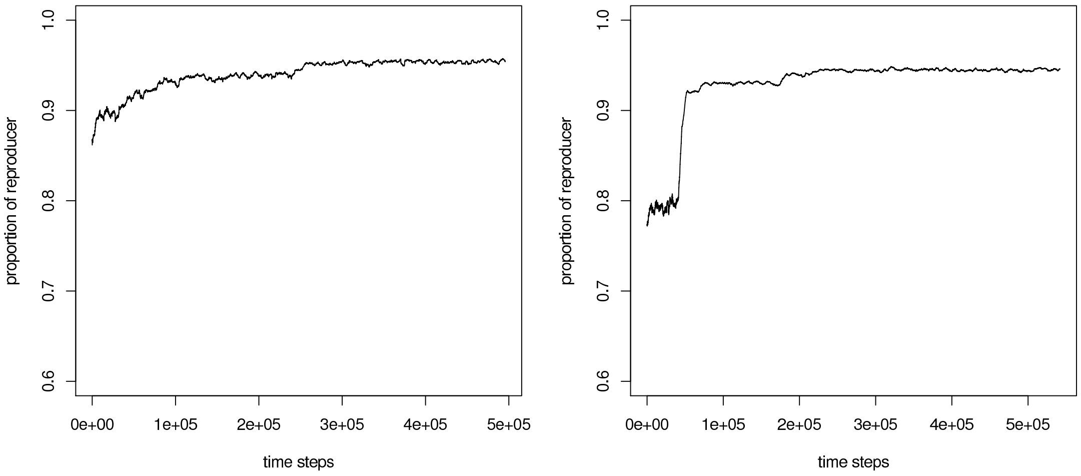


Figure 7.3: Evolution of proportion of cells reproducing (measured over rolling 4500 time-step windows): Multicellular filament with (left) and without inter cellular communicative ability (right).

of cells able to communicate large spikes drop as low as 0.4, which hints to some sort of indirect multicelled behaviour (see Section 2.4.1). With the sets of statistics available we are able to comment on those spikes. First it is not related to a drop in individual potential life expectancy (life expectancy and number of offspring are related, as if the cell lives longer it has a higher probability to reproduce more), as we can see from the graphs in figure 7.1. A second comment is that they are not related to some kind of differentiation as the proportion of reproducers in figure 7.3 is very close to one (meaning that almost all cells in the filament reproduce). This is consistent with a hypothesis that the cells are 'experimenting' with some kind of birth regulation. Without analysis in depths of the GRNs in the cells of the filament during those spikes, we can not know exactly what kind of control this might be (but the model is very adequate for this kind of in depth analysis) but one can hypothesise one of three main processes: (1) a "green beard" kind of control (Dawkins, 1976), meaning that the cells use communication to recognise each other and don't cooperate with cells which do not signal appropriately and hence block the reproduction of such cells, or (2) a population control, where the cells try to control the growth of the population so as not to overuse the environment, or (3) cells prevent badly mutated cells ("cancer" cells) from reproducing. We can also notice that the spikes are not unique, more than one of diverse amplitudes occur. This also could hint at a typical *cheater* appearance: some sort of cooperative behaviour appears and gets invaded by a phenotypes disrupting and abusing the cooperators until no cooperator is left. This type of evolutionary dynamics can cycle, as another (or eventually the same) cooperative behaviour invades again, and so on.

7.4 Conclusion

In this experiment we have first evolved a viable cell able to perform a reproductive cycle and an abstraction of house-keeping metabolism, and then let this cell seed a simple multicelled environment where under free evolution with the potential for inter-cellular communication, we have seen that without an ability to communicate the cells are stuck in an evolutionary stasis, whereas with the ability to communicate the colony of cells appear to "experiment" with birth regulation behaviours and/or diverse communicative behaviours.

But the main contribution of this study is an intuitive artificial life framework for the study of evolution of multicelled behaviour and (possibly) multicellularity, complete with a set of generally applicable measures allowing easy identification of cooperative behaviour, and more in-depth analysis. The measures defined in section 7.3.2.2, allow to detect automatically if a colony of cells evolved some sort of multicelled cooperation, and reduces greatly the need for anthropocentric observations of behaviour (as is often the case in artificial life models). It is also possible with such system to do any kind of phylogenetic studies (phylogenetic trees, Manhattan plots...), network analyses, contextual analyses, application of Price's equations, as one would like to for a biological system.

One can get data from this model that are not available (or at least not without painstaking efforts) in biology, data that can be fitted to mathematical models of varying complexity. This is one issues of mathematical models, there simply is not enough data available to confirm any particular model of a phenomena. Models like the one presented here can help fill this gap. However one has to be careful, certain assumptions have been taken in this model as well which can differentiate it qualitatively from biology

We also hope that with further studies of parameters and better evolved single cells we might be able to observe complete differentiation of reproductive (cells only performing the reproductive cycle) and soma (cells only doing the metabolism task) cells. This framework can be extended to many more complex population dynamics if we introduce perturbation of the environment, cell movement and migration, breaking up and fusing of different filaments, diverse metabolic tasks, sexual behaviour (exchange of genetic material), etc. Finally, extensions of this kind of artificial life systems may help us to gain some insight into the grey areas of major evolutionary transitions (Buss, 1987; Maynard Smith and Szathmáry, 1995; Okasha, 2006), and perhaps move beyond and complement the traditional paradigms for the study of evolution.

Chapter 8

Summary and Conclusion

> "[A] curious aspect of the theory of evolution is that everybody thinks he understands it."
>
> Jacques Monod (1975)

Computers have been getting faster and faster over the last decades, but not only that, they have also become more complex, more networked and more ubiquitous. This leads to one of the major problems of computer sciences nowadays: how can we use and program these complex distributed networks of processing power? This problem is not only in the realm of computer sciences, new technologies like nanomachine, and bioengineering struggle with similar issues.

There is one process that we know of that has been designing "programs" for this kind of systems for millions of years: evolution. Multicellular organisms populate the earth; cooperation and division of labour are present at every level of the evolutionary tree. And what are multicellular organisms but extremely complex highly distributed systems? If one could harness the power of evolution to build artificial "multicellular" entities, one would have made a great leap toward using all the processing power available to us nowadays.

The problem is that the evolution of multicellularity is not very well understood, mostly from the aspect of dynamics. During the course of this thesis research, I have tried to address certain issues about the evolutionary theory surrounding the evolution of multicellularity and its application to computer sciences. How can we evolve computational multicelled or multicellular systems? What are the necessary conditions? What happens during a transition in evolution?

8.1 Context and Methodology

In Chapter 2, I have presented a survey of the actual state of evolutionary theory. Noticeably the theories and frameworks for the study of evolution of complex adaptations, such as those qualified as major transitions in evolution like the advent of multicellularity. I then argue how computer sciences have a two-fold link with certain of these issues: an utilitarian need, and as a modelling tool; explicitly, computer sciences could be greatly helped if evolution could be harnessed as a development tool for highly complex distributed computational systems, and also, conversely computer science models could provide modelling tools that would greatly improve upon the actual (mostly mathematical) modelling paradigms used in evolutionary theory.

I followed then with a presentation (Chapter 3) of how computer science has addressed these engineering and modelling aspects so far. I presented algorithms that have been inspired by biology and evolution, these are used in engineering as well as for the modeling of biological processes. Some of these tools have also been used (as optimization tools, as well as modelling tools) in my practical work (Sections 4 to 7). In Section 3.4, I present some models that study evolutionary theory with methodologies closer to informatics than biology.

It is important, if we want to transfer knowledge on questions of evolution, that the biology and computer science communities share a common language. If computational models are too abstract, it can be very difficult for a biologist to exchange and use the information gained, hence it is important to think about designing models that can be compared and presented to biology and its research community. On the other hand many models are limited in the levels of complex behaviour they can evolve. The levels of complex behaviours models can evolve are important, both for the computer scientist, and the biologist. The computer scientist wants to evolve complex systems as an engineer, and the biologist wants to understand the evolution of complex behaviour and adaptations. Yet the search for complexity often has a drawback: the complexity of analysis. So there has to be a trade-off between the complexity of possible behaviours, and the ease of analysis.

8.2 Experiments on Multicelled Systems

In the next few chapters I presented a set of models and methodologies to address these points. Chapter 4 presented the main tools I use to build my experiments: an artificial cell model (Section 4.1), a genetic regulatory network model (Section 4.1.1), and evolutionary algorithms (Section 4.2). These tools address one of the previously mentioned points: cross-disciplinarity. Even though they are algorithmic tools, they are reasonably easily understood by biologists.

In the first experiment (Chapter 5), I have presented a model that uses more or less standard computational paradigms to create a multicelled system to colour graphs, as an experimental scenario for the evolution of multicelled cooperation. I use this experiment to show some of the classical problems of standard optimization methods (GAs, for example): the problems of scalability, fitness, and adaptability. Standard optimization metaheuristics can usually handle very small multicellular entities in a not too dynamical environment, but will fail if the number of cells is too high or the environment is changing permanently too much; this is what I call the scalability and adaptability problems. The issue with fitness is that one wants massively parallel distributed systems to be very versatile, but designing a fitness function to emulate the needed versatility is unrealistic. I use this experiment also as a test bed for certain engineering choices for the follow-up models (mostly about the implementation of communication). I argue as conclusion that to improve and evolve complex distributed systems, it would be very helpful to understand in which conditions, and how, multicellularity can evolve, and how this can be controlled, how it interacts with multicellular development, growth and policing.

The second experiment, presented in Chapter 6, is an endeavour to approach the question of the necessary conditions for evolution of multicelled cooperation, as well as the relationship of cooperating and non-cooperating cells. I have also used this experiment to study the topology of the genotype-phenotype mapping of my artificial cell; verifying whether the mapping is complex enough for interesting evolutionary behaviours without it being too much. I have evolved clonal colonies of artificial cells where cells can behave in a cooperative or non-cooperative manner, both behaviours contributing to different competing fitnesses. I have here shown that the fixation of one or the other behaviour is not especially dependent on population size (number of colonies involved with the genetic algorithm), but more on environmental variables, and the difficulty (computationally speaking) of the cooperative behaviour. This could not have been shown by standard population dynamics systems.

The last experiment (Chapter 7) is a setup that ties the lessons of the two first experiments together with the conclusions of Chapters 2 and 3. Artificial cells evolve freely in a setting where multicellularity can evolve. It has no explicit fitness function[1], no predefined fitness and levels of selection, and only natural selection acts. It is understandable by biologists, and tools developed for biology can be used to analyse it. Some novel measures have also been developed that are specific to artificial life models. These are population dynamics measures (life expectancy, and average number of offspring), but applied to cells that I can make "live again" without their cellular environment, hence without communication (which is quite difficult in biology). These have allowed to show that some form of cooperative multicelled control of reproduction evolved, which had not been foreseen. This model has allowed us to show that it is possible to design models that have the three characteristics identified in our methodological goals: they evolve complex behaviours, are easy to understand (by a wide variety of scientists), and are easily analysable (with a wide variety of tools and at different levels).

8.3 Contribution to Research Questions

In this work I have endeavoured to close in on the first research question of a strategy for finding new ways to design massively parallel computational systems, and this starting question led me to the evolution of multicellularity. I concluded from my literature reviews (Chapters 1 to 3) and my first experiment (Chapter 5) that genetic algorithms of a standard type are not especially adequate to evolve multicellular systems, hence emerged the idea to evolve multicellularity itself as it happened in biology. To develop computational multicellular system we need to understand multicellularity itself. The design of massively parallel computational systems and the understanding of the evolution of multicellularity requires a "cell" with sufficiently rich

[1] Beyond these used to evolve the initial cells in isolation, before the actual experiment begins.

dynamical and evolutionary potential. Discrete genetic regulatory networks (GRNs) as a basis for studying the evolution of cooperative differential multicellularity are motivated and introduced, and a cell model containing desired characteristics is presented (Chapter 4).

These thoughts led to the second research question I have addressed with this work: understanding the details and mechanisms of the evolution of multicellularity; or more specifically: *how to evolve multicellularity in an artificial systems*? Chapters 6 and 7 present two such starting points: one using a fitness driven model that has two explicit levels of organisation, each driving different measures of fitness; and one where artificial cells evolve freely without the constraint of a fitness function, they just interact with their environment and neighbouring cells. These models show novel ways how questions about the evolution of multicellularity do not have to be studied the more traditional way, and that Artificial Life style models can help greatly the understanding of these questions and artificial life.

With this research I have first shown the importance of new methodologies for the study of evolution, and more particularly the evolution of multicellularity. This is important both for computer science to develop new computational tools, and for biology. The new methodologies requires models that are be cross-disciplinary, have the potential of highly complex behaviour, and easy to analyse . I also have presented two models that share those characteristics. These have shown behaviours and results (understandable both by computer scientists and biologists) that would not have been predictable by the standard methodologies used in the field of evolutionary theory.

8.4 Remaining Issues and Future Work

The research done during this thesis has shown that it is important and possible to design new types of models for the evolution of multicellularity: models that show a large possibility of complex behaviour, and that are understandable by biologists and computer scientists alike. But there still is a long road ahead.

Even though the filament-like setup (Chapter 7) did show some evidence suggesting cooperation, they are far from conclusive. This is one of the major problems of this sort of models: because the goal is not “hard-coded” into the system (with a fitness function, or otherwise), one can never be sure when, which, or even if one will get an interesting result. Nevertheless, one can use measures like those developed in chapter 7 to detect relevant activity. One can put in all the “ingredients” for the evolution of some interesting behaviour, but might have forgotten the “pinch of salt”, and find nothing. Hence this kind of models does need to be played with extensively, for interesting results to happen. But once interesting results have been found (even better if repeatable), in depth analysis, unimaginable to biologists can be done.

To dodge this issue, one could use models like the one in Chapter 6, where the “evolution of multicellularity” is very scaffolded with two fitness functions and the GA. These kinds of models are closer to the more traditional mathematical models of evolution. It is very easy to see the effect of certain variables on the outcome, but such models, with the rich internal structure of cells and interaction between them, can show more complex behaviours than mathematical models. Yet the information you get from them is difficult to transfer due to the nature of the model, it would be difficult to fit real biological data to them like it would for a mathematical model, or even to compare it (in a qualitative way) to biological systems, as is often done with other Artificial Life systems. Nevertheless, they can be a fast and easy way to try out some hypothesis, or parameters, before building more elaborate models (like the filament setup).

One difficult issue that this thesis did not resolve, is the complete closing of the circle, how to apply the gained knowledge to computer sciences to build efficient and effective massively parallel computational systems. One of the motivations of this thesis is to understand evolution of cooperation and multicellularity better so that we can apply these understandings to computer sciences and the design (or evolution) of massively parallel computational systems; this part is deep and (sadly) has not been achieved by anyone, yet. That does not mean, though, that this return loop will never be possible, the lessons one can get from this kind of models will be exceedingly helpful for computer sciences, evolutionary theory, biology, and medicine (especially oncology).

However the road to gain this kind of insights is still long and twisty, there is still too much to discover on how multicellularity evolves. One of the first results that would have to be achieved next would be to have a model that shows full division of labour, for example, using a model similar to the one presented in Chapter 7 where the cells would evolve that either perform reproduction or the computational task. These simulations can then be studied in depth with all possible tools so that the driving forces of this evolution can be well understood. The next steps would be to do similar studies but changing some of the starting assumptions: adding sexual reproduction, mobility, lateral gene transfer, genes with different functions, inheritable protein levels, etc; and then see how these different assumptions influence the way multicellularity evolves. These results should allow us to get a much better understanding of the driving forces behind the evolution of multicellularity (and maybe even other major transitions). This better understanding would be of immense help for the design of computational multicellular systems, and the main biological applications: the understanding of cancer and of development.

But the applications do not stop here, similar approaches to those in this thesis can also be used for different questions in evolutionary theory, such as the evolution of the first replicators or the evolution of sex. Most of the major transitions in evolution are still very badly understood, and the more mainstream modelling methods tend to be inadequate to study them, however new Artificial Life methodologies, like the ones presented in this thesis will be needed and will eventually lead to some of the answers to these questions.

Appendix A

List of Published Papers

A.1 Effect of Multi-Level Fitnesses on the Development of Multicellular Artificial Organisms

"Effect of Multi-Level Fitnesses on the Development of Multicellular Artificial Organisms" (Buck and Nehaniv, 2006b) was presented at the 7^{th} German Workshop in Artificial Life (GWAL-7), July 2006, in Jena, Germany. This paper presents an early version of the artificial cell presented in chapter 4, which used a continuous GRN, and used for an experiment very similar to the one presented in 6.

A.2 Discrete Developmental Genetic Regulatory Networks for the Evolution of Cooperation

"Discrete Developmental Genetic Regulatory Networks for the Evolution of Cooperation" (Buck and Nehaniv, 2006a) was presented at the AAAI Fall Symposium (October 13-15, 2006, Arlington, Virginia) on the Developmental Systems track. This paper presents the first version of the discrete GRN used for the artificial cell across this thesis. It is used in a preliminary version of the experiment of chapter 6.

A.3 Colouring graphs using a GRN/cell-based system

"Colouring graphs using a GRN/cell-based system" (Buck and Nehaniv, 2007) was presented at 7^{th} International Workshop on Information Processing in Cell and Tissues (IPCAT), August 2007, in Oxford, UK. The work presented here was some preliminary work to chapter 5.

A.4 Communication and complexity in a GRN-based multicellular system for graph colouring

"Communication and complexity in a GRN-based multicellular system for graph colouring" (Buck and Nehaniv, 2008a) is an extended version of (Buck and Nehaniv, 2007) it has been published in *BioSystems* in 2008. The results presented in chapter 5 are based on the results presented in this journal paper.

A.5 Looking for Evidence of Differentiation and Multicellular Cooperation

"Looking for Evidence of Differentiation and Multicellular Cooperation" (Buck and Nehaniv, 2008b) was presented at the 8^{th} German Workshop in Artificial Life (GWAL-8), July-August 2008, in Leipzig, Germany. The work presented here was some preliminary work to chapter 7.

A.6 Looking for Evidence of Differentiation and Cooperation: Natural Measures for the Study of Evolution of Multicellularity

"Looking for Evidence of Differentiation and Cooperation: Natural Measures for the Study of Evolution of Multicellularity" is an extended version of (Buck and Nehaniv, 2009) it has been published in *Advances in Complex Systems* in 2009. The results presented in chapter 7 are based on the results presented in this journal paper.

Bibliography

Adami, C. and Brown, C. T. (1994). Evolutionary learning in the 2d artificial life system “avida”.

Axelrod, R. (1985). *The Evolution Of Cooperation.* Basic Books.

Banzhaf, W. (2004). On evolutionary design, embodiment, and artificial regulatory networks. In *Embodied Artificial Intelligence (LNAI 3139)*, pages 284–292. Springer Verlag, Heidelberg.

Biggs, N. (1990). Some heuristics for graph colouring. In Nelson, R. and Wilson, R. J., editors, *Graph Colourings*, pages 87–96. longman.

Bongard, J. (2002). Evolving modular genetic regulatory networks.

Bornholdt, S. (2008). Boolean network models of cellular regulation: prospects and limitations. *Journal of The Royal Society Interface*, 5(Suppl 1):S85–S94.

Buck, M. and Nehaniv, C. L. (2006a). Discrete developmental genetic regulatory networks for the evolution of cooperation. In Kumar, S., Hornby, G. S., and Bongard, J., editors, *Developmental Systems: Papers from the AAAI Fall Symposium (October 13-15, 2006, Arlington, Virginia)*, pages 9–15. Association for the Advancement of Artificial Intelligence.

Buck, M. and Nehaniv, C. L. (2006b). Effect of multi-level fitnesses on the development of multicellular artificial organisms. In *Proceedings of the* 7^{th} *German Workshop in Artificial Life (GWAL-7)*, Berlin. Akademische Verlagsgesellschaft Aka.

Buck, M. and Nehaniv, C. L. (2007). Colouring graphs using a grn/cell-based system. In *Proceedings of IPCAT 2007.*

Buck, M. and Nehaniv, C. L. (2008a). Communication and complexity in a GRN-based multicellular system for graph colouring. *Biosystems*, 94(1-2):28–33.

Buck, M. and Nehaniv, C. L. (2008b). Looking for evidence of differentiation and multicellular cooperation. In *Looking for Evidence of Differentiation and Multicellular Cooperation — - Leipzig, Germany, 30 July-1 August.*

Buck, M. and Nehaniv, C. L. (2009). Looking for evidence of differentiation and cooperation: Natural measures for the study of evolution of multicellularity. *Advances in Complex Systems.*

Bull, L. and Alonso-Sanz, R. (2008). On coupling random boolean networks. In et al., A., editor, *Automata 2008: Theory and Applications of Cellular Automata*, pages 292–301. Luniver Press.

Buss, L. W. (1987). *The Evolution of Individuality.* Princeton University Press.

Costa, D. and hertz, A. (1997). Ants can colour graphs. *Journal of the Operational Research Society*, 48:295–305.

Crick, F. H. (1958). On protein synthesis. *Symposia of the Society for Experimental Biology*, 12:138–163.

Davidson, E. H. (2001a). *Genomic Regulatory Networks: Development and Evolution.* Academic Press.

Davidson, E. H. (2001b). *Genomic Regulatory Systems: Development and Evolution.* Academic Press.

Dawkins, R. (1976). *The Selfish Gene.* Oxford University Press, USA, 3 edition.

de Jong, H. (2002). Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology*, 9(1):67–103.

Dobzhansky, T. (1973). Nothing in biology makes sense except in the light of evolution. *The American Biology Teacher*, 35(3):125–129.

Eggenberger-Hotz, P. (1997). Evolving morphologies of simulated 3D organisms based on differential gene expression. In *Proceedings of the Fourth European Conference on Artificial Life*, pages 205–213, Cambridge, MA. MIT Press.

Egri-Nagy, A. and Nehaniv, C. L. (2003). Evolvability of the genotype-phenotype relation in populations of self-replicating digital organisms in a tierra-like system. In *Advances in Artificial Life*, volume 2801 of *Lecture Notes in Computer Science*, pages 238–247. Springer Berlin / Heidelberg.

Foster, K. R., Parkinson, K., and Thompson, C. R. (2007). What can microbial genetics teach sociobiology? *Trends in Genetics*, 23(2):74 – 80.

Foster, K. R., Shaulsky, G., Strassmann, J. E., Queller, D. C., and Thompson, C. R. L. (2004). Pleiotropy as a mechanism to stabilize cooperation. *Nature*, 431(7009):693–696.

Goodnight, C. J. (2005). Multilevel selection: the evolution of cooperation in non-kin groups. *Population Ecology*, 47(1):3–12.

Graham Kendall, Xin Yao, S. Y. C. (2007). *The Iterated Prisoner's Dilemma: 20 Years On.* World Scientific.

Hamilton, W. (1964a). The genetical evolution of social behaviour. i. *Journal of Theoretical Biology*, 7(1):1–16.

Hamilton, W. D. (1964b). The genetical evolution of social behaviour. ii. *Journal of theoretical biology*, 7(1):17–52.

Hammond, B. J. (1993). Quantitative study of the control of hiv-1 gene expression. *Journal of Theoretical Biology*, 163(2):199 – 221.

Hawkins, J. D. (1996). *Gene Structure and Expression.* Cambridge University Press, 3rd edition.

Hoffmann, R. (2000). Twenty years on: The evolution of cooperation revisited. *Journal of Artificial Societies and Social Simulation*, 3(2).

Hogeweg, P. (2000). Evolving Mechanisms of Morphogenesis: on the Interplay between Differential Adhesion and Cell Differentiation. *Journal of Theoretical Biology*, 203(4):317–333.

Hornby, G. S. and Pollack, J. B. (2001). Evolving l-systems to generate virtual creatures. *Computers & Graphics*, 25(6):1041–1048.

Iguchi, K., Kinoshita, S., and Yamada, H. (2005). Rugged fitness landscapes of kauffman models with a scale-free network. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 72(6).

Jacob, F. and Monod, J. (1961). Genetic regulatory mechanisms in the synthesis of proteins†. *Journal of Molecular Biology*, 3(3):318–356.

Jansen, T. and Wegener, I. (2005). Real royal road functions–where crossover provably is essential. *Discrete Applied Mathematics*, 149(1-3):111 – 125. Boolean and Pseudo-Boolean Functions.

Karp, R. M. (1972). Reducibility among combinatorial problems. In Miller, R. E. and Thatcher, J. W., editors, *Complexity of Computer Computations*, pages 85–103. New York, Plenum.

Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22:437?467.

Kauffman, S. A. (1993). *The origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press.

Kauffman, S. A. and Smith, R. G. (1986). Adaptive automata based on darwinian selection. *Physica D: Nonlinear Phenomena*, 22(1-3):68 – 82. Proceedings of the Fifth Annual International Conference.

Kicinger, R. (2006). Evolutionary developmental system for structural design. In for Artificial Intelligence, T. A. A., editor, *Developmental Systems. Papers from the AAAI Fall Symposium. Technical Report FS-06-03*, Menlo Park, CA, 1-8.

Knabe, J. F., Nehaniv, C. L., Schilstra, M. J., and Quick, T. (2006). Evolving biological clocks using genetic regulatory networks. In Rocha, L. M., Yaeger, L. S., Bedau, M. A., Floreano, D., Goldstone, R. L., and Vespignani, A., editors, *Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 15–21. MIT Press.

Kropotkin, P. A. (1904). *Mutual aid, a factor of evolution.* Heinemann, London :, popular ed. edition.

Kumar, S. (2005). A developmental genetics-inspired approach to robot control. In *Proceedings of the Second Workshop On Self-Organization in Representations For Evolutionary Algorithms: Building complexity from simplicity, GECCO-2005*. ACM Press.

Kursawe, F. (1993). Evolution Strategies – Simple 'Models' of Natural Processes? *Journal Internationale de Systémique*, 7, N^o 5:627–642.

Langton, C. G. (1995). *Artificial Life: An Overview.* MIT Press, Cambridge, MA, USA.

Lewontin, R. C. (1961). Evolution and the theory of games. *Journal of Theoretical Biology*, 1(3):382–403.

Lewontin, R. C. (1970). The units of selection. *Annual Review of Ecology and Systematics*, 1(1):1–18.

Lindenmayer, A. (1968a). Mathematical models for cellular interactions in development i. filaments with one-sided inputs. *Journal of Theoretical Biology*, 18(3):280–299.

Lindenmayer, A. (1968b). Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. *Journal of Theoretical Biology*, 18(3):300–315.

Lipson, H. and Pollack, J. B. (2000). Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978.

Liu, T., Liu, X., Chen, Y., and Wu, R. (2007). A computational model for functional mapping of genes that regulate intracellular circadian rhythms. *Theoretical Biology and Medical Modelling*, 4(1).

Marée, S. (2000). *From Pattern Formation to Morphogenesis.* PhD thesis, University of Utrecht.

Margulis, L. (1981). *Symbiosis in Cell Evolution.* W.H.Freeman & Co Ltd.

Mark, B., Norman, A., and Packard, H. (1992). Measurement of evolutionary activity, teleology, and life.

Marx, D. (2004). Graph colouring problems and their application in scheduling. *Periodica Polytechnica Ser. El. Eng.*, 48(1):11–16.

Maynard Smith, J. and Szathmáry, E. (1995). *The Major Transitions in Evolution.* Oxford University Press.

McAdams, H. H. and Arkin, A. (1998). Simulation of prokaryotic genetic circuits. *Annu Rev Biophys Biomol Struct*, 27:199–224.

Michod, R. E. (1999). *Darwinian Dynamics: Evolutionary Transitions in Fitness and Individuality.* Princeton University Press.

Michod, R. E. (2003). Cooperation and conflict in the evolution of complexity. In *Computational Synthesis: From Basic Building Blocks to High Level Functionality*, AAAI Spring Symposium.

Mjolsness, E., Sharp, D. H., and Reinitz, J. (1991). A connectionist model of development. *Journal of Theoretical Biology*, 152(4):429 – 453.

Mode, C. J. (1958). A mathematical model for the co-evolution of obligate parasites and their hosts. *Evolution*, 12(2):158–165.

Monod, J. (1975). On the molecular theory of evolution. In *Problems of scientific revolution*, Herbert Spencer Lecture 1973, pages 11–24.

Mueller, F. (1993). Register allocation by graph coloring: A review.

Nash, J. F. (1950). Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(1):48–49.

Ohno, S. (1970). *Evolution by Gene Duplication.* Springer Verlag, New York.

Okasha, S. (2006). *Evolution and the levels of selection / Samir Okasha.* Clarendon Press, Oxford :.

Prestwich, S. (1998). Using an incomplete version of dynamic backtracking for graph colouring. In Wallace, M., Caseau, Y., Jacquet-Lagreze, E., Simonis, H., and Pesant, G., editors, *CP98 Workshop on Large Scale Combinatorial Optimisation and Constraints*, volume 1 of *Electronic Notes in Discrete Mathematics.* ELSEVIER. http://www.elsevier.nl:80/cas/tree/store/disc/free/endm/menu.sht.

Price, G. (1972). Fisher's "fundamental theorem" made clear. *Annals of Human Genetics*, 36:129â140.

Quick, T., Nehaniv, C. L., Dautenhahn, K., and Roberts, G. (2003). Evolving embodied genetic regulatory network-driven control systems. In *Advances in Artificial Life (Proc. European Conference on Artifical Life - ECAL'03)*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 266–277. Springer Verlag.

Ray, T. S. (1991). Evolution and optimization of digital organisms. In Billingsley, K. R. et al., editors, *Scientific Excellence in Supercomputing: The IBM 1990 Contest Prize Papers*, pages 489–531. The Baldwin Press, The University of Georgia.

Ray, T. S. (2000). Evolution of complexity: Tissue differentiation in network tierra. *ATR Journal*, 40(8):12–13. in Japanese, but translated on the web.

Reil, T. (1999). Dynamics of gene expression in an artificial genome - implications for biological and artificial ontogeny. In *Advances in Artificial Life (Proc. European Conference on Artifical Life - ECAL'99)*, volume 1674 of *Lecture Notes in Artificial Intelligence*, pages 457–466. Springer Verlag.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. In Stone, M. C., editor, *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, volume 21, pages 25–34, New York, NY. ACM.

Roughgarden, J. (1979). *Theory of Population Genetics and Evolutionary Ecology: An Introduction.* Macmillan Publishing Company.

Rozenberg, G. and Salomaa, A. (1992). *Lindenmayer Systems: Impacts on Theoretical Computer Science, Computer Graphics, and Developmental Biology.* Springer, 1 edition.

Schilstra, M. J. and Nehaniv, C. L. (2008). Bio-logic: gene expression and the laws of combinatorial logic. *Artificial life*, 14(1):121–133.

Schwefel, H.-P. and Kursawe, F. (1998). On Natural Life's Tricks to Survive and Evolve. In Fogel, D. B., Schwefel, H.-P., Bäck, T., and Yao, X., editors, *Proc. Second IEEE World Congress on Computational Intelligence (WCCI'98) with Fifth IEEE Conf. Evolutionary Computation (IEEE/ICEC'98)*, volume 1, pages 1–8, Piscataway, NJ. IEEE Press.

Shawe-Taylor, J. and Zerovnik, J. (1995). Analysis of the mean field annealing algorithm for graph colouring. *Journal of Artificial Neural Networks*, 2:329–340.

Sims, K. (1994). Evolving 3D morphology and behavior by competition. *Artif. Life*, 1(4):353–372.

Smith, J. M. (1982). *Evolution and the Theory of Games*. Cambridge University Press.

Smith, J. M. and Price, G. R. (1973). The logic of animal conflict. *Nature*, 246(5427):15–18.

Stanley, K. O. and Miikkulainen, R. (2003). A taxonomy for artificial embryogeny. *Artificial Life*, 9(2):93–130.

Swan, L. (2009). Synthesizing insight: artificial life as thought experimentation in biology. *Biology and Philosophy*.

Takeuchi, N. and Hogeweg, P. (2008). Evolution of complexity in RNA-like replicator systems. *Biology Direct*, 3(11).

Thomas, R. (1991). Regulatory networks seen as asynchronous automata: A logical description. *Journal of Theoretical Biology*, 153(1):1 – 23.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59:433–460.

Turing, A. M. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641):37–72.

Vohradsky, J. (2001). Neural network model of gene expression. *The FASEB Journal*, 15:846–854.

Voit, E. O. (2000). *Computational Analysis of Biochemical Systems : A Practical Guide for Biochemists and Molecular Biologists*. Cambridge University Press.

von Neumann, J. (1953). The theory of automata: Construction, reproduction, homogeneity. The Theory of Automata: Construction, Reproduction, Homogeneity, in Burks (1966), pp. 89-250. Based on an unfinished manuscript by von Neumann. Edited for publication by A.W. Burks.

Watson, J. D., Baker, T. A., Bell, S. P., Gann, A., Levine, M., and Losick, R. (2003). *Molecular Biology of the Gene.* Benjamin Cummings, fifth edition.

West, S. A., Diggle, S. P., Buckling, A., Gardner, A., and Griffin, A. S. (2007). The social lives of microbes. *Annual Review of Ecology, Evolution, and Systematics*, 38(1):53–77.

West-Eberhard, M. J. (2003). *Developmental Plasticity and Evolution.* Oxford University Press.

Williams, G. C. (1996). *Adaptation and Natural Selection.* Princeton University Press.

Wong, P., Gladney, S., and Keasling, J. D. (1997). Mathematical model of the lac operon: inducer exclusion, catabolite repression, and diauxic growth on glucose and lactose. *Biotechnology progress*, 13(2):132–143.