Biose

Iago Bonnici, Abdelkader Gouaich

LIRMM

M. Iago BONNICI
Supervisor: Dr. Abdelkader GOUAICH

Tuesday 29th March, 2016
Montpellier

Biose

towards a mechanistic model of the living

Research report

Contents

Introduction

1 Context

My name is Iago Bonnici. I have been studying Biology at the *École Normale Supérieure* (ENS) in Paris for four years and graduated there in early summer 2015 (M2). There and around the world, I have been specialized in the fields of Ecology, Biodiversity and Evolution. In particular, I have been instructed in the use of *modelling* among these fields.

During my last school year — the current year —, my School supervisor Dr. Régis Ferrière has granted me permission to explore original ideas concerning modelling of the living. I am currently working on this project as an intern student at the Montpellier Laboratory of Informatics, Robotics and Microelectronics (LIRMM), under supervision of Dr. Abdelkader Gouaich. By the end of the year, I am willing to defend it as a formal PhD project before the ENS *Allocations Spécifiques* (AS). If it were accepted, we would be able to continue this work.

This document stands as an intermediate research report, summing up the whole current state of the project: positionning, main ideas, current work and perspectives.

2 Background

This project stands at the interface between Biology and Informatics. Like population dynamics [51] or cellular automata [25], it can be refered to as a plain modelling activity about the living.

The *living world* is indeed our first topic and motivation. It is huge and intricate, both mesmerizing and puzzling no matter the scale one looks at it or the particular detail one is gaping at. In order to study this object, we are looking for ways to represent it — or at least some of its aspects — into abstractions we call "models of the living" or "life models". Building such models requires a careful examination of the targetted object. It also demands careful thoughts about its very nature and the underlying dynamics. In the first place, we therefore expect that this activity could teach us anything significant about the living.

2.1 Invariants and Abstraction

When it comes to examinate the living with care, we notice that, even though it is enormous and complicated, there seem to exist mysterious *invariants* among it. These are processes and properties which can be spotted similarly in various systems even if the latter turn out not to be made of the same objects, or they do not obey to the same rules. Here are a few examples:

Invariance among situations

It regularly turns out that one biological model can describe very different biological systems and their dynamics. Invariant processes like *predation* for instance exhibit similar properties in various systems, may they be fish predated by sharks, rabbits predated by foxes or mice predated by eagles. This is were models derived from the principles of Population Dynamics have already proven to be fairly useful for many decades in describing the living [51]. They might need to be refined to support other invariants like *cooperation* dynamics also exhibited by various systems like plants and pollinators, ants and acacia, lichens or human societies [12]. Trickier invariant phenomena like *eusociality* sometimes need totally new ideas to be incorporated into the models [78], but they describe equally well termite colonies, ant colonies, bees colonies or naked mole-rat colonies. Perhaps the most spectacular invariant among all living systems has been pointed out by Darwin in 1859 [29]: *evolution* takes place in every reproducing population on Earth, no matter whether they are plants or algae, fungi, animals or microbes.

Invariance among scales

In addition, it turns out that similar processes can be spotted in the living regarless of the *scale* one is studying it at. This is less obvious because there is very much fewer diversity in “scales of the living” on Earth than there is “horizontal diversity” within each of them. Indeed, biological scales are confined between the atom level and the biosphere level, whereas Earth is tall enough to receal an extraordinary profusion of various biotopes and ecosystems at the geographic level, of various communities, of various species at the species level, of various individuals, of various types of cells, of various proteins at molecular level *etc.* As a consequence, the existence of such scale-invariants is more debated than the existence of "horizontal" invariants within scales.

However, a lot of thoughts have been put in considering that evolution could also act on *groups* of organisms similarly to the way it acts on organisms [91]. As multicellular organisms ourselves, we can assert that such processes have necessarily happened in the past [48, 83]. Models trying to capture this relate a lot to Game Theory and with a phenomenon refered to as the *Tragedy of the Commons* [49]. It turns out that this phenomenon is another example of such scale-invariants. Indeed, it can cause collapsing of any structured cooperative system regarless of its scale, should it be a human society struck by internal "cheater" vandals or one of its citizen dying from internal "cheater" cancerous cells [9]. Other kinds of models, like models developped in the field of *epidemiology* [77] and the study of disease dynamics at human populations scale, might turn out to describe surprising processes quite well at molecular level, like the propagation of a nocive conformation in prion proteins [26]. More generally, loose connections can be made between any pair of resembling structured processes at different scales. Compare for instance the definite development of organisms [18] like the ontogeny of a bird and the definite development of communities like a neat plant successions in a young forest [14]. The relevance of such connections is still uncertain, but they do deserve to be pointed out in our view.

Invariance among systems

This is the most general type of invariants one might be confronted to in the sense that the previous two are both instances of this one. If, say, gravity were not effective on Earth, there could still have been weightless *predators*, *parasites*, *ontogeny*.. and the *evolution* process would certainly have occured, even though with different outcomes. All these processes seem therefore not dependent on the actual *rules* of a particular world, nor on its actual constitutive *objects*.

If this is true, then properties of biological systems can very well be exhibited by other systems, should them belong to other scales (see previous section), or should they even be *non-biological* systems. As far as engineering and technology are concerned, this feeds all activities one would refer to as *Biomimetics* [64] — think about resistance and resilience of biomimetic materials for instance. Regarding *evolution* and *adaptation*, there has been a lot of interest in the development and study of genetic algorithms [41] and genetic programs [59], because they can find good solutions to intricate problems without anyone needing to derive them, just like actual biological populations do [29]. One is also confronted to familiar processes known as "predation", "competition", "symbiose", "parasitism", "adaptation" when one studies finance, economics and the relations between lucrative companies: this feeds a whole field of research known as *Ecology of Economics* [36]. This is where the abstract nature of many life models like Population Dynamics prove to be particularly useful, since their predictions still hold when "life dynamics" are studied within non-biological contexts.

There also exist very interesting artificial systems like Conway's famous "Game of Life" offered in 1970 [27], Thomas Ray's "Tierra" (1992) [84] followed by "Avida" (1994) [8] or Karl Sims's striking "artificial creatures" evolved in 1994 [89]. The constitutive *objects* of these systems are nothing but virtual, computer data objects *ruled* by nothing but an explicit computer program. But they somehow exhibit many interesting features of the living (see section 2.2). In this respect, they constitute quite relevant "life models" in the sense that they seem to capture fundamental constituents of the living, and give us an insight into the internals of these mysterious invariants.

Noticing all these invariants requires one to exert a natural ability for generalization. This in turn naturally entails a mental abstraction process and the building of abstract representations of them. These abstract representations enable the invariants to be translated into other systems, no matter then whether they differ by the situation (sharks *vs.* eagles), the scale (prions *vs.* patients) or more generally the whole "world" they instanciate within (Earth biotopes *vs.* machines architecture). All these considerations are essential to our modelling activity.

Considering such invariants among the living and the relevance of abstraction in this context, John von Neumann offered a very interesting view according to which « *Life is a process which can be abstracted away from any particular medium* ». This would mean that all *features* of the living (see next section) do not rely upon the actual "living" objects we are familiar with, but only on their properties and their relations to each other. In this sense, any set of exotic objects might exhibit these features as well even if they are not made out of flesh, wood, cells, DNA, proteins *etc*. They just need to somehow display an instance of these abstract properties and relations. With respect to this view, life on Earth appeared because prebiotic chemical objects have somehow operated this instanciation.

Throughout this project, we will undoubtely think this way and consider abstract objects regardless of their « *medium* » in the sense of von Neumann. They might be pieces of code, cells in an automaton grid, robots, vectors, species, mathematical variables, organs, chemicals, electronic components, data bytes, emotions, evolving words, lucrative companies, *etc*. They will all have something in common — they exhibit features of the living —, but cannot be refered to as "biological" objects in order not to feed a latent semantic war. We will then refer to them as plain *biotic* objects. They constitute *biotic* systems supporting *biotic* features, exhibiting *biotic* properties and hosting *biotic* processes. We have made this choice because *biotic* is both pretty similar to "biological" in the sense that they share the same etymology and pretty different in the sense that it is, to knowledge, much less used and debated. In a nutshell, we intend to offer *biotic* as a generalization of "biological", so that the living system we are studying is freed from the incidental nature of its constitutive objects.

Biose (french for *Biosis*, from βίος-ωσις, the *living condition* or *living process*) is all about building abstract biotic models.

2.2 The biotic features

In this context, one particular set of questions is often raised. It relates to the very definition of the word "alive" and to all preconceptions people cherish about it. Can cells in an automaton grid be considered "alive"? What splits "life" apart from the "non-living", if anything does so? This is an alluring, metaphysical semantic debate. However, we are not interested in this debate here, because we trust it has no particular biological nor *biotic* relevance.

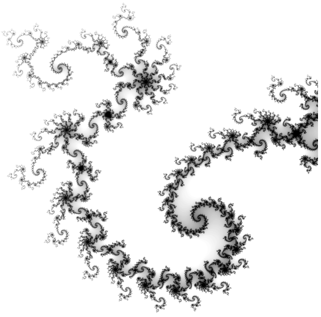
Instead, we will try to emphasize every interesting invariant property of systems commonly refered to as "biological" or experienced as "biotic". We find them relevant, not because we think that they reveal anything essential about the underlying dynamics of the living, but because they allow discriminating of existing life models (see table 1 page 20). To our sense, listing them is a first step towards answering the question: "What is interesting about the living world?". We collect these invariants into loose categories which we call the biotic features. They are listed and described hereafter along with commonly related questions regarding them.

To our opinion, a biotic model does not usually aim to answer the definition of the word "life", but instead tries to support one or some of these features, or to answer one or some of these questions:

Emergence

It is a common opinion that life somehow appears from the non-living. Things get organized while we would expect them to stay pretty messy in the first place [56]. This feature relates to very fundamental questions about the origin of life and to the mysterious behaviour of complex systems [47, 34, 35]. It is very difficult to deal with because explaining complex behaviours is still difficult to anyone today [97].

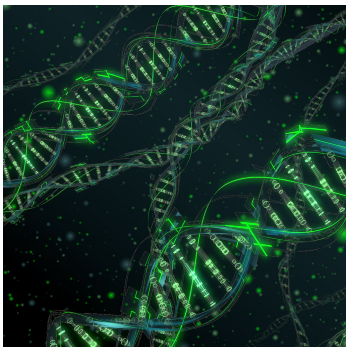
- What is the origin of auto-organization?
- How does complexity works?
- How did life start?



Information

Biotic objects' properties appear to be encoded, either directly or indirectly, by some kind of information one calls "genetic" or "epigenetic" in biology. This information is intimately linked to the objects in the sense that their are somehow holding it, expressing it. But it is also rather independent in the sense that objects are just reflects of this information in the world, a subservient embodiment of itself [30].

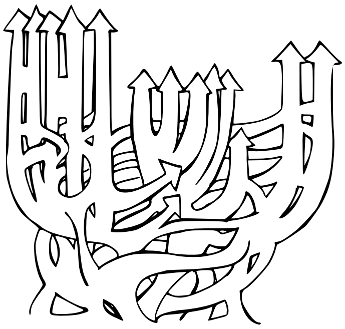
- Is this encoding made explicit by the model?
- How does this encoding work?
- How does information evolve over time?
- Are information dynamics ultimately more essential than actual objects dynamics? [30, 45, 46]



Transfer

Biotic Information is somehow exchanged from biotic objects to other biotic objects. These phenomena are known as “transmission” in biology, “heredity” , or “sex” [15], “conjugation” in bacteria [99], “gene transfers” by viruses [22], “introgression” between species [50], or even “GMOs” and horizontal tranfer of more abstract pieces of information like “memes” [30]. The very existence of these phenomena suggests that life’s internal language is universal.

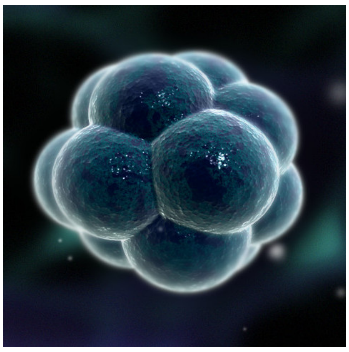
- Does the model support these?
- What are the underlying mechanisms?
- How does it spur life’s dynamics?
- What is the common language?



Development

Biotic objects constantly transform. In particular, their form of interest is built from a succession of earlier stages forms. This is called “development” and “ontogeny” in biology and relates to the way biotic Information is *decoded* into actual biotic objects. Altering the developmental process of an object is altering the actual object. This entails heavy repercussions on Evolution, and the process must be somehow carefully regulated [18].

- Does the model take this into account?
- What are the underlying mechanims?
- How is Information decoded?
- How do these transformations happen?
- How is the process regulated?



Environment

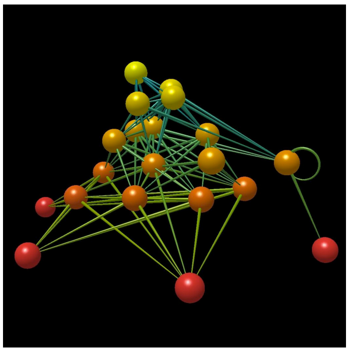
Biotic objects work within an environment they interact with. They are both responsive to it and acting on it. In particular, they exhibit a certain kind of *behaviour* within it and a certain *plasticity*. Biotic objects can obviously be considered as the environment of other ones. This is a transversal feature because it also concerns every other feature.

- Is this interaction explicited by the model?
- How do biotic objects / processes affect the environment?
- How does environment affect biotic objects / processes?
- How do other biotic features respond to this interaction?

Interaction

Biotic objects also interact with each other in a way that is not "neutral" like they would do if some were just considered as the Environment of the others. It is "not neutral" because some kind of *resource* is involved, and they are confronted to each other by some kind of "game" in the sense of rationality and Game Theory. This constitutes a substantial part of *Ecology* in biology, with all phenomena described as "symbiosis", "predation", "parasitism", "amensalism", "facilitation", "competition", *etc.* [96] but also "cheatery", "cooperation", "cancer" [49, 91, 9]

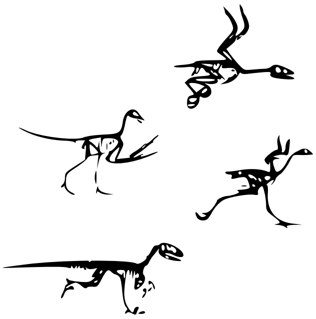
- Can the model deal with such interactions between objects?
- What do they imply in terms of dynamics?



Evolution

This transversal feature is perhaps the most famous one. Biotic objects, as they somehow replicate themselves over successive "generations", change over time. Not necessarily because each object changes, but because each replicate accumulates alterations. Henceforth, what one gets after many generations is not what one had at the beginning [29].

- Can the model reproduce this?
- How does this relate to other living features?
- What are the causes of variation as far as Information is concerned?
- How do they affect the object reconstruction as far as Development is concerned?
- What are the evolution dynamics? Where does it go? What is it driven by?



Adaptation

Biotic objects somehow change over time. This phenomenon can be directly related to their changing Environment, or it can be drawn out along the Evolution process. Anyway, anytime one snapshots them, they often turn out to fit in their environment particularly well, and to be particularly successful in their activities.

- Can the model reproduce this?
- How to measure "fitness" or "successfulness" in an given environment?
- Does Adaptation lead to optimal situations?
- Can biotic objects be well-suited to anything?
- Does successfulness hold good when Environment changes?
- Does Adaptation occur at the individual scale or at a populational level?

Coevolution

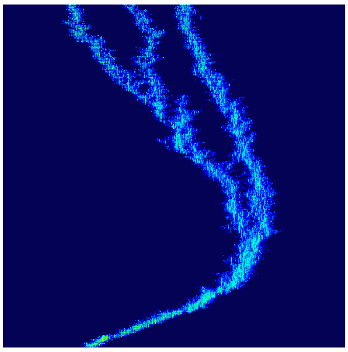
Biotic objects are of many different types, they interact together and they evolve together. Therefore, their types and their interactions must evolve in an integrated way.

- How do Interactions evolve accross generations?
- Are they maintained, changed, reinforced, inverted?
- Do them somehow cycle?
- Can the model reproduce this?
- How does Evolution of a biotic object depend on the Evolution of other ones?

Speciation

As generations pass, biotic objects of one type give rise to biotic objects of several types. New types look adapted to new sets of Environments and Interactions. They might also be very different from one another and from their ancestor types.

- Can the model capture this?
- Can it handle all intermediate cases like biological processes known as "ring species" [53], "hybridogen speciation" [21], "introgression" [50] *etc.*?



Scaling

A group of biotic objects working together can eventually work as one single biotic object. These are wonders of biology called “multicellularity”, “group selection”, “selection level shift” [91] with fascinating and hard-to-qualify systems like early multicellular organisms [48, 83], ant colonies [78], slime molds [87], Pando [76].

- Can the model deal with such objects whose multiplicity can’t easily be defined?
- Can group selection happen?
- How relevant is it to consider symbionts as a single organism?

Novelty

Evolution can lead biotic objects to acquire properties they never had before. They are not just simple alteration of previous properties in the sense that they fundamentally change the rules or the underlying dynamics of the system.

- Can the model deal with objects which regularly pop out of their frame?
- Can new traits appear?
- Is Evolution open-ended?
- Can highly unexpected things happen in the system?

Once again, these biotic features may have no particular, fundamental relevance. They constitute nothing but a prism one may watch the whole living system through (see figure 1 page 23).

3 Problem Statement

In this project, we consider working a path towards a utopic *complete* and abstract life model. It would be *complete* in the sense that it would be able to represent every invariant property of the living and every invariant process it undergoes. It would be abstract in the sense that the latter could translate into any other biotic system.

Reaching this goal is impossible for two reasons. First, achieving *completeness* would require at least that one has been able to study, understand, list and formalize all “living processes and properties”, which already is the utopic goal of biological Science. Second, achieving abstraction would mean that all biotic systems could be accurately described by only one model, which is impossible by essence of the modelization process.

However, we still think that it is an interesting target to get closer from. For instance, a necessary step towards *completeness* is the support of at least all biotic invariants we have been able to point out, which is a more realistic goal. In our view, a model is candidate for being *complete* only if it supports at least all previously described biotic features. In the next, we shall refer to this restrained meaning of *completeness* or *weak-completeness*. On the other hand, a necessary step towards abstraction is that the model statement should rely on no incidental, particular *medium* in the sense of von Neumann, which is a more realistic goal as well. With a little work, we can expect that such a model could describe at least *many* various biotic systems. In the next, we shall refer to this restrained meaning of abstraction.

In the end, the first goal of this project is to design a biotic model:

- supporting all previously listed biotic features (*weak-completeness*)
- whose statement does not involve incidental properties of particular objects (abstraction: do not rely on physics, biochemistry, neigbourhood rules in an automaton grid, *etc*.)

Unfortunately, there seems to be an incompatibility between Emergence support and abstraction of the model. Consider the following:

- Emergence has to deal with the way particular objects (*e.g.* chemicals, data bytes, companies, phonemes) in a particular *medium* (*e.g.* Earth, machine architecture, stockmarkets, languages) aquire biotic properties.
- Abstraction suggests that, once these abstract properties (*e.g.* self-replication, resource exchange, heredity) are acquired, all the remaining biotic features can eventually develop within this medium.

⟶ A model featuring Emergence *must* deal with the particular, actual nature of its constitutive objects. But an abstract model must *not*, in order to translate into other systems.

As a consequence, we think that an abstract life model cannot support Emergence phenomena.

Henceforth, we shall not try to answer any question related to Emergence and the early origins of life within this project, considering this as Biose's first limit. We shall start from the assumption that biotic, organized objects do exist. Once this assumption has been made, we wish to represent these objects in a way that makes it easy then to support every other biotic feature, regardless of the nature of the system.

In a nutshell, Biose is an abstract model of the very basic properties of biotic objects, which we think can behave together as a system supporting all the previously listed biotic features except the one questionning their initial apparition. In this sense, it attempts to stand as a mechanistic model of the living.

4 State of the Art

There already exist many life models. They are designed to support various biotic features — with respect to our view — and they address various use cases. Let us sketch a brief overview of them.

4.1 Biology

Life models in the field of Biology are obviously built with a real scientific care. They may aim to be *explanatory*, "mechanistic" in the sense that their expression implements explicit hypotheses about the underlying dynamics of the living (fundamental biology). But one may also be concerned with their *predictive* power, because this is what applications of biology demand (*e.g.* biodiversity management, conservation ecology, health care).

It is sometimes quite straightforward to meet both concerns. For instance, models developped within the theory of Population Dynamics [51] are heavily used today for both applications and fundamental research. In particular, Epidemiology and the modelling of disease dynamics [77] has become a crux of societal health care concerns. They support very well Interaction between biotic objects and their relation to Environment (see table 1). Yet, one might feel curious about the way a nocive mutation spreads into a city *via* genetic transmission, or about the way innovative alleles are selected in a protected species. These new concerns have to deal with Information, Transfer and Evolution features in our view, which are better supported in the field of Population Genetics [66]. Both these categories of models can be expressed in mathematical terms, so they can provide interesting analytical results. However in situations where keeping an analytical approach is too difficult, like when it comes to deal with spacially explicit situations, Emergence phenomena or Developmental processes, falling back on simulations like Cellular Automata has also proven to be quite useful [25].

Biological systems' complexity might make it too hard to produce decent applicated predictions. At this point, life models are still interesting because they offer mechanistic views of living processes. This is were fundamental biology takes over. For instance, integrating tools and results from Game Theory has lead to very relevant new visions on the living [96]. With these visions, it is possible to deal with all Interaction phenomena plus Evolution with a quite unified language, and even with the puzzling Scaling feature when it comes to cooperative systems and the tragedy of the commons [49]. Quite recently, a new integrated approach taking after Population Dynamics, Population Genetics and Game Theory has broken through with a powerful combination of both analytical results and individual-centered simulation opportunities: the field of Adaptive Dynamics [32]. In particular, it can feature with elegance intricate situations regarding Speciation phenomena (see table 1). However, it has not been concerned yet with Developmental processes within individuals. This makes sense since the approach is "individual-centered", and so individuals are considered as atoms of the models. But it is therefore inadequate to represent biotic objects *developping* into one another, or groups of objects mysteriously Scaling into one "individual".

In order to deal with these phenomena, one needs to change paradigm and get interested in the field of Ontogeny and the study of Development in biology. Discovering of very generic key processes in this field like the existence of regulatory genes, homeotic genes and morphogenetic gradients have been huge steps towards understanding Development and modelling it [90]. Further away from the ecological scale, there has been a lot of interest in “wet systems” and studies about prebiotic chemistry and exobiology. Since the famous experiment by Miller and Urey in 1953 [74] and the retrospective advent of an RNA world [65], these “wet” life models are the best hope we have to understand origins and the early Emergence of life on Earth. However, by essence, these results cannot be *abstracted* away from biochemistry and translated into non-biological systems (see section 3).

All these life models developped in the field of Biology exhibit a particular scientific goal. It ultimately relates to studying the living and understanding it better.

4.2 Informatics and Technology

Life models developped outside the context of Biology address quite different use cases. In general, one is more concerned with the ultimate *properties* they exhibit than their actual biological relevance. This necessarily makes them very interesting because they would not have been developped if they were not. Biology just persists then as a vague "biological inspiration". Still, they rely on the idea that grasping some key features of the living and implementing them into a model can be quite beneficial.

Obviously, the field of Biomimetics perfectly sticks to this philosophy [64]. But since they are much focused on actual products, their constructions cannot be « *abstracted away* » and translated to other systems (see table 1). In contrast, Genetic Algorithms [41] and Genetic Programs [59, 100] constitute a wonderful example of models clearly supporting biotic features like Evolution and Adaptation in an *abstract* way. This makes it possible to use them in solving almost any kind of problems, and makes them dramatically useful. In general, all activities related to *Artificial Life* [80, 1] also look promising when it comes to design artificial systems exhibiting innovative properties. In Robotics for instance, one can now animate arbitrary machines with very interesting behaviours [95, 89], even metamorphic and swarming robots featuring Interaction, Scaling, and obviously reaction to their Environment [7]. Other interesting biotic features like robustness and adaptiveness of Developmental processes in reaction to their Environment are currently being explored in the field of Ontogeny and Artificial Embryogeny [94, 16, 72, 73]. These obviously link to Multi-Agent approaches of modelling, which are one of the numerous paths explored towards Artificial Intelligence [62, 20].

4.3 Art and entertainment

Life models have also been used in *arts* for a while, certainly because properties of the living are rather mesmerizing. For instance, there exists bio-inspired constructions based on artificial life principles and refered to as "Art Generators", whose only purpose today is to produce pieces of art [98]. They are much focused on noticeable, sensitive properties of the living like puzzling patterns Scaling into one another, or gradual Evolution of them. But they barely bother with the biotic relevance of underlying processes like Information, Transfer *etc*. (see table 1). Interestingly, Karl Sims offered his procedure to evolve artificial creatures in 1994 as a graphical *animation* procedure [89].

Another surprising use case of life models is their use in *games*. Most basically, they can be used as a plain poetical universe hosting any kind of exotic rules [13]. "Spore", a video game relased by Maxis within Electronic Arts in 2008 [11] is a good example of such an approach, along with a flourishing set of internet games [23, 2, 3]. But the biotic features that life models exhibit can also be used as *constituents* of the actual game dynamics. For instance, the ambitious "SimLife" published by Maxis in 1992 [60], was a sophisticated cellular automaton featuring many interesting biotic dynamics regarding Evolution, Coevolution, Interaction and Adaptation. They were actually part of the game. They also are in a recent board game called "Evolution" (North Star, 2014) [28], which succeeds in supporting the same kind of features with very simple rules. These link "artificial life" and the development of biotic models to some kind of ongoing industry. But like in biomimetics, these model are so much focused on actual products that their principles can hardly be abstracted away and translated into other systems.

4.4 Didactics and Proof of Concept

From a strictly productivist point of view, some life models seem not to fulfill any particular use case. Yet every life model has a certain *didactic* relevance in the sense that it teaches us something about the living. And the following ones have sent theirs to their height which is the reason why they are still famous.

The first one is Conway's "Game of Life" offered in 1970 [27], which is a very simple cellular automaton. Its basic objects and rules are loosely inspired from the functionning of biological cells, but the dynamics it exhibits strike in the fact that they evoke early Emergence phenomena and Development of organisms into a virtual Environment they interact with.

The second one is Ray's "Tierra" offered in 1992 [84], soon followed by "Avida" in 1994 [8], which are quite puzzling computer programs. Executing them populates one's machine RAM module with virtual objects made out of computer code. The latter are supposed to represent "organisms" able to reproduce themselves and competing for CPU time. As the simulation goes and generations pass, the machine ends up hosting a wonderful diversity of such objects and undergoes familiar processes directly relating to what we call Interaction, Evolution, Coevolution and Adaptation features of life.

The third one is Sims's procedure offered in 1994 to evolve striking "virtual creatures" [89]. It is a genetic algorithm whose evolving objects are quite generic formal grammars [71] (like

	weak-completeness													
	Emergence	Information	Transfer	Development	Environment	Interaction	Evolution	Adaptation	Coevolution	Speciation	Scaling	Novelty	…	abstraction
Population Dynamics	×	×	×	×	yes	yes	hard	hard	hard	×	×	×		yes
Population Genetics	×	yes	yes	×	hard	hard	yes	hard	yes	yes	×	×		yes
Game Theory	×	hard	×	×	hard	yes	yes	yes	yes	hard	yes	×		yes
Adaptive Dynamics	×	yes	hard	×	yes	yes	yes	yes	yes	yes	×	×		yes
Genetic Algorithms	×	yes	yes	×	hard	hard	yes	yes	hard	hard	×	×		yes
Genetic Programming	hard	yes	yes	×	hard	hard	yes	yes	hard	hard	×	yes		yes
Cellular Automata	yes	hard	×	yes	yes	hard	hard	×	hard	hard	yes	×		hard
Ontogeny	hard	yes	×	yes	yes	×	hard	×	×	hard	×	yes		yes
L-Systems	hard	yes	×	yes	×	×	×	×	×	×	yes	×		yes
Biomimetic Materials	yes	×	×	hard	yes	×	×	yes	×	×	×	yes		×
Artificial Intelligence	hard	×	×	hard	yes	hard	×	yes	×	×	yes	yes		yes
Wet systems	yes	yes	hard	hard	yes	×	hard	×	×	×	hard	hard		×
"Evolution" board game	×	×	×	×	hard	yes	yes	hard	yes	×	×	hard		×
SimLife	×	yes	hard	×	yes	yes	yes	yes	yes	×	×	×		hard
Spore	×	×	×	hard	hard	hard	hard	×	×	×	×	×		×
Art Generators	yes	×	×	yes	yes	hard	yes	×	hard	×	×	yes		×
Karl Sims's Creatures	×	yes	×	hard	yes	yes	yes	yes	yes	hard	×	yes		hard
Tierra, Avida	×	yes	hard	×	hard	yes	yes	yes	yes	yes	×	×		hard
Biose	×	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes		yes

Table 1: Illustrative overview of existing biotic models and approaches. The first 12 columns represent interesting biotic features as described in section 2.2 and our concern for *weak-completeness*. The last one tells whether or not the model supports abstraction in the sense that processes and properties it exhibits can be translated into any system. yes labels models obviously supporting a feature, or even *designed* to address it. hard is used when supporting the feature would require substantial patches to the model, reinterpretation of it, or if the approach has just not been designed to address it in the first place. × is used when the model does not support the feature, cannot represent it or it would be *very* hard.

L-systems [82]) and arbitrary control solutions [95, 89] powering Development of virtual robots in a 3D rigid physics environment. They can adopt a wonderful variety of forms and behaviours, blindly used in an Evolution-Adaptation procedure to make them achieve arbitrary goals.

When it comes to realizing that biotic processes and properties can be exhibited within non-biological systems, the latter life models are the best ones we are aware of. However, they hardly represent an abstraction of the living in the sense that they are also much focused on the actual objects they are dealing with (square cells and their neighbours, computer code and CPU resource, 3D rigid boxes and their physical actuators). It is hard then to reuse their "rules" and "principles" into other kinds of systems, nor to extract any general statement about the true bases of biotic invariants.

5 Discussion

All these life models and modelization approaches somehow attempt to represent the living. Should we project them through the "prism" that our previously defined biotic features constitute (see section 2.2), we would end up with something close to the illustrative table 1. In this table, we can see that they all succeed in supporting the features they were initially designed to address, and sometimes more.

However, none of them supports both abstraction and *weak-completeness* in the sense that they cannot support every biotic feature in any kind of system (see section 3). This is obviously not an issue since, to our knowledge, none of them has actually been aiming to such a goal in the first place. This is where Biose attempts to stand with an original attitude towards modelling the living.

From now on, we will be primarily concerned with these enigmatic invariants noticed among all biotic systems, and with these elusive "biotic processes and properties". As stated in section 2.2, the only relevance of what we have called biotic features so far is that they could discriminate among life models refered to in our **State of the Art**. In a word, they just helped us build table 1. We do think that the biotic features somehow connect to the invariants, packing them into loose categories, so we will keep refering to them. However, we do not think that they provide any direct key to the fundamental understanding of them. Instead, we trust that

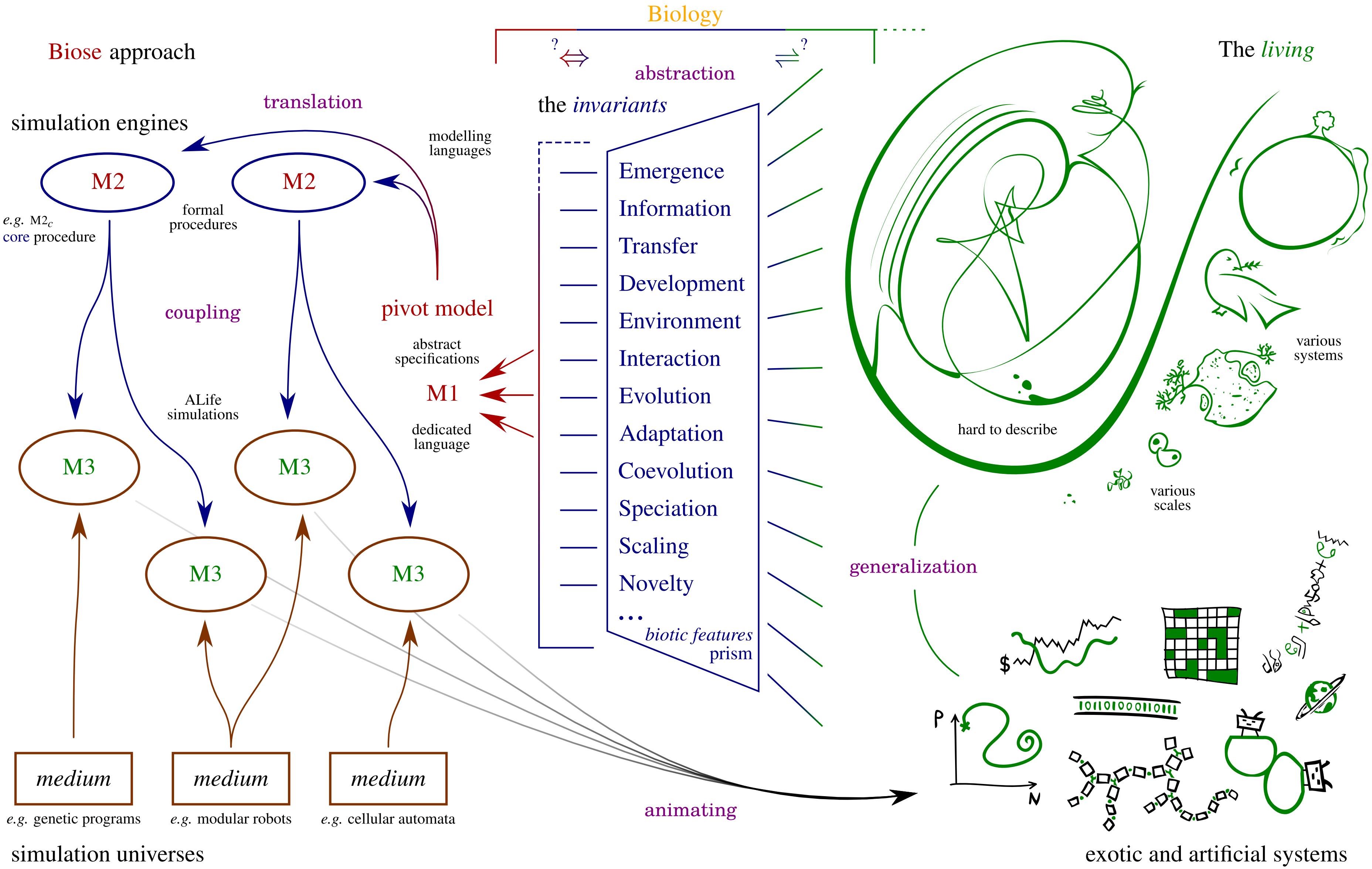


Figure 1: Illustrative overview of Biose approach.

all invariants derive from more essential, abstract foundations of the living, which we shall try to capture into a biotic *pivot model*. The latter would constitute an abstract basis from which any biotic feature of interest could be derived (any but Emergence). And this derivation could be done regardless of the actual system used, provided it can still match the pivot model. If such a model exist, we think that any other model aiming for both *weak-completeness* and abstraction can be somehow expressed in its terms. Biose is an attempt to identify essential, abstract foundations of the living and to express them into a central, consensual model.

In order to check validity of such a model, we will have to assert that one particular expression of it is relevant. In other words, we will have to check for its ability to entail any invariant biotic process or any invariant biotic property in any biotic system (see **Evaluation** section 9). Using formal reasoning and mathematics to perform such a derivation would require that one is comfortable dealing with the emergent, possibly chaotic behaviour of complex systems [47, 34, 35], which is still something no one might claim being able to do today [97]. In addition, it would require the "invariants" to be formally defined in the first place — or at least commonly agreed among biologists —, which is definitely not the case either (What is *predation*? What is a *population*? What is a *group*, an *individual*, a *trait*? What is a *species*? What is Scaling? *etc.*). For these two reasons, we suggest that *simulation* tools should better be used as a first approach.

In the next chapter, we shall offer our proposition.

Proposition

In this chapter, we shall describe Biose in detail and the current state of the project. It is the life model we intend offering.

Several media exhibit biotic invariants. We therefore expect our life model not to depend on the medium. In addition, there exist several ways to model biotic invariants regardless of the medium. We therefore expect the model not to rely on a particular modelling language. In order to meet both concerns, we are offering this roadmap sketched figure 1:

- section 6: first build a *pivot model* M1 with our own abstract language
- section 7: translate M1 to explicit modelling languages to get M2: biotic procedures
- section 8: couple M2 to various media to get full M3 models: artificial life simulations
- section 9: observe M3 and check them for *weak-completeness* regarding the biotic features

The overall idea is to study relations between M1 and the biotic features, because they constitute fundamental Biology (see figure 1). However, this cannot be done directly today for the two reasons developped in section 5. These are the reasons why we still need to build M2, M3 and to fall back on simulations for now.

6 Pivot model

This section states Biose pivot model: M1. Since we do not want it to depend on a particular medium, it will be elusive about "environmental" objects and properties. Since we do not want it to be constrained by a particular modelling language, we shall use a small abstract, dedicated language to state it.

- section 6.1: offer the language in which M1 shall be stated
- section 6.2: state the global structure of M1

- section 6.3: state the global dynamics of M1
- section 6.4: state basic constraints on M1

Specifications of the pivot model are roughly divided up into five *categories*. Each category is supposed to bring more interesting properties to the model:

Activity: this category describes integration of biotic objects into their Environment.

Holonics: this category brings flexibility to the model basic objects. We think that this flexibility is essential to Scaling, Speciation and Development.

Resource: this category introduces resource dynamics which we think underlie all invariant processes related to Interaction.

Encoding: this category explicits the way Information is dealt with in the model, how it is decoded during Development and how it is Transfered among objects.

Alteration: this category triggers changes within the former base. It is how we intend to support Evolution, Adaptation, Coevolution and Novelty.

6.1 Language

This section offers the basic language we shall use in order to state the pivot model M1. This language is dedicated to Biose and should be translated to any other, more explicit modelling language before defining an explicit biotic procedure (see figure 1 and our proposition section 7).

Objects and relations

The pivot model M1 involves particular *objects* (like a, b) connected to each other by some kinds of *relations* (like $a — b$).

The language has three levels of abstraction:

components level l_1 : basic components of the language like a or —

generic level l_2 : generic statements like $a — b$ representing *all* objects of type a and b

specific level l_3 : specific statements like $$\begin{tikzcd} a_1 \arrow[dr, dash] & b_1 \\ a_2 \arrow[r, dash] & b_2 \end{tikzcd}$$ representing *instance* objects of type a and b

Aliasing

The first, practical relation we need is the *alias* relation, stated as follows on generic level l_2:

$$a \longrightarrow b$$

This means that some objects a are called b. They essentially are the same objects being given different names. For instance, on specific level l_3, one may get:

$$a_1 \qquad a_2 \qquad a_3 = b_1 \qquad a_4 \qquad a_5 = b_2 \qquad a_6$$

Here, a_3 and b_1 are two names for the same object, which may be different from a_2. In contrast, a_6 is only known as a_6. Objects with no names yet (like a_7, c_{12}, ♉$_0$) do not exist to the model, even though they might be there somewhere.

If $a \longrightarrow b$, no object b is not an object a.

In addition, two other, more informative types of relations will be used a lot: *composition* and *influence*.

Composition

When some objects a are *constitutive* elements of some other objects b, we shall draw a *composition* relation between them, stated as follows on generic level l_2:

$$a \rightarrowtail b$$

$a \rightarrowtail b$ does not imply that b is only made of a. Also, it does not imply that every a object contributes to constituting a b. However, it does reveal a certain kind of structural hierachy between the two: b is made of objects a; if there is no a, there is no b.

Cardinalities

Composition comes into four different cardinalities on level l_2:

1. $a \overset{\therefore \quad \therefore}{\rightarrowtail} b$: any composition relation may be drawn between a and b

2. $a \overset{\therefore \quad \cdot}{\rightarrowtail} b$: a b may only be composed of one a; only one arrow may point to it

3. $a \overset{\cdot \quad \therefore}{\rightarrowtail} b$: an a may only be part of one b; only one arrow may start from it

4. $a \overset{\cdot \quad \cdot}{\rightarrowtail} b$: any b has an exclusive relation to one a; one arrow end per object

For example, here are possible situations on the specific level l_3. Each situation corresponds to a different cardinality. Faded arrows are not allowed:

1. $$\begin{tikzcd}
 & b_2 \\
a_1 \arrow[ur, tail] \arrow[r, tail] & b_1 \\
a_2 \arrow[ur, tail] &
\end{tikzcd}$$

2. $$\begin{tikzcd}
 & b_2 \\
a_1 \arrow[ur, tail] \arrow[r, tail] & b_1 \\
a_2 \arrow[ur, tail, opacity=0.2] &
\end{tikzcd}$$

3. $$\begin{tikzcd}
 & b_2 \\
a_1 \arrow[ur, tail, opacity=0.2] \arrow[r, tail] & b_1 \\
a_2 \arrow[ur, tail] &
\end{tikzcd}$$

4. $$\begin{tikzcd}
 & b_2 \\
a_1 \arrow[ur, tail, opacity=0.2] \arrow[r, tail] & b_1 \\
a_2 \arrow[ur, tail, opacity=0.2] &
\end{tikzcd}$$

These will enable structural statement of our pivot model M1 (section 6.2).

Affinity

As a particular example of composition relations, we define a specific *affinity* relation as follows:

$$a \overset{\therefore \quad \cdot}{\rightarrowtail} d \overset{\cdot \quad \therefore}{\leftarrowtail} b$$

a and b are said to be *affine* together and d is the *descriptor* object describing (or "measuring") their affinity. This situation is aliased to:

$$a \overset{\therefore \quad \cdot}{\rightarrowtail} \widetilde{ab} \overset{\cdot \quad \therefore}{\leftarrowtail} b$$

On the specific level l_3, we will also use the following aliases:

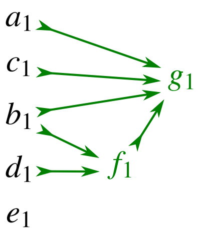
$$a_1 \rightarrowtail \widetilde{a_1 b_1} \leftarrowtail b_1$$

$$\begin{tikzcd}
a_1 \arrow[r, squiggly, no head] & b_1
\end{tikzcd}$$

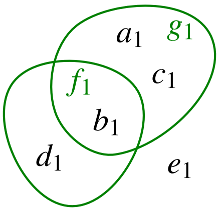
We will make use of affinity relations in the next sections.

Bubble representation

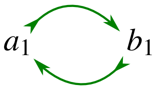
For the sake of readability, we will often use bubble representations of composition relations. For instance, this situation on specific level l_3:



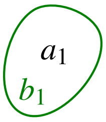
can be represented this way on the same level:



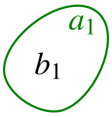
We are aware that this representation is not straightforward, since the following situation:



needs two bubbles representations:



and



The user of the model might be interested in this opportunity. However, in the next, we will not have to deal with such circular cases. As a consequence, bubble representation will be both useful and non-ambiguous.

Influence

In M1, objects may also *influence* each other:

$$a \longrightarrow b$$

This implies that their state change over time, which constitutes their basic *dynamics*. When a influences b, b dynamics depend on the state of a. Put it another way, a exerts a certain *activity* on b.

Target of the influence might either be:

- an object, like in $a \longrightarrow b$
- a composition relation, like in $$\begin{tikzcd} & c \\ a \arrow[r] & {} \\ & b \arrow[uu] \end{tikzcd}$$

Influence on objects comes into three different flavours on level l_2:

1. $a \xrightarrow{\backsim} b$: objects a may *alter* the state of objects b
2. $a \xrightarrow{-} b$: objects a may *delete* objects b
3. $a \xrightarrow{+} b$: objects a may *create* objects b

Influence on composition relations comes into two different flavours on level l_2:

1. $$\begin{tikzcd} & c \\ a \arrow[r, "-"] & {} \\ & b \arrow[uu] \end{tikzcd}$$: objects a may *delete* composition relations
2. $$\begin{tikzcd} & c \\ a \arrow[r, "+"] & {} \\ & b \arrow[uu] \end{tikzcd}$$: objects a may *create* composition relations

The alias $a \xrightarrow{\widetilde{cd}} b$ is used when the target of the influence is also influenced by the affinity between objects c and d. In other words, a activity depends on $\widetilde{cd}$.

Properties

Here are a few basic, intuitive properties on these relations which deserve to be made explicit. They all are illustrated with one example on specific level l_3, where instance objects a constitute instance objects b: $\begin{tikzcd} a \arrow[r, tail] & b \end{tikzcd}$

If a is under alteration influence, then so it b. Their dynamics both depend on this influence. In other words, alteration influence propagates through composition:

$$\begin{tikzcd} \underset{\sim}{\overset{\curvearrowright\,\sim}{a_1}} \arrow[r, tail] & \underset{\sim}{b_1} \end{tikzcd}$$

On the other hand, if b is under alteration influence, a is not necessarily affected. Alteration influence does not propagate downwards:

$$\begin{tikzcd} a_1 \arrow[r, tail] & \underset{\sim}{\overset{\sim\,\curvearrowleft}{b_1}} \end{tikzcd}$$

If a is deleted, then so are the composition relations it was being involved in. And so is b if it was exclusively made of one a. Deletion propagates upwards:

$$\begin{tikzcd} \underset{-}{\overset{\curvearrowright\,-}{a_1}} \arrow[r, tail, "\cdot", "-"'] & \underset{-}{b_1} \end{tikzcd}$$

However, if b is made of several a, it won't be deleted until the last constitutive a is deleted, which may defer deletion propagation:

$$\begin{tikzcd} \underset{-}{\overset{\curvearrowright\,-}{a_1}} \arrow[r, tail, "\therefore", "-"'] & b_1 & a_2 \arrow[l, tail, "\therefore"'] \end{tikzcd}$$

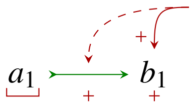
If b is deleted, then so are the composition relations it was being involved in. But this does not affect a. Deletion does not propagate downwards:

$$\begin{tikzcd} a_1 \arrow[r, tail, "-"'] & \underset{-}{\overset{-\,\curvearrowleft}{b_1}} \end{tikzcd}$$

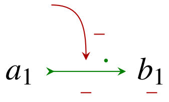
If a new a is created, it will not necessarily contribute to constitute a b:

$$\underset{+}{\overset{\curvearrowright\,+}{a_1}}$$

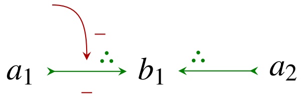
On the other hand, creating a new b implies that its relations to constitutive objects a are created along with it. Also, it obviously implies that such constitutive objects a already exist in the first place:



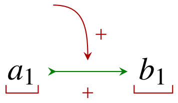
Deleting a composition relation entails the deletion of constituted object b if the latter only relied on this relation. Deletion propagates upwards:



However, once again, propagation may be defered if b still has other relations to rely on:



Finally, creating a composition relation only implies that both objects already exist:



These will enable dynamic statement of our pivot model M1 (section 6.3).

The language we are offering is similar in philosophy to the language used in category theory [61]. We have not used the latter because M1 needs particular concepts (like *inclusion*, *state* or *dynamics*) which would have been too much work elaborating using only categories. In addition, there could have been several ways to elaborate them, and we think that biotic support should not rely on any particular construction of such concepts. For these reasons, we have decided to state our pivot model with a dedicated language.

However, should the biotic features be formalized one day so there would be a need to formalize M1, we suggest that categories would be the right tool to use because they use very few preconceptions.

6.2 Structure

In this section, we shall state the whole structure of M1. That is: every object and their composition relations to each other. Each generic l_2 statement comes along with a specific l_3 example. They all fall into into the five statement categories.

Activity

There first needs to be a “world” for biotic dynamics to occur within it. Should it be a biochemical world, virtual data spread over a computer architecture, primitive biomaterials in an organic soup, lucrative companies competing in a stockmarket, arid rocky landscapes or lush exotic woodlands (see figure 3). We also call it *medium* or *environment*: E. It is made of various objects e which we have called “actual, incidental objects of the system” so far. Some of these objects are called *bodies* b.

The actual *meaning* of E, e, b is left free to the user of the model. Their actual *content* as well. They are not necessarily atomic, nor distinct from each other.

$$\begin{tikzcd} E & e \arrow[l, tail, dotted] \arrow[loop above, dotted] \arrow[r] & b \arrow[r, tail, dotted] & o \end{tikzcd}$$

Figure 2: Structural generic l_2 statement regarding Activity. E is the environment, e are its constitutive external objects. Bodies b are specific e objects. Bodies integrate oligoms o into the environment as they are their unique connection to E.

In order to animate the medium with a biotic behaviour, we shall have it “controlled” by internal, virtual objects. In order for M1 to be usable, there needs to be at least one basal, atomic object defined within M1. We call this internal atom *oligom* o (from ὀλίγος-μα, the *few objects*). Oligoms are the elemental, basic building blocks of M1. They are artificially decoupled from E.

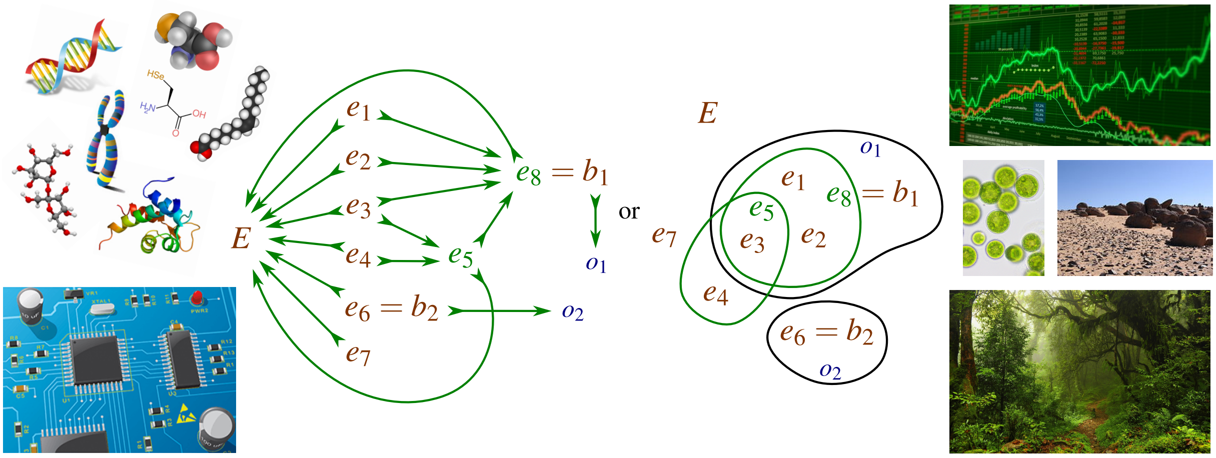


Figure 3: Specific l_3 example of Activity objects. Left: a formal view rigorously using the language defined in section 6.1. Right: an equivalent bubble representation. Some e objects like e_5 are composed of other subobjects e_3 and e_4. Some others like e_6 are bodies b. Bodies may constitute oligoms like o_1 and o_2, which are not directly part of the environment.

Oligoms are the only connection point between external spontaneous medium activity and internal spontaneous biotic activity. The only role of such virtual internal objects is to abstract biotic properties away from their actual medium. Carefully coupling the latter two together is a necessary step towards building M3 models and artificial life simulations (see figure 1 and our proposition section 8).

Holonics

In order for the atom o not to make the model rigid, we allow oligoms to aggregate together into loose *agents* a [38]. Agents are internal objects. They are constituted of oligoms connected together by a special kind of dynamic link, the *biotic link* ℓ.

$$\begin{tikzcd} b \arrow[r, tail, "\cdot \quad \cdot"] & o \arrow[r, tail, "\therefore \quad \therefore"] & \ell \arrow[r, tail, "\cdot \quad \therefore"] & a \end{tikzcd}$$

Figure 4: Structural generic l_2 statement regarding Holonics. Oligoms o and their bodies b have an exclusive composition relation to each other, which connects oligoms to the medium E. Oligoms are also connected to each other *via* biotic links ℓ so that they form agents a together.

Oligoms are abstract functionnal modules inspired from biological "organs", even though they are *not* "organs". Agents are abstract congregated objects inspired from biological "organisms" and "individuals" , although they are *not* "individuals" either.

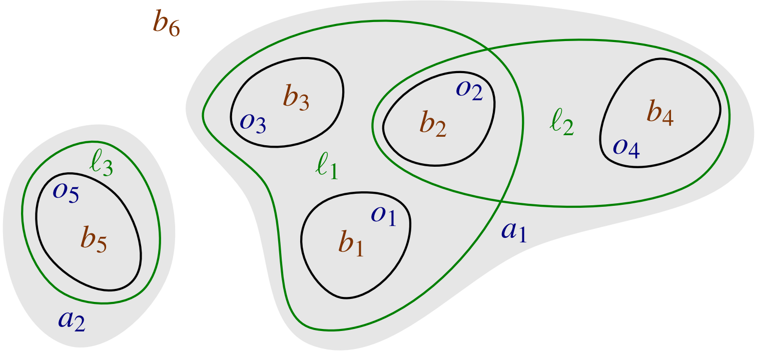


Figure 5: Specific l_3 example of Holonics objects. Black bubbles represent the exclusive relation between oligoms o and their constitutive bodies b. Green bubbles represent biotic links ℓ. Grey bubbles represent oligoms gathered together into a biotic agent a. Here, o_1, o_2 and o_3 are linked together by ℓ_1, o_2 and o_4 by ℓ_2. They all constitute agent a_1. Oligom o_5 is the only oligom in its agent a_2. b_6 has no connection to an oligom and is therefore only influenced by the environment E.

Agents are said to be *holonic* in the sense that one agent split into two becomes two agents, and that two agents merged into one are now one agent [39] (see Constraints section 6.4). This is an invariant biotic property which we think is essential to biotic dynamics. It brings a lot of flexibility to the model, even though we have settled a particular atom o (see Dynamics section 6.3).

Resource

With this category, we intend introducing *rationality* into M1 in the sense of Game Theory. We assume that there exist one or several kinds of *resources* r in the environment (*e.g.* light, fat, funds, mates, holes, heat, thrust, energy, minerals, disk space, credit, political support, CPU time), which are occasionally contained within bodies b. Oligoms access resources *via* dynamic abstract structures we call *pipes* p.

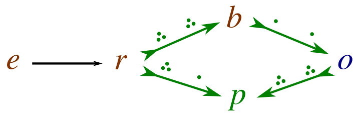


Figure 6: Structural generic l_2 statement regarding Resource. Resources r are specific external objects e constituting the bodies. They are drained by oligoms through pipes p. Each pipe only drains one resource.

One pipe p may be mounted over several oligoms, which introduces all resource *exchanges*: sharing, stealing, storing, spending, giving (see Dynamics section 6.3).

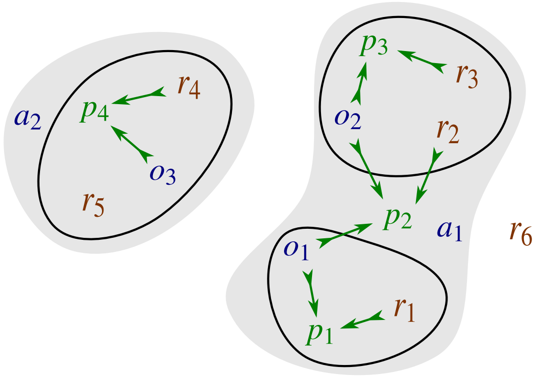


Figure 7: Specific l_3 example of Resource objects. Black bubbles represent oligoms and the resources r constituting their bodies. Grey bubbles represent agents a. Here, oligom o_1 accesses both its own resource r_1 and resource r_2 of oligom o_2. They share pipe p_2. Resources r_5 and r_6 are not drained by a pipe, so they are only influenced by the environment E (see section 6.4).

Encoding

With this category, we intend making it explicit the way Information is dealt with in the model. Since we have chosen oligoms as atomic objects, it seems quite natural to associate them with atomic pieces of information in which their properties would be encoded. We call the latter pieces of information *codes* c. Codes held by an oligom may help it building *generators* g able to create new oligoms.

Also, since we expect supporting information Transfer, we need to add another *copier* object κ to the model, enabling codes transmission.

$$\begin{tikzcd} g & c \arrow[l, tail, dotted, "\cdot\ \therefore"'] \arrow[r, tail, dotted, "\cdot\ \therefore"] & o & \kappa \arrow[l, tail, dotted, "\therefore\ \cdot"'] \end{tikzcd}$$

Figure 8: Structural generic l_2 statement regarding Encoding. Codes c may be expanded into generators g by oligoms, to create new oligoms. Each code is held within one oligom. Oligoms also contain copiers objects κ able to transmit information.

Since codes encode and create oligoms, their dynamics might ultimately be more essential than oligoms dynamics from an evolutionnary point of view [30, 46].

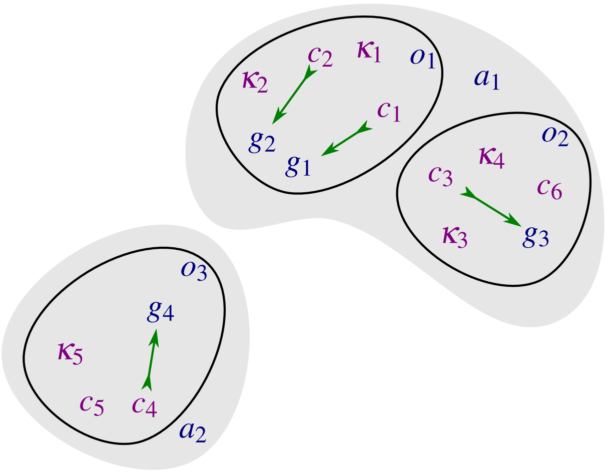


Figure 9: Specific l_3 example of Encoding objects. Black bubbles represent oligoms and the codes and copiers c and κ they contain. Grey bubbles represent agents. Oligom o_1 is currently reading codes c_1 and c_2, expanded into regulated generators g_1 and g_2. Copiers κ also constitute each oligom.

Alteration

Evolution is driven by changes in biotic information [29]. In order to trigger this phenomenon within M1, we need to add one last object to the model. This object is called *mutator*. Mutators live within the medium.

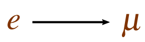


Figure 10: Structural generic l_2 statement regarding Alteration. Some external objects are mutators μ.

6.3 Dynamics

Now that the whole structure of M1 has been stated, we shall make its basic dynamics explicit. This will be done using influence relations defined in section 6.1. Each dynamic statement also falls into one of the five statement categories.

Activity

Environment E undergoes a spontaneous activity. This is modelled as a plain influence on its constitutive objects e, which implies that the latter may interact with each other. Oligoms also influence their bodies b. What these activities actually consist in is left free to the user of M1.

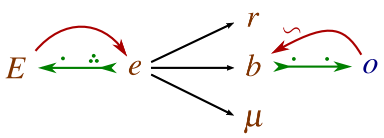


Figure 11: Dynamic generic l_2 statement regarding Activity. Environment E exerts an influence on itself *via* its constitutive objects e, b, r, μ. In addition, oligoms o may alter the state of their bodies b. One plain $\xrightarrow{\quad}$ is an alias for $\xrightarrow{+}$, $\xrightarrow{-}$ and $\xrightarrow{\backsim}$ altogether.

This enables a loop interaction between external and internal objects: the medium influences oligoms *via* $\utilde{b} \overset{\cdot \quad \cdot}{\rightarrowtail} \utilde{o}$, and oligoms influence the medium *via* $o \xrightarrow{\backsim} b$. Note that an oligom cannot directly create and delete e objects. However, this can still result from its body's activity: $\utilde{b} \rightarrowtail \utilde{E}$, $E \longrightarrow e$

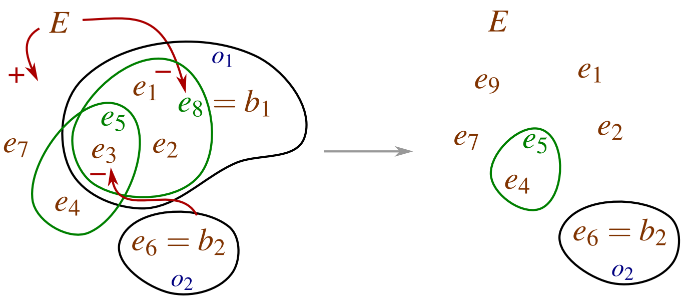


Figure 12: Specific l_3 example of Activity dynamics: e_8 is deleted under environmental influence, which entails deletion of o_1 due to upwards deletion propagation. e_3 is also deleted due to indirect b_2 activity, which reduces e_5 composition structure. Also, e_9 is created as a result of environmental activity.

In the context of Robotics and control functions, one may consider $\underset{\backsim}{b} \overset{\cdot\quad\cdot}{\rightarrowtail} \underset{\backsim}{o}$ as oligoms' basic sensor *inputs*, and $o \xrightarrow{\backsim} b$ as their basic control *outputs*. This is the reason why we refer to them as "functionnal modules".

Holonics

Agents a exert a top-down influence on their oligoms. Put it another way, the activity of one oligom o depends on the state of all other oligoms it is biotically linked to. This is what we call *integration* of the the agent.

Also, biotic links are not fixed: they can be created and broken. This introduces agents metamorphic dynamics.

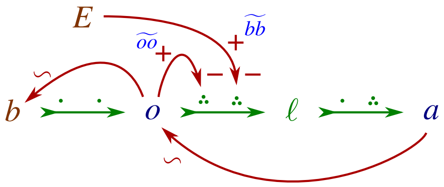


Figure 13: Dynamic generic l_2 statement regarding Holonics. Agents a exert a top-down influence on oligoms o. This integrates activity of all bodies b which are under the influence of the same agent. Oligoms connections to links ℓ can be unmade under external E or internal o influence. Also, they can be created by oligoms depending on affinity between them $\widetilde{oo}$, or by the environment depending on affinity between external bodies $\widetilde{bb}$.

This gives a first meaning to the biotic link ℓ: external *inputs* are shared through biotic links among oligoms in an agent. This can make oligoms' *output* activities consistent with each other. In this sense, one agent a can be seen as a metamorphic, modular "control function".

Many influence relations will be drawn in this section. However, two of them are fundamental. $E \longrightarrow e$ represents the whole spontaneous external activity of the medium. It is left free to the user of the model, and choosing it is closely related to choosing a medium. $a \xrightarrow{\backsim} o$ represents the whole spontaneous internal activity of biotic objects in the sense that any other internal activity relies on this one. Connecting both spontaneous activities together by defining $o \xrightarrow{\backsim} b$ is the most central part of the coupling process (see figure 1 and section 8).

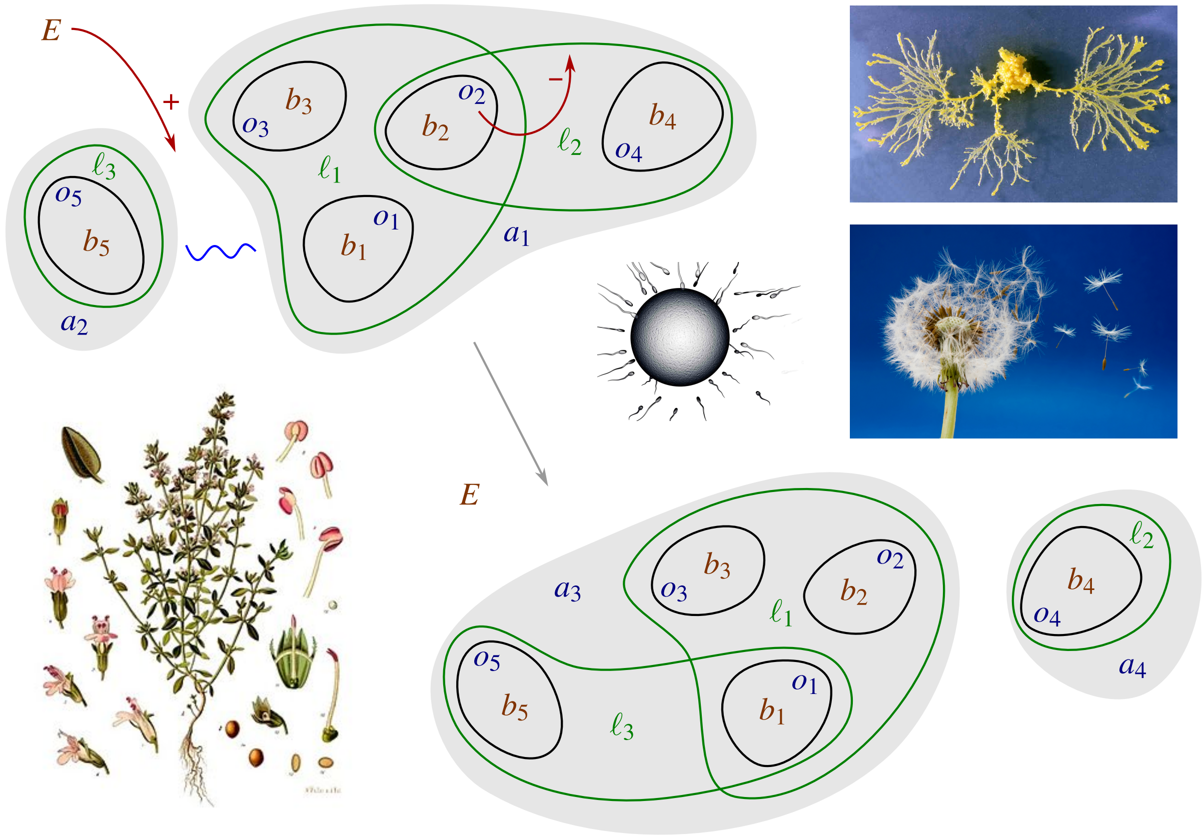


Figure 14: Specific l_3 example of Holonics dynamics. Oligom o_2 leaves link ℓ_2 as a result of its own activity. This isolates o_4 appart from the initial agent a_1. Depending on external influence and constitutive affinity between bodies $b_1 \sim\!\!\sim\!\!\sim b_5$, oligom o_1 joins ℓ_3, which links it to o_5. The agents configuration is changed: a_3 is composed of o_1, o_2, o_3 and o_5; a_4 only contains o_4.

These dynamic statements provide flexibility to M1. For instance, the situation offered figure 14 may represent many biological phenomena hard to express with rigid preconceptions on the meaning of "individuals". Agents a_1, a_2, a_3, a_4 may represent dynamic, incidental pieces of myxogastria plasmodia [87]. Or agent a_1 may represent a thyme plant (o_1 roots, o_3 leaves, o_2 flowers) releasing o_4 as a pollen gametophyte, or seed embryos. o_5 may represent a virus, or an inosculating adjacent plant, or mycorrhizae. Or the fusion may represent a fertilisation process.

Resource

Resource r is spent by the oligoms *via* pipes p, so pipes obviously exert an influence on resources. Also, pipes can be opened and closed under internal or external influence, which makes the whole resource network organisation dynamic.

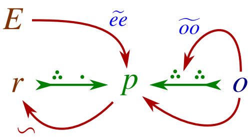


Figure 15: Dynamic generic l_2 statement regarding Resource. Pipes p drain resource r. Oligoms o may open, close and alter pipes. This can also be done under external influence depending on affinity between objects $\widetilde{ee}$. Oligoms may also change their access relations to pipes, depending on affinity between them $\widetilde{oo}$.

As a result, resource may flow from oligoms to oligoms even though it is not initially contained within the same bodies. This gives a second meaning to the biotic link ℓ: resource may transit through it.

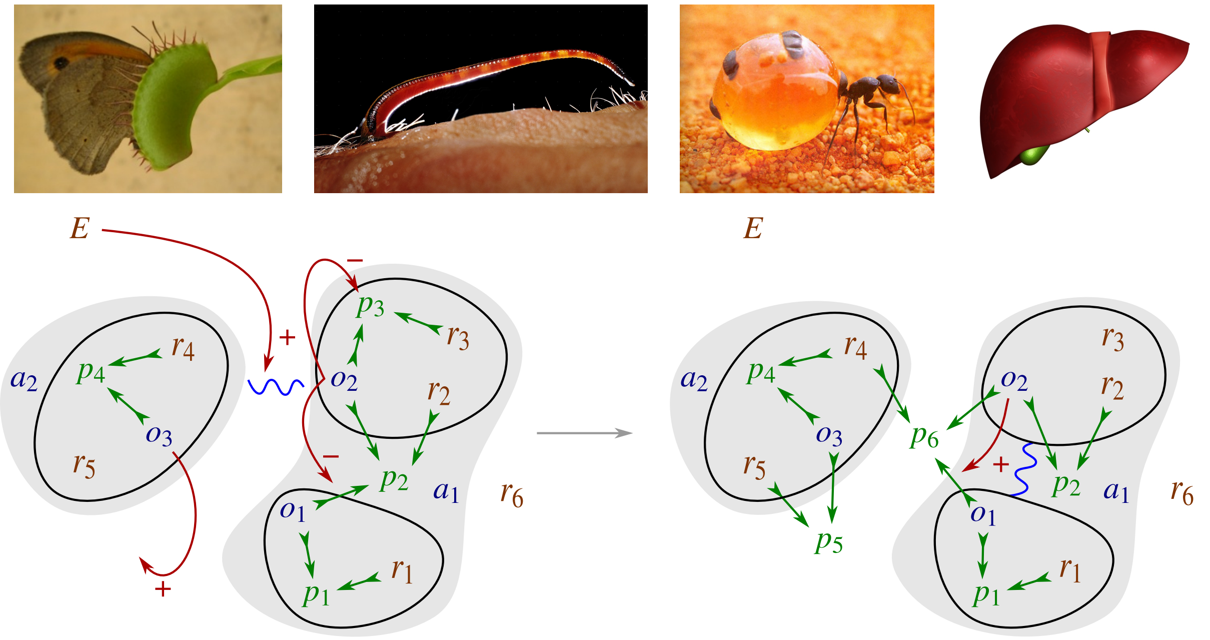


Figure 16: Specific l_3 example of Resource dynamics. Depending on external influence and on the affinity between resource r_4 and o_2's body, a new pipe p_6 is opened between r_4 and o_2. Also, o_3 opens pipe p_5 to the resource r_5 contained it its body. As a result of o_2's activity, pipe p_3 is closed and access to p_2 is denied to o_1. However, depending on $o_1 \sim\!\sim o_2$, o_2 may still provide o_1 with an access to its new pipe p_6.

We expect this resource exchange infrastructure to enable all basic activities underlying selfish dynamics and other phenomena expressed in terms of the Game Theory. For instance, in the situation offered figure 16, agent a_1 may represent a predator feeding on a_2, so p_6 would symbolize a digestive process. Or a_1 could be a parasite draining r_4. Or r_4 could have been stored in this very purpose, like if a_2 were a honeypot ant feeding its sister a_1, so that p_6 would symbolize trophallaxis. Here is a more tricky situation: imagine that o_2 is bound to provide o_1 access to resource, just like a mammalian liver (o_2) having evolved towards such an altruistic behaviour towards, say muscles (o_1). o_2 does this *via* p_2. Now, if there were a danger of biotic links to agent a_1 being broken any time, that is o_1 and o_2 suddenly becoming independent (see Holonics), then o_2 should better feed o_1 with anything else than its own resource when possible, in order not to lack resource after the link is broken. This rational behaviour may evolve and motivate such a spontaneous closure of $p_2 \longleftarrowtail o_1$ and opening of $p_6 \longleftarrowtail o_1$ due to rational o_2 activity.

Encoding

Encoding dynamics are slightly tangled. We shall consider two parts of it:

Growth activity : oligoms may *read* the codes c they are enclosing. *Reading* a code is to expand it into a generator g whose activity on the oligom is reciprocal: generators influence their oligom while oligoms regulate their generators. Generators are the only objects able to create new oligoms.

Transfer activity : oligoms contain copiers κ which may transfer information from one oligom to another. This consists in one code c being duplicated by a copier, then in its composition relation $c \overset{\cdot \quad \therefore}{\rightarrowtail} o$ to be set with another oligom. Transfer may also be performed under external influence.

All these activities heavily depend on affinity relations among internal objects $\sim\!\sim\!\sim$.

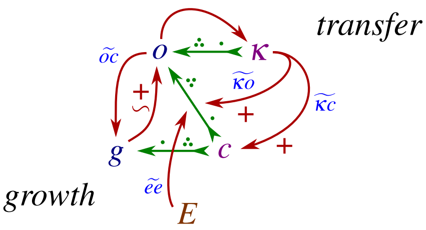


Figure 17: Dynamic generic l_2 statement regarding Encoding. Depending on their affinity with codes $\widetilde{oc}$, oligoms o may expand the latter into new generators g, exert an alteration influence on them or delete them. Generators influence in turn oligoms' activity and eventually create new oligoms. Oligoms also create and regulate copiers κ. Copiers κ create new codes c from the existing ones depending on their affinity with them: $\widetilde{\kappa c}$. They also set their new relations to other oligoms depending on their affinity with them: $\widetilde{\kappa o}$. This can also be done under external influence depending on affinity between external objects $\widetilde{ee}$.

As a result, biotic information may transit from oligoms to oligoms regardless of what it is actually encoding. This gives the biotic link ℓ a third and last meaning: information may be transfered through it.

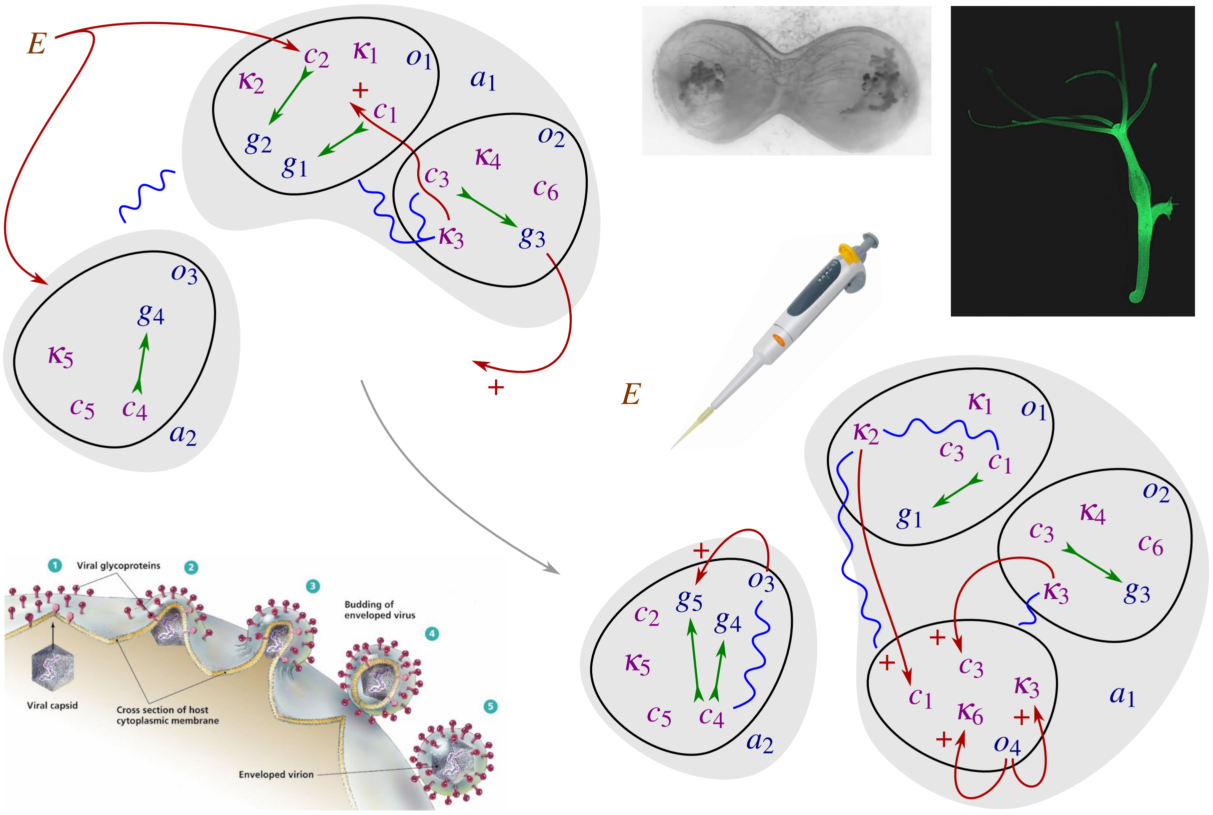


Figure 18: Specific l_3 example of Encoding dynamics. Due to external influence and affinity between bodies of o_1 and o_3, code c_2 is transfered from o_1 to o_3. As a result, generator g_2 collapses and o_1 is not submitted to its influence anymore. Copier κ_3, affine with both codes $\kappa_3 \sim\!\sim\!\sim c_3$ and oligoms $\kappa_3 \sim\!\sim\!\sim o_1$, copies c_3 to o_1 while generator g_3 is creating a new oligom o_4. Oligom o_3 reads c_4 again and expands it into a new generator g_5. o_4 immediately constitutes itself with copiers κ_6 and κ_3 which were encoded in c_3. Also, due to its affinity with κ_3 and κ_2, o_4 receives two new codes c_3 and c_1. Note that o_4's body b_4 (not represented) must have been somehow created in E prior to this event. This may — or not — have occured under indirect influence of g_3: $g_3 \xrightarrow{\sim} o_2$, $o_2 \xrightarrow{\sim} b_2$, $\underset{\sim}{b_2} \rightarrowtail \underset{\sim}{E}$, $E \xrightarrow{+} b_4$.

These statements introduce another fundamental biotic property: life builds from itself. For instance, on figure 18, oligom o_4 has been produced from another oligom o_2, like a hydra budding from another one or a plain cell mitosis. This is the most important phenomenon underlying Development of organisms into one another and overall life expansion, which we have refered to as *growth* in this category. Considering *transfer* and the spreading of biotic information, transmission of codes c_3 and c_1 to newly created o_4 may represent vertical genetic heredity, whereas horizontal transmission of c_2 from o_1 to o_3 may result from external GMO manipulations or bacterial conjugation. Thought of another way, c_3 has been able to spread from o_2 to o_1, and to get another oligom o_4 created by o_2, along with a copy of itself and of its copier κ_3. This is typical of viral activities.

Alteration

As a last dynamic statement, mutators μ exert an influence on codes and on their relations to the oligoms.

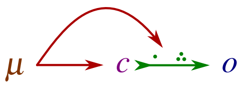


Figure 19: Dynamic generic l_2 statement regarding Alteration. Codes are created, deleted and altered under influence of external mutators μ. Mutators also change their relation to enclosing oligoms.

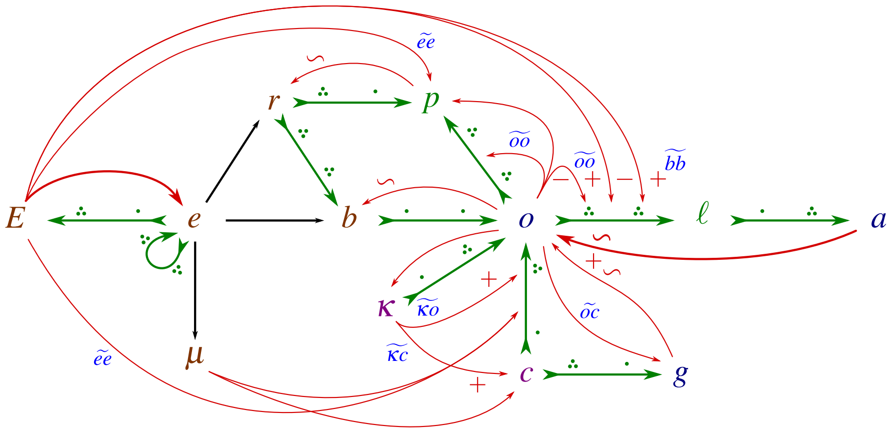


Figure 20: Whole structural and dynamic statement of the pivot model M1: basic objects and relations, basic activities. Thick influence lines stress the two most fundamental activities in the model: spontaneous medium dynamics and spontaneous biotic dynamics. Coupling them together with $o \xrightarrow{\backsim} b$ is also the crux of elaborating actual M3 artificial life simulations. Constraints are not represented.

6.4 Constraints

Now that the overall structure and dynamics of the pivot model have been set, M1 may already represent a variety of situations, some of which might not have any biotic relevance (see figure 20). In this section, we shall state a few basic, intuitive constraints on the model. Their purpose is to narrow its scope down to more consistent situations, without dropping our concern for biotic *weak-completeness*.

Most constraints only aim to gift M1 with a basic *locality* principle: objects interact together provided they are somehow connected to each other.

Every specific l_3 example given above does respect these constraints.

Activity

$\boxed{1}$ *via* $o \xrightarrow{\quad\backsim} b$, an oligom o may only influence its constitutive body b

This gives a meaning to the oligom: it is tied to one part of the environment with a *locality* principle and cannot exert any direct influence outside this “body”.

Holonics

The following two constraints aim to settle the meaning of the biotic link ℓ and of the agents a. They constitute what we call *integration* of the agent:

$\boxed{2}$ every oligom o must be constitutive of exactly one agent a.

This implies that no link ℓ can be not constituting an agent, and that no oligom o can be not constituting a link ℓ (should it be alone in that link, see o_5 figure 5). In addition, starting from one o and following up the compositions relations, it is not possible to end up into two different agents. There is no agent overlap.

Put it another way, agents are *isolated* from each other. No biotic link can be made between two agents or they must become one agent.

$\boxed{3}$ there must exist a path of biotic links ℓ between any two oligoms o in one agent a

That is, if you follow up and down the composition relations $o \rightarrowtail\overset{\therefore\quad\therefore}{\longrightarrow} \ell$, you can get from any oligom in one agent a to any other oligom in the same agent. There is no independence between subagents.

Put it another way, agents are *connected* by biotic links. No agent can be split into two or it must become two agents.

These two constraints introduce *individuality* in M1 and give a meaning to the biotic link. However, the link organisation is still dynamic, so the agents definition must somehow adapt to any link creation or deletion not to break the above two rules (see figure 14). This is also the reason why agents cannot be refered to as neat “individuals”: they are dynamic structures.

Here is another *locality* constraint:

$\boxed{4}$ $$\begin{tikzcd} & & \ell \\ \textit{via}\ \ o \arrow[r, "\widetilde{oo}"] & {} & \\ & & o \arrow[uu] \end{tikzcd}$$, oligoms may only create and delete relations within their agent a

Resource

Here are constraints supposed to bring an intuitive meaning to resource objects r:

$\boxed{5}$ $b \longrightarrow E$ activity has a *cost* in resource r

$\boxed{6}$ $b \longrightarrow E$ cannot occur if its cost has not been *spent* from r

These just set what "resource" is needed for. The next constraints are only supposed to feed the *locality* principle:

$\boxed{7}$ oligoms o may only spend resource *drained* by the pipes p they constitute: $o \xrightarrow{\therefore\quad\therefore} p$

$\boxed{8}$ pipes p may only drain resource r constituting them: $p \xleftarrow{\cdot\quad\therefore} r$

$\boxed{9}$ $o \longrightarrow p$ may only influence pipes to resource contained in the active oligom o

$\boxed{10}$ $$\begin{tikzcd} & & p \\ o \arrow[r] & {} & \\ & & o \arrow[uu] \end{tikzcd}$$ may only influence relations to pipes involving the active oligom

These introduces *selfishness*. An oligom "owns" its resource and decides who might access it or not.

$\boxed{11}$ $$\begin{tikzcd} & & p \\ o \arrow[r] & {} & \\ & & o \arrow[uu] \end{tikzcd}$$ may only occur if both oligoms are constituting the same agent a

This integrates selfishness to the agent scale. An agent "owns" and controls resource access among its oligoms.

Encoding

Growth

Here are trivial *locality* constraints on codes:

$\boxed{12}$ *via* $o \xrightarrow{\quad} g$, an oligom may only influence generators g for of codes c it contains:

$$o \xleftarrow{\therefore\ \ \cdot} c \ , \ c \xrightarrow{\therefore\ \ \cdot} g$$

$\boxed{13}$ *via* $g \xrightarrow{\backsim} o$, a generator g may only influence the oligom o containing its code c

Put it another way, an oligom o may only read, expand and regulate the codes c it is constituted of. And codes c may only influence (*via* generators g) the oligom o they constitute.

$\boxed{14}$ new oligoms by $g \xrightarrow{+} o$ are biotically linked to the oligom containing g's code

Should it not last. This integrates *growth* to the agent scale. It also gives the biotic link ℓ an explicit way of being set (see figure 18). Agents creating oligoms are expanding themselves. Links need to be broken for new agents to be created (see figure 14).

Transfer

$\boxed{15}$ new codes by $\kappa \xrightarrow{+} c$ are *copies* of existing codes, containing the same information

This is the base for *heredity*. We think it is another a fundamental invariant biotic property because it enables *self-replication*. Here are other few *locality* constraints about it:

$\boxed{16}$ *via* $o \xrightarrow{\quad} \kappa$, an oligom may only influence copiers κ it contains: $o \xleftarrow{\therefore\ \ \cdot} \kappa$

$\boxed{17}$ *via* $\kappa \xrightarrow{+} c$, copiers κ may only copy codes c contained within their oligom

$\boxed{18}$ *via* $\kappa \xrightarrow{+}$ $\begin{tikzcd} o \\ c \arrow[u, tail] \end{tikzcd}$, copiers κ may only create relations to codes c in their oligom

$\boxed{19}$ *via* κ

$$\begin{tikzcd} & o \\ \kappa \arrow[r, "+"] & {} \\ & c \arrow[uu, <->] \end{tikzcd}$$

, copiers κ may only create relations to oligoms in the same agent a

This restrains information Transfer to the agent scale. Any transfer might still happen due to external activity:

$$\begin{tikzcd} & c \\ E \arrow[r, "\widetilde{ee}"] & {} \\ & o \arrow[uu, <->] \end{tikzcd}$$

.

Alteration

$\boxed{20}$ codes c only change due to mutator activity $\mu \xrightarrow{\backsim} c$

$\boxed{21}$ mutator activity $\mu \longrightarrow c$ is random

This introduces *blindness* of the source model: biotic dynamics exhibit no teleology. We think this is fundamental to Novelty support.

$\boxed{22}$ $\mu \longrightarrow c$ activity is weak compared to overall $E \longrightarrow e$ dynamics

This ensures stability of the evolutionnary system.

$\boxed{23}$ any state of c can be reached by mutational activity $\mu \longrightarrow c$

This thwarts $\boxed{22}$ in the sense that it provides maximum power to mutational dynamics on the long run, which also relates to Novelty support: Evolution may take you anywhere.

Parcimony

Now that these constraints have been set. Any user of M1 is free to translate it to any other modelling language to get an actual biotic procedure M2 (see figure 1).

Translation may make M2 more rich than M1 due to the inherent richness of the target language, mostly because it may be more explicit. But it may also make it more constrained due to inherent limitations of the target language. Depending on the use case of M2, one might be not concerned by these additional constraints. For instance, considering resources r as null objects and only a binary type of affinity descriptor $\widetilde{ab}$ (affine *vs.* not affine) may not be a problem if the only goal of M2 is to perform Adaptation by evolving one basic code c_1 in a particular situation (see one example section 8.1 or genetic algorithms [41]).

Nevertheless, we are concerned by *weak-completeness* of our life model. Accordingly, we shall try building an explicit M2 translation affixing as few additional constraints as possible on M1. If we do not, some of our targetted biotic features might not be supported, hence our concern for *weak-completeness* might not be met.

This concern for translation *parcimony* entails many interesting technical challenges we shall expose in the next section, along the with the solutions we have found.

7 Candidate translation

Once the pivot model M1 has been stated, we must interpret it using more explicit modelling languages in order to translate its internal objects and dynamics into actual, formal procedures M2 (see roadmap figure 1). This may be used for two purposes:

- Reinterpret existing biotic models in terms of M1 (like genetic algorithms, cellular automata, artificial ontogenies, see table 1). This will not introduce anything new to these models, but it is a necessary step towards checking validity of M1 as a generic frame.
- Elaborate new biotic models by translating M1.

In this section, we are reporting our work on the second purpose so far, and we describe our first translation project: M2$_c$. In order to meet our concern for *weak-completeness* (see section 3), we have tried to affix as few constraints as possible on M2$_c$ compared to M1. This raises interesting technical challenges which we shall address with the various *modules* listed hereafter. In order to meet our concern for abstraction, we are building M2$_c$ regardless of the actual, external medium it will animate. Choosing the medium and coupling it to M2$_c$ will be done later on in sections 8 and 9. Therefore, objects E, e, b, r and μ will remain abstract for now.

M2$_c$ aims to be able to animate various media with the same biotic dynamics. For instance, regardless of whether M3$_c$ will involves a cellular automaton, reconfiguration procedures of modular robots or genetic programs, the same M2$_c$ may be used to power them. In consequence, we shall refer to $\mathrm{M2}_c$ as an abstract simulation engine or *biotic engine*. We also call it Biose core procedure $\mathrm{M2}_{core} = \mathrm{M2}_c$.

- 7.1 *Generalities*: in this section, we will sketch the overal design of $\mathrm{M2}_c$ before breaking it down into several modules
- 7.2 *Resource module*: this module will explicit the way we intend modelling resource exchanges (r dynamics) with $\mathrm{M2}_c$. It is currently being prototyped.
- 7.3 *Affinity module*: this module will explicit the way we intend dealing with affinity relations $\widetilde{ab}$ in $\mathrm{M2}_c$. It is presented along with a working prototype.
- 7.4 *Behavioural module*: this module will explicit the way we intend modelling agents behaviour $a \xrightarrow{\backsim} o$ in $\mathrm{M2}_c$. Much work on it has already been done by Karl Sims in 1994 [89], but it does not support the holonic structure of M1 yet. Our adaptation to Biose is currently being prototyped.
- 7.5 *Developmental module*: this module will explicit the way we intend supporting developmental dynamics and *growth* in $\mathrm{M2}_c$. Much work on it has already been done by Karl Sims in 1994 [89], but it does not support the holonic structure of M1 yet. Our adaptation to Biose is currently being designed.

7.1 Generalities

Here, we shall state the overall approach of our translation project $\mathrm{M2}_c$: goals, theoretical tools and implementation tools. This section can only be completed when all working modules will be put together.

Goal: *in silico* simulations

Biose core $\mathrm{M2}_c$ is a biotic simulation engine dedicated to power *in silico* artificial life simulations. It will have three use cases:

1. Start validation of the model. A necessary step towards *weak-completeness* is that (at least) many biological situations can be simulated using Biose. The core will help us powering such simulations. (see **Evaluation** section 9)

2. Be offered as basis for an evolution game, similar in philosophy to the one published by Maxis in 1992 [60]. The game interest would reside in the fact that the biotic engine will

support 11 of the previously described 12 biotic features (see table 1). We expect benefiting from the computer games ongoing industry: they provide resource, development opportunities and they already gather enthusastic communities.

3. Be offered as a ready-to-use translation of M1 for Biological research. We expect that specific eco-evolutionnary hypotheses may be checked using $\mathrm{M}2_c$ simulations of various specific biological situations.

For now, our main scientific concern is the validation of the model: having it checked for both biotic and biological relevance. For these reasons, we will first focus on an $\mathrm{M}2_c$ core only dedicated to power *in silico* simulations.

Theoretical tools: set theory

Translation of M1 into a computer program requires rich conceptual toolsets. We are using tools and objects from *set theory* and classic mathematics, because they are consensual, powerful, versatile, and their implementation into computer programs quite well mastered. As a consequence, every "object" declared in the next may be refered to as an *element* in the sense of set theory.

Time

Influence relations stated in M1 require that the model undergoes certain *dynamics*. We choose to model time as a classic, endless continuous arrow $t \in \mathbb{R}^+$. Dynamics are translated into every object's state being a function of time.

Continuous time will obviously be discretized during implementation. In this regard, we shall consider adapting DEVS simulation formalism of complex dynamic systems to Biose [102].

Internal objects

Every generic internal object of M1 like o, a, ℓ, c, *etc.* will be modelled as a *set*. And their instance objects like o_1, c_3 *etc.* will be modelled as *elements* of these sets. Unless otherwise specified, composition relations between objects will be translated as plain set inclusion.

For instance, $c_1 \rightarrowtail o_1$ will be translated as $c_1 \subset o_1$.

Biotic links

Agents a are translated to hypergraphs. Their nodes are oligoms o and their edges are biotic links ℓ. Constraints $\boxed{2}$ and $\boxed{3}$ about their *integration* are easily translated to graph theory language: agents must be *connected* and *isolated*.

As suggested by M1, biotic links ℓ have several meanings because they underlie several phenomena:

inputs : oligoms in an agent are sensitive each other states: in particular to $\underset{\sim}{b} \rightarrowtail{\overset{\cdot\ \ \cdot}{\longrightarrow}} \underset{\sim}{o}$

resource : resource exchanges may occur within agents through biotic links ℓ

information : code exchanges may occur within agents through biotic links ℓ

Accordingly, we define three different sets of links: behavioural biotic links ${}^{b}\ell$, resource biotic links ${}^{r}\ell$ and information biotic links ${}^{i}\ell$. They define three corresponding sets of agents ${}^{b}a$, ${}^{r}a$ and ${}^{i}a$. The latter three may be overlapping graphs.

In the early days of mitochondria symbiogenesis, for instance, it is believed that prokariotic endosymbionts were able to exchange products of respiration with their eukariotic host, but not genes [69]. This may be modelled as a set of two oligoms involved in the same resource agent graph ${}^{r}a$ but two different information agents graphs ${}^{i}a$.

Implementation tools:

In order to write M2$_c$ down to actual computer code, we are using conceptual tools and objects taken from object-oriented paradigm. They are consensual, powerful, and they make it convenient to break the project down into several consistent modules. UML language translation to M1 is also quite straightforward [4]. We plan on using it to sketch an overview of M2$_c$ once every such module has been prototyped.

Coupling

In our core translation M2$_c$, there exists several oligom *types* depending on the type of body b they are animating, on their control over it, and on their reaction to other internal objects. These oligom types are described by formal *classes* in the sense of class-based object-oriented programming. An oligom class is defined by:

- the description of one particular type of body object: b

  root, leaf, hand, paw, mouth, fin, shell, ..

- methods making $o \xrightarrow{\backsim} b$ explicit

  grow, bite, jump, freeze, loop, mate, frown, ..

- methods making $o \longrightarrow \bullet$ explicit, where $\bullet$ represent every other internal object

  break a link ℓ, open a pipe p, read a code c, create a copier κ, repress a generator g, ..

- the description of every internal oligom property o^p (members) feeding the latter methods

  mood, target, hunger, stress, age, confidence, audacity, ..

Put it another way, oligom types derive from an abstract oligom class, and methods describing $o \xrightarrow{\backsim} b$ are virtual methods. No other internal object in M2$_c$ will be described with unimplemented virtual methods (see modules). In the end, the coupling process for M2$_c$ is just the process of implementing a set of actual oligom types. We call this set an oligom *library*. Biose library will be built by the user of M2$_c$. As an instance, we shall build one in section 9 in order to evaluate the procedure.

Oligom properties

At any time, the internal state of an oligom is accurately described by the state of all its internal properties o^p, so that it can be modelled by a plain mixed vector of properties.

The value of each property o^p results from both its corresponding code property c^p value (*encoding*) and on the current environment of the oligom (*integration*). So it is affected by both a spontaneous value and incidental events of the simulation. Spontaneous values may be inherited from one oligom to another *via* codes and their copiers κ.

In addition, due to M1 constraint $\boxed{23}$, mutators μ can make the code properties c^p take any value during the simulation. As a consequence, every spontaneous configuration of oligom properties o^p may be explored and evolved during the simulation.

Modularity

In order to make it easy designing the core procedure, we split oligom properties o^p into two categories:

- some properties o^p relate to virtual methods and will be set by the user of M2 during the coupling process

bite frequency, growth investment, loop size, ..

- some properties o^p relate to the internal functionning of the core

target code c to read, properties of created copiers κ, strength of pipes p, ..

Properties of the latter category are the only ones we shall refer to while building M2$_c$. Put it another way: from the core procedure viewpoint, oligoms are nothing but a set of o^p properties which may be acted upon. This structure ensures that the various modules constituting M2$_c$ can be designed independently depending on the o^p subset they involve. In the next sections, we shall describe each of these modules.

The first M2$_c$ procedure we are offering is currently being prototyped in Python 3 language. Source code is available at

https://gite.lirmm.fr/iago-lito/Biose/

This is a work in progress.

7.2 Rationality : Resource tanks

In this module, we intend making it explicit the way *resource* is dealt with in M2$_c$. For now, its only global design is presented, since the module is currently being prototyped.

Resources r are translated into positive, continuous variables like $r_1 \in \mathbb{R}^+$ (*e.g.* energy, minerals). There might still exist a need to model quantic types of resources explicitly (*e.g.* nuts, eggs), but we are considering the latter as particular, fixed amounts of elemental continuous resource for now. In our translation, resources are wrapped into particular objects we call *tanks* t: $r \overset{\cdot \qquad \cdot}{\rightarrowtail} t \overset{\therefore \qquad \therefore}{\rightarrowtail} b \overset{\cdot \qquad \cdot}{\rightarrowtail} o$: oligoms contain tanks.

Each tank t is characterized by:

$C \ \in \mathbb{R}^+$, a total resource *capacity*

$r \ \in [0,\ C]$, the actual *resource* it contains

Some of its dynamical characteristics are not directly controlled by the oligom:

$\varepsilon \ \in \mathbb{R}^+$, an *entering* flux given by other tanks whose resource are piped to the oligom *via* p

$\nu \; \in \mathbb{R}^+$, a *creation* flux generated by the body: $b \xrightarrow{\backsim} \nu$

$\lambda \; \in \mathbb{R}^+$, a *loss* flux wasted by the body: $b \xrightarrow{\sim} \lambda$

These constitute a *basal*, unchosen “usable” flux: $\beta = \varepsilon + \nu - \lambda$.

If $\beta > 0$, then the following flux are directly controlled by the oligom as a result of $o \xrightarrow{\backsim} b$ activity:

$\sigma \; \in [0,\ \beta]$, an actual *spent* flux allocated to oligom’s body activity $b \longrightarrow E$

$\gamma \in [0,\ \beta - \sigma]$, a *giving* flux allocated to other tanks: $\varepsilon = \sum\limits_p \gamma$

The *remaining* flux $\rho = \beta - \sigma - \gamma$ is sinked to / drained from the tank: $\frac{\mathrm{d}r}{\mathrm{d}t} = \dot{r} = \rho$, with respect to $r \in [0,\ C]$ constraint.

In order to respect *locality* constraints (see section 6.4), oligoms may only choose flux allocation (σ, γ) for tanks contained in their bodies b. Also the sum $\varepsilon = \sum\limits_p \gamma$ only involves tanks the oligom is piped to *via* p connections.

In order to respect intuitive *conservation* constraints, there should be no resource leak or creation except for ν (*e.g.* autotrophy) and λ (*e.g.* dissipation). Also, the *cost* of $b \longrightarrow E$ must be fixed.

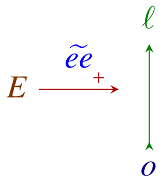
Basic tanks dynamics only formalize *altruistic* resource transfers. In order to allow any other type of transfer, we state that pipes p may also force-connect any tank loss flux λ to another tank creation flux ν. For instance, one oligom may be *stealing* resource from other tanks, or just scavenging from a leak. This way, we expect that all useful resource exchanges can be formalized in terms of M2$_c$ biotic procedure.

7.3 Arbitrary interactions: Promels

In this module, we intend making it explicit the way *affinity* relations $\widetilde{ab}$ are dealt with in M2$_c$. As a first product of this project, we have been able to formalize the problem using tools from standard mathematics, to find a candidate solution, to get a working prototype and to have it checked by a first set of simulations.

Problem Statement

M1 states that biotic dynamics depend on a certain “affinity” relation between objects, but it does not make it explicit what these relations actually are. However, there is still something we may derive from it. For instance, consider the following activity:



.. that is: oligoms may be biotically linked to each other due to external activity. This may be used to model gamete fusion, species hybridation, plants inosculation, endosymbiosis or trophallaxy. The idea is that, depending on affinity between external objects like their bodies: $\widetilde{bb} \subset \widetilde{ee}$, this new linkage may or may not be done. Put it another way, if two situations involving the same environment E and the same internal state o, ℓ, a, c *etc*. only differ by $\widetilde{bb}$, the linkage may occur in one but not in the other. In a nutshell, not *any* linkage is allowed.

However, in order to build a final product M3$_c$, the core procedure must be able to decide, any time, whether or not:

- two oligoms will be linked together with ℓ
- a particular pipe p will be opened between an oligom and a particular resource r
- an oligom will read a particular code c and expand it to g
- a copier κ will copy a particular code c
- an oligom will be targetted by a particular copier κ
- an oligom will be sensitive to a particular input in its agent a
- *etc*.

These will depend on a large network of objects affinity relations to each other: which codes c a copier κ is affine with, which copier κ an oligom o is affine with, *etc*. There must exist an explicit procedure to build this network.

In order to meet our concern for *weak-completeness*, we are willing to add as few constraints as possible on M1 with M2$_c$ (see section 6.4). Therefore, we wish that the latter network could spontaneously take any particular layout, so that every possibility is likely to be explored during the simulation. This is quite an intricate problem since the number of objects may be constantly changing, but the layout resolution cannot be totally random or no situation would be stable nor meaningfull. If, say, you decide to shake hands with your favourite houseplant, it is likely that you will not turn green, feel its crave for the sun, nor exchange a few of your genes with it. And this is not just because you are unlucky, it is because, to our knowledge, nothing biological enables such an intimate connection between you two at that point: there is no *affinity* between you two. This prevents our two species fusing together into a big mess. However, genes may still be transfered from one organism to the other if both are somehow affine with each other, or if the genes are transported by viruses affine with both of them. Even more puzzling: some highly unexpected new connections can be made between biological objects: see for instance human sperm-egg penetration which might have been taken from viruses [86], or the amazing *Elysia chlorotica* which has become able to breed chloroplasts from its algae meals and become a photosynthetic animal [17]. If a formal, deterministic layout resolution procedure is able to mimic such a particular configuration of relations between things, then how to handle the moment a totally new object will show up? How to define the relations it will have with the others without drawing them at random? How to have them selected so they have a meaning to Evolution?

Promels

Our solution comes in two steps: first, we consider that there are two types of affinities. The first, "incident" one, is an "affinity" deriving from plain *locality* principles: typically *distance*. Grizzlies and Polar Bears just cannot hybridate as long as they don't *meet* each other for instance [68]. The second, more "fundamental" one, is an underlying invariant property of the relation, which does not actually depend on distance. For instance: you will not create hybrids between lions and rhinos, no matter how close you rub them together. In this module, we shall only focus on this second type of affinity, since "incidental affinities" like physical "distance" may only be defined during coupling of $\textsc{m2}_c$ with a medium[1].

In a second time, we consider that "fundamental affinity" between two objects only results from the objects properties. In other words: if you know the objects, you know their affinity with each other. In this fashion:

- drawing the layout of relations between objects needs no random process, but a deterministic operation involving pairs of objects: this makes the simulation stable.
- this property may be derived from plain oligom properties o^p, encoded in codes c and evolved by mutators μ just like any other property: this makes the simulation open-ended.

This particular property of internal objects is inspired from the *shape* of biological objects. Shapes can match or not with each other. For instance, Yorkshire Terrier and English Mastiff

[1] Ultimately, the only difference between "fundamental" and "incidental" affinities is that the first one results from internal activity, whereas the second one is dictated by the medium. As a consequence, drawing the line between the two is only up to the user of $\textsc{m2}_c$: it just results from choosing a medium.

cannot mate for they obviously are a *non-match*. On the other hand, proteins, RNA, DNA 3D configuration and chemical properties can make them *match* with each other.

This module is all about defining such an artificial *shape* property of objects and such a *match* operation resulting in their affinity relation to each other. The main challenge is to add as few constraints as possible to M1 (stay valid for any affinity descriptor type, keep it possible to describe any layout of descriptors). The main technical challenge is that relations between objects $\mathcal{O}\left(n^2\right)$ must be somehow encoded into nothing more than the objects properties $\mathcal{O}\left(n\right)$.

The solution we have found, a *matching key*, resembles early days of Artificial Immune Systems [33]. It is described in detail within another research report [19] along with our prototyped candidate, the *promel*.

7.4 Arbitrary control functions: Proćens

In this module, we intend making it explicit the way agents behaviour $a \xrightarrow{\quad\backsim} o$ will be translated in M2$_c$. For now, its only global design is presented, since the module is currently being prototyped.

Problem Statement

M1 states that behaviour of agents a must be *integrated*. That is, activity of one oligom o may depend on the state of all other oligoms it is biotically linked with. In our translation, oligoms o and biotic links ${}^{r}\ell$ are constitutive modules of behavioural agents ${}^{r}a$. The latter can be seen as virtual, modular "robots", with every $\underset{\backsim}{b} \overset{\cdot\quad\cdot}{\rightarrowtail} \underset{\backsim}{o}$ constituting their environmental inputs, or *sensors*, and every $o \xrightarrow{\quad\backsim} b$ constituting their control outputs, or *actuators*. As a consequence, $a \xrightarrow{\quad\backsim} o$ can be seen as a *control function*.

The first problem is that, before coupling the core to a medium, we have no idea what the inputs may be (light, neighbours, CPU time, angles, fear, random winds), nor what the output activities may consist in (grow, bite, jump, freeze, loop, mate, frown). Put it another way, the control function signature is not typed. The second problem is that the agent configuration itself may change over time, due to their holonic properties (split into two, merge into one, grow new oligoms, loose oligoms). Put it another way, the control function signature is not fixed. The overall difficulty is that we are willing to affix as few constraints as possible on M1 with M2$_c$.

However, once again, activity of the agents cannot be random or it would be not meaningful to Evolution. So it must be somehow encoded in oligom properties o^p. How to encode a dynamic control function into oligom properties without adding unnecessary constraints on $\textsc{m}2_c$?

We are aware that these problems closely relate to cognitive science and artificial intelligence [20]: no one can explain and simulate arbitrary behaviour of *any* agent, since a "behaviour" may involve advanced integration processes, memory, clever associations *etc*. We do not claim being able to go beyond any such limit with this module. However, we can still build procedures entailing at least *rich*, *various* and *flexible* deterministic behaviours.

In a nutshell, this module aims to provide a formal procedure enabling dynamic building of interesting, metamorphic control functions.

Proćens

The solution we have found is inspired from Karl Sims animation procedure [95, 89], from electronic circuits and from neural networks [75]. It makes a heavy use of the *oligom property* notion o^p.

Inputs

No matter how oligoms input methods $\underset{\backsim}{b} \overset{\cdot\quad\cdot}{\rightarrowtail} \underset{\backsim}{o}$ will be implemented by the user during the coupling process, they will end up with a change in some oligom properties o^p. We call the latter properties *input properties*: $in \subset o^p$. The set of all input properties constitutes oligoms sensors list. They are the entry point into ${}^r a$ internal procedure.

Outputs

No matter how oligoms output methods $o \xrightarrow{\quad\backsim} b$ will be implemented by the user during the coupling process, they will feed from a set of oligom properties o^p. We call the latter properties *output properties*: $out \subset o^p$. The set of all output properties constitutes oligoms actuators list. They are the end of ${}^r a$ internal procedure. Note that in and out may overlap.

Mixed nature

in and out properties may be of various different mixed types, depending on the domains they live in:

	in		*out*
$\mathbb{N}$	(number of neighbours)		(need in nuts)
$[\![0,\ n]\!]$	(pixel color)		(mood)
$\mathbb{R}$	(target position)	.. *processing* ..	(self-confidence)
$\mathbb{R}^+$	(temperature)		(desired speed)
$[0,\ 2\pi[^2$	(gravity direction)		(desired direction)
$\vdots$	$\vdots$		$\vdots$

One first step towards making ${}^r a$ internal procedure general and flexible is to cast every *in* property into a vector of *visions* $\varphi \in [0,\ 1]$. Functions performing such casts will be chosen by the user during the coupling process. A library of such "casters" will be offered with M2. Parameters of such function are themselves oligom properties o^p.

Similarly, every *out* property must be read from a vector of *criteria* $\xi \in [0,\ 1]$ and casted to its own domain. Functions performing such casts will be chosen by the user or picked from M2's library. Their parameters also are oligom properties o^p.

As a result, and regardless of the choosen medium, animating a behavioural agent ${}^r a$ is nothing more than converting a vector of visions $in = (\varphi_1, \varphi_2, \dots) \in [0,\ 1]^{|in|}$ into a vector of criteria $out = (\xi_1, \xi_2, \dots) \in [0,\ 1]^{|out|}$. This can be done in many interesting ways [95, 89]. The procedure we are currently prototyping exhibits:

flexibility : $|in|$ and $|out|$ change over time: agents may be transformed without being undefined

non-linearity : *in* processing into *out* is complex, so agents behaviour is not trivial

memory : agents behaviour depend on past inputs *in* so they can *defer* reactions *out* and *learn*

evolvability : agents behaviour is deterministic, so it can be described by oligom properties o^p, encoded in codes c, inheritated *via* copiers κ and altered by mutators μ

The main idea is to reuse Karl Sims's processing « neuron nodes » [89], to restrict them to $[0,\ 1]$ domain, and to associate them with promels shapes (see module 7.3) in order to make the processing network both metamorphic and inheritable.

This is a work in progress.

7.5 Arbitrary forms: Development

In this module, we intend making it explicit the way codes reading $o \longrightarrow g$ is dealt with in M2$_c$. For now, its only global idea is presented, since the module is currently being designed.

Problem statement

As stated in M1, oligoms are created by generators from the codes c they contain. Also, new oligoms are biotically linked *via* ℓ to the oligom which has created them. In our translation, there are several *types* of oligoms, and accordingly several types of codes. Development implies that biotic agents a change continuously due to *growth* and holonic dynamics.

Since we do not want additional constraints to be put on M1 by M2$_c$, we wish that any form could spontaneously result from codes decoding. In other words, given any set of codes c (*i.e.* any set of oligom types) and any agent graph a_1 containing oligoms of corresponding types, we wish that any new agent graph a_4 could result from developmental activity.

By “developmental activity”, we mean:

- $g \xrightarrow{+} o$: oligom creation (like growth)
- $$\begin{tikzcd} & \ell \\ {} \arrow[r, "+"] & {} \\ & o \arrow[uu] \end{tikzcd}$$: link creation (like fusions)
- $$\begin{tikzcd} & \ell \\ {} \arrow[r, "-"] & {} \\ & o \arrow[uu] \end{tikzcd}$$: link deletion (like splitting)
- $$\begin{tikzcd} {} \arrow[d, bend left, "-"] & & \\ \underset{-}{b} \arrow[rr, tail, "\cdot \quad \cdot", "-"'] & & \underset{-}{o} \end{tikzcd}$$: oligom deletion (like apoptosis)

With deletions, it is straightforward that any agent may be turned into a trivial agent a_2 containing only one oligom. Therefore, the problem can be reduced to finding a way to get, from any trivial agent a_2 containing one oligom, any new agent a_4.

Oligoms may only be created recursively *via* $g \xrightarrow{+} o$, which implies that they must be linked to their parent oligom due to M1 constraint $\boxed{14}$. As a consequence, only *tree* agents may be grown from a_0, in the sense of graph theory (no cycles).

However, for any hypergraph a_4, there exist a tree graph a_3 from which a_4 may be derived by adding or removing links, or removing oligoms. Therefore, the problem can be reduced to finding a way to get, from any trivial agent a_2, any tree agent a_3 using only $g \xrightarrow{+} o$. This is what this module is about.

Formal grammars

Some relevant work on this has already been done by Karl Sims in 1994 [89], with an elegant solution using formal grammars [71, 82]. However, this solution does not support the holonic structure of our agents a since Sims's model is individual-centered (no split, no merge).

We are currently making use of promels shapes (see module 7.3) in order to adapt Sims developmental procedure to metamorphic individuals. Just like the affinity procedure, the procedure will be:

- *complete*: any form may result from it
- *integrated*: its dynamics will depend on the environment
- *evolvable*: it can be described by oligoms properties o^p, therefore encoded in codes c and evolved by mutators μ towards anything else

This is a work in progress.

8 Candidate Media

In this section, we shall describe various media and how we expect coupling our core procedure M2 to them (see roadmap figure 1). Although we are currently working on the first one, Biose is designed to make it possible developping any other coupling.

- 8.1 *2D rigid Physics*: This medium is our main focus for now. It involves a rich 2-dimensional, spatialized environment ruled by classical mechanics of rigid bodies.
- 8.2 *AlternateMedia*: Gather ideas on other types of media. These are open sections:
  - *Genetic Programs*: This medium inspired from Tierra [84] involves pieces of computer code aggregating together into programs and competing for machine resource.
  - *Cellular Automata*: Coupling M2 to cellular automata. They have been used for a while in biological modelling [25].

- *Modular Robotics*: Self-reconfigurable robots may be animated with Biose core procedure M2 [101].

8.1 2D rigid Physics

The main medium we are focusing on for now is a virtual environment undergoing basic 2D physics dynamics. They are both rich and easy to compute. They are powered by a generic physic engine called Chipmunk [63] (C library), interfaced with Python *via* Pymunk module [5].

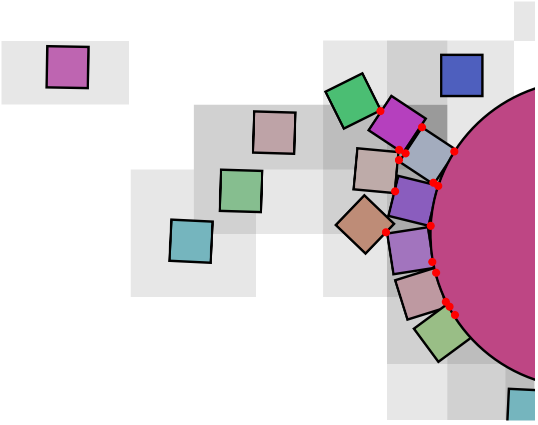


Figure 21: Chipmunk rigid shapes and collision detection [63]. Grey area represent spatial hash, red dots represent processed collision points. No gravity is set here, but forces (not drawn) are pulling squares towards the disk.

Biotic bodies

With this medium, *environment* E is mostly made of Chipmunk's physical "world" involving rigid bodies, forces, torques, joints *etc.* as e objects (see figure 21).

We are using Chipmunk rigid shapes as a basis for biose bodies b. They can be assembled together with various physical *joints* (*e.g.* pins, sliders, pivots, springs, in green figure 22) and controlled with precise forces and torques. Each such assembly may constitute a biose body b animated by a dedicated oligom type o.

Biotic links

With built-in *joints*, Chipmunk provides a natural way to embody the biotic link ℓ between oligoms. In this medium, and for the sake of *locality* principles, we will only link oligoms together when their bodies b are connected by physical joints. In this fashion, agents a will be

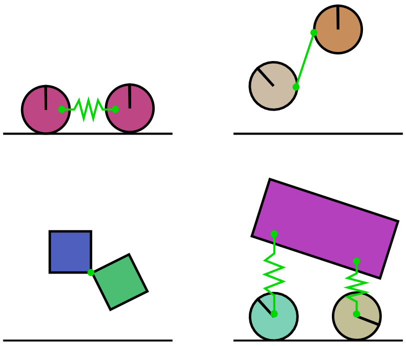


Figure 22: A few Chipmunk basic joints in a gravitic simulation: “springs”, “rope”, “pivot”. They may be used to elaborate sophisticated Biose bodies b animated by oligoms o.

embodied as consistent physical entities. Joints may also be set between bodies even if their oligoms are not linked together, just like barnacles attach to whales.

Simulations

With this medium, a M3 coupled life model just looks like a 2D rigid physic simulation. In order to visualize the output and dynamically interact with it, we are using SFML graphical C library interface [42] and its binding to Python *via* pySFML module [6]. Here are two small examples of such simulations.

Evolving a trivial agent

See figure 23 a small example of a Biose M3 model. It has been designed for an educational project to IUT students:

- E: plain, agravitic, damped Chipmunk “world”
- e_1: one landmark point (white cross)
- e_2: one path (dashed line)
- e_3: one rigid disk, $e_3 \rightarrowtail b_1$ (blue disk)
- e_4: one force, $\vec{F} = e_4 \rightarrowtail b_1$ (fainted white line)
- $e_5 = b_1$: one Biose *body*
- o_1: one trivial oligom, $b_1 \rightarrowtail o_1$
- ℓ_1: one trivial link, $o_1 \rightarrowtail \ell_1$
- a_1: one trivial agent, $\ell_1 \rightarrowtail a_1$
- c_1: one code encoding o_1 internal properties

Here, $o_1 \xrightarrow{\sim} b_1$ may only result in $\vec{F}$ being applied to one particular point of the disk e_3. The oligom o_1 has very few internal properties:

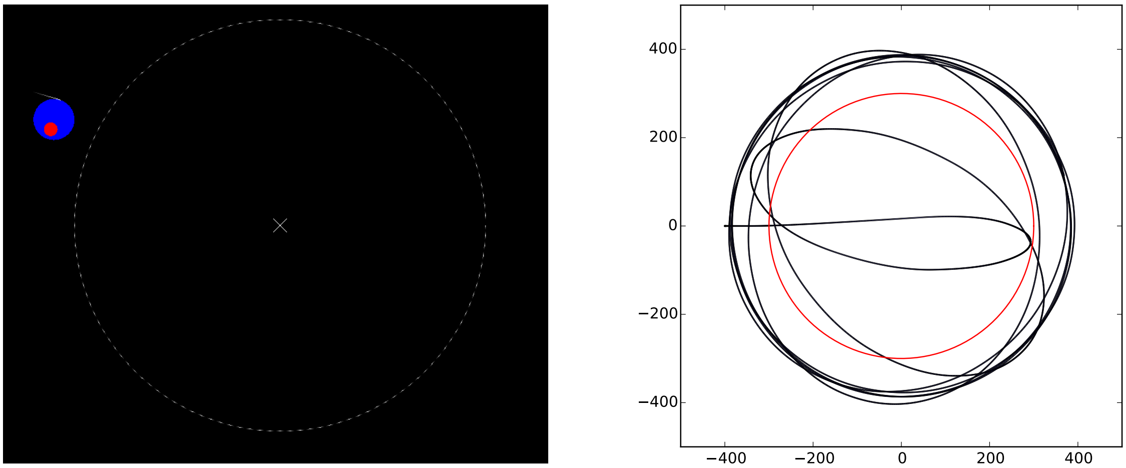


Figure 23: Basic simulation involving one trivial agent a_1. Left: snapshot of real-time visualisation interface. Right: overall trajectory of the disk e_3 since the beginning of the simulation. This has stood as a basis for IUT students project « Evolution game and Genetic Algorithms ».

Input properties $in \subset {o_1}^p$:

- $\delta \in \mathbb{R}^+$: shortest distance from the disk e_3 to the path e_2
- $\alpha \in [0,\ 2\pi[$: direction of the mark e_1

Output properties $out \subset {o_1}^p$:

- $\iota \in \mathbb{R}^+$: intensity of $\vec{F}$
- $\delta \in [0,\ 2\pi[$: direction of $\vec{F}$ relatively to α
- $\pi \in \mathbb{R}^+$: intermittence period of $\vec{F}$ application

Here, *out* properties are only influenced by c_1, so they are only spontaneous and do not depend on the environment. In other words, agent a_1 exhibits no sophisticated behaviour. However, it does adjust direction of $\vec{F}$ depending on the direction of the landmark. With a simple genetic algorithm, the students have been able to evolve c_1 so that new agents trajectory fit the given path e_2.

This trivial simulation only instanciates one particular biotic situation. It involves at least Environment and Adaptation features, like any genetic algorithm. However, they have been described within the frame of Biose pivot model M1.

In this simple simulation (see figure 24), trivial oligoms continuously split and merge into metamorphic agents:

- E: agravitic, non-damped, plain Chipmunk "world"
- $e_1 = b_1 \rightarrowtail o_1 \rightarrowtail \ell_1$: red, heavy disk
- $e_2 = b_2 \rightarrowtail o_2 \rightarrowtail \ell_2$: green, heavy disk
- $e_3 = b_3 \rightarrowtail o_3 \rightarrowtail \ell_3$: gold, light disk
- e_4: gravitational interaction between e_1 and e_2
- e_5: gravitational interaction between e_1 and e_3
- e_6: gravitational interaction between e_2 and e_3
- $e_{\dots}$: dynamical rope joints between disks

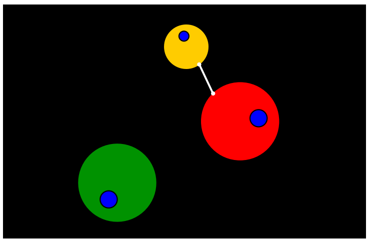


Figure 24: Simulation testing the metamorphic structure of the agents.

The trajectories followed by e_1, e_2, e_3 are highly non-linear, complex and chaotic since they are an instance of the planar three-body problem. During the simulation, disks alternately get twirling very close from each other then move away. Basic implementation of $a \overset{\backsim}{\longrightarrow} o$ entails that oligoms join the same biotic link ℓ when their disks collide, and leave it when they collide again provided the third one is closer than a certain threshold. Links are embodied by rope joints between the bodies (white line figure 24) so that their dynamics are constrained when two of them are connected together.

As a result, oligoms o_1, o_2 and o_3 continuously link and unlink, so that the agents a they form continuously merge and split appart. This simulation is a basic step towards supporting the holonic structure of Biose in this medium.

Further work on physical simulations and Chipmunk environment will be done once every abstract module presented in section 7 will have been prototyped, and once all these modules will have been associated together into one single M2 biotic procedure.

8.2 Alternate Media

In this section, will gather ideas about alternate types of M3 simulations involving coupling with other kinds of media. These are just open proposals for now since we are focusing on the 2D physics medium.

Genetic programs

Genetic programs are programs able to write other sub-programs, to run them and to evolve them with editions [59]. Considering such sub-programs as agents a and the chunks of computer code they are made of as oligoms o, one may develop an interesting medium involving evolving sub-programs interacting and competing against each other for computer resource r (CPU time, RAM usage, disk usage), similarly to Thomas Ray's Tierra simulations [84]. For instance, they may be evolved towards being efficient programs performing a particular task.

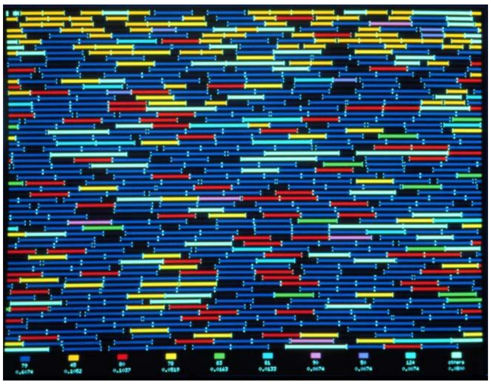


Figure 25: Visualisation of a RAM module during a Tierra simulation: colored lines are pieces of machine code competing together for CPU time.

This is an open section.

Cellular automata

Cellular automata are simple, discrete spatialized environments whose atoms are grid cells connected to each other by neighbourhood relations. They have been used for a while in biological modelling [25]. They are known to exhibit a variety of interesting behaviours often based on very simple rules, like the famous Game of Life offered by Conway in 1970 [27]. Considering cells and groups of cells as agents a and their properties as oligoms o, one may develop rich media gifted with locality principles, and M3 models producing interesting outputs.

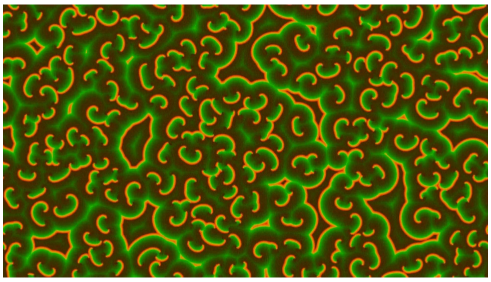


Figure 26: Cellular automaton undergoing a simple reaction-diffusion process.

This is an open section.

Modular Robotics

Modular, self-reconfigurable robots are robots built from basic modules able to connect and disconnect from each other. They can take various configurations, change configuration on their own, swarm, and exhibit integrated behaviours. Considering them as holonic agents a and their modules as oligoms o, one may use them as an actual medium and animate them with our biotic procedure M2.

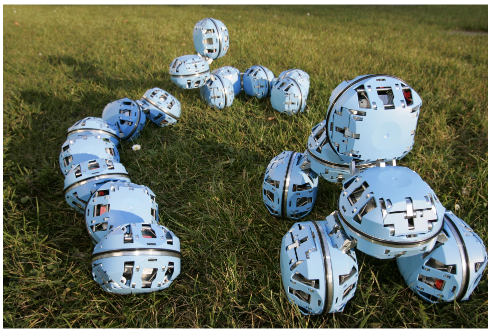


Figure 27: ATRON [55], an example of modular, self-reconfigurable robot

This is an open section.

9 Evaluation

Once we have been able to translate M1 to a formal core procedure M2$_c$ (see section 7) and we have coupled it to a medium (see section 8), we have finally got an actual M3$_c$ artifical biotic system. There is still a need to check that M3$_c$ exhibits a biotic behaviour, in the sense that it supports the biotic features described in section 2.2. In this section, we shall check ability of the model to reproduce various biological situations.

9.1 Strategy

Our two main scientific concerns about Biose are abstraction and *weak-completeness* (see section 3):

- A necessary check towards *weak-completeness* is that our M3$_c$ proposition can reproduce *at least* many biological situations, such as all situations listed section 9.2.
- A necessary check towards *abstraction* is that *at least* a few other M3$_c$ models, coupled to other media or derived from other M2 translations of the pivot model, can do the same thing.

These are just basic, necessary sanity checks for the project to be valid. Failing one of them will falsify the whole approach, but passing all of them will not prove it right. In order to prove the approach right, one would need being able to study directly the relations between Biose pivot model M1 and the biotic features (see figure 1), which is something we cannot do yet for the two reasons developped in section 5. As a consequence, and as a first step, we offer focusing on one M3$_c$ model for now and on the biological situations listed next section.

Still, passing all these tests will require significant work to be done, since we will need to:

- get a working prototype for each module presented in section 7
- assemble every module together into one core procedure M2
- complete the coupling of M2 with Chipmunk medium presented in section 8.1 to get M3$_c$
- build various M3$_c$ initial conditions entailing all processes listed next section

9.2 Tests

Here, we shall list every biological process we expect being able to reproduce with a M3$_c$ model:

- Any basic kind of Interaction between species and their evolutionary dynamics concerning Coevolution:
  - competition
  - facilitation
  - commensalism
  - symbiosis
  - parasitism
  - amensalism
  - predation
  - mutualism
  - *-ism, *etc.*

- Any tricky situation concerning Speciation:
  - ring species [53]
  - introgression [50]
  - allopatry [85]
  - hybridogen speciation [21]
  - sympatry [31]
  - *-patry, *etc.*
- Many known tricky systems involving life Scaling feature:
  - eusociality [78]
  - *Physarum polycephalum* [87] and slime molds in general
  - obligatory symbioses like lichens [52], pollination [40]
  - Pando almost-clonal forest [76]
  - tragedy of the commons and group selection [49, 91, 48, 83]
  - *etc.*
- Any way of Transfering biotic information:
  - heredity
  - sex [15]
  - viruses [22]
  - GMOs
  - conjugation [99]
  - *etc.*
- Developmental features of life: [18]
  - undefinite growth (like plants)
  - maturation (like puberty)
  - symbiotic development (like lichens)
  - monsters (like two-headed cattle)
  - homeosis [24]
  - definite growth (like animals)
  - vegetative reproduction (like plants and hydrae)
  - metamorphosis (like axolotl)
  - plasticity [81]
  - *etc.*
- misc — but not least — phenomena known as:
  - migration [79]
  - mimicry [88]
  - hitchiking [92]
  - complexification [70]
  - *etc.*
  - phenology [67]
  - senescence, death [37]
  - sexual selection [10]
  - punctuated equilibria [43]
  - drift [57]
  - evolvability [58]
  - exaptation [44]
  - immunity [54]

Obviously, there also exist on Earth many intricate and highly sophisticated systems and processes like:

- vision
- cognition
- language
- human societies

which would obviously be *very* difficult to reproduce with such a basal approach (although not absolutely *unthinkable* [93]). Biose is dedicated to represent very basic and general biotic processes, not heavily structured situations exhibiting very particular traits.

As a first test, we plan on modelling the system described by Susse Hansen and Paul Rainey in 2007 [48, 83], where bacteria in a biofilm, alternatively cooperating and cheating, form a self-replicating emergent structure behaving like an early multicellular organism. This biological situation involves many sensitive features of the living like Evolution, Interaction, Scaling, and may be modelled in various ways depending on whether we consider that each bacteria is an agent *a* or the draft they constitute. For these reasons, we think it would be a good first test candidate.

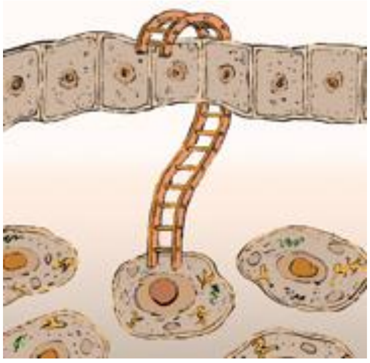


Figure 28: First candidate biological situation to reproduce: Paul B Rainey. "Unity from conflict" in: *Nature* 446.7136 (2007), pp. 616–616.

9.3 Results

This is a work in progress.

Conclusion

Fundamental questionning about *Life* has to deal with two different kinds of emergent phenomena. The first, *hard* kind of emergence, is the mysterious Emergence of early living, biotic objects from a non-living world, which is one of the most puzzling processes Biology will ever have to cope with. The second, *easier* kind of emergence, is the emergence of all known properties of the living and all known biotic processes — which we have refered to as the *invariants* or biotic features throughout this document — from these early mysterious objects. After having taken on not to address the first kind of emergence, we have been able to design a model of its early products, which we think it enables representing any aspect of the second one. Since we also have been willing not to rely on any particular biological system, we have abstracted this model away so that these second emergence phenomena could be translated into any other kind of systems, provided they can still match the basics of Biose.

Accordingly, the two main features of Biose we intend being able to offer are:

- Abstraction: Independence on the universe of simulation:
  - reusabilty
  - general reasoning
  - independence on incidental physics and biochemistry, computer rules or formalism
- *Weak-completeness*: Independence on usual biological assumptions like “two species cannot reproduce” or “individuals are basic selection units” or “genes have one way to replicate”:
  - easy modelling of systems close to the limits of such assumptions
  - integration of multiple scales
  - one descriptive language for any biotic situation
  - exploring alternative forms of life

The principal work we have to supply for Biose to be implemented into an actual, usable formal procedure, is to gather all existing pieces of solution together into one single program (formal grammars, sensor-actuator networks, game theory, graph node properties, *etc.*). This task is not completed yet, and we hope to find soon opportunities to go on with the roadmap sketched in sections 7 and 9. We would encourage in turn any other work aiming to formalize the mysterious relations between Biose *pivot model* M1 and the biotic features (see figure 1). We would encourage it because we trust that some real, major and central fundamentals of Biology are hiding there.

Bibliography

[1] URL: http://www.alife.org/ (visited on 12/07/2015).

[2] URL: http://agar.io/ (visited on 12/07/2015).

[3] URL: http://www.kongregate.com/games/mrdn/biocage (visited on 12/08/2015).

[4] URL: http://www.uml.org/ (visited on 02/26/2016).

[5] URL: http://www.pymunk.org/en/latest/ (visited on 03/08/2016).

[6] URL: http://www.python-sfml.org/ (visited on 03/08/2016).

[7] Tarek Ababsa et al. “Decentralized approach to evolve the structure of metamorphic robots”. In: *Artificial Life (ALIFE), 2013 IEEE Symposium on*. IEEE. 2013, pp. 74–81.

[8] Chris Adami, C Titus Brown, and W Kellogg. “Evolutionary learning in the 2D artificial life system ‘Avida’”. In: *Artificial life IV*. Vol. 1194. Cambridge, MA: MIT Press. 1994, pp. 377–381.

[9] C Athena Aktipis et al. “Cancer across the tree of life: cooperation and cheating in multicellularity”. In: *Phil. Trans. R. Soc. B* 370.1673 (2015), p. 20140219.

[10] Malte B Andersson. *Sexual selection*. Princeton University Press, 1994.

[11] Electronic Arts. “Spore”. In: *Electronic Arts* (2008).

[12] Robert Axelrod and William Donald Hamilton. “The evolution of cooperation”. In: *Science* 211.4489 (1981), pp. 1390–1396.

[13] Richard A Bartle. *Designing virtual worlds*. New Riders, 2004.

[14] FA Bazzaz. “The physiological ecology of plant succession”. In: *Annual review of Ecology and Systematics* (1979), pp. 351–371.

[15] Leo W Beukeboom and Nicolas Perrin. *The evolution of sex determination*. Oxford University Press, 2014.

[16] G Beurier. “Codage indirect de la forme dans les systemes multi-agents: émergence multi-niveaux, morphogénese et évolution”. In: *These de l’université de Montpellier* 2 (2007).

[17] Debashish Bhattacharya et al. “Genome analysis of Elysia chlorotica egg DNA provides no evidence for horizontal gene transfer into the germ line of this kleptoplastic mollusc”. In: *Molecular biology and evolution* 30.8 (2013), pp. 1843–1852.

[18] John Tyler Bonner et al. “The evolution of development.” In: *The evolution of development.* (1958).

[19] Iago Bonnici and Abdelkader Gouaich. *Promels: Formalize Spontaneous Emergence of Arbitrary Relations among Objects.* Research Report. LIRMM, Mar. 2016. URL: http://hal-lirmm.ccsd.cnrs.fr/lirmm-01294560.

[20] Rodney A Brooks. “Intelligence without representation”. In: *Artificial intelligence* 47.1 (1991), pp. 139–159.

[21] Luciano Bullini. “Speciation by hybridization in animals”. In: *Italian Journal of Zoology* 52.1-2 (1985), pp. 121–137.

[22] Carlos Canchaya et al. “Phage as agents of lateral gene transfer”. In: *Current opinion in microbiology* 6.4 (2003), pp. 417–424.

[23] Jenova Chen. *flOw*. 2011. URL: http://cognitivelabs.com/flow/fl0w_widescreen.swf (visited on 12/07/2015).

[24] Siddhartha Chib and Edward Greenberg. “Understanding the metropolis-hastings algorithm”. In: *The american statistician* 49.4 (1995), pp. 327–335.

[25] B Chopard and M Droz. *Cellular automata*. Springer, 1998.

[26] Fred E Cohen and Stanley B Prusiner. “Pathologic conformations of prion proteins”. In: *Annual review of biochemistry* 67.1 (1998), pp. 793–819.

[27] John Conway. “The game of life”. In: *Scientific American* 223.4 (1970), p. 4.

[28] Dominic Crapuchettes, Dmitry Knorre, and Sergey Machin. “Evolution”. In: (2014).

[29] Charles Darwin. “On the origins of species by means of natural selection”. In: *London: Murray* (1859), p. 247.

[30] Richard Dawkins. *The selfish gene*. 199. Oxford university press, 2006.

[31] Ulf Dieckmann and Michael Doebeli. “On the origin of species by sympatric speciation”. In: *Nature* 400.6742 (1999), pp. 354–357.

[32] Ulf Dieckmann and Régis Ferrière. “Adaptive dynamics and evolving biodiversity”. In: *Evolutionary conservation biology* (2004), pp. 188–224.

[33] Leah Edelstein and R Rosen. “Enzyme-substrate recognition”. In: *Journal of theoretical biology* 73.1 (1978), pp. 181–204.

[34] Bruce Edmonds. “What is Complexity?-The philosophy of complexity per se with application to some examples in evolution”. In: *The evolution of complexity* (1995).

[35] Bruce Edmonds. “Syntactic measures of complexity”. PhD thesis. University of Manchester, 1999.

[36] Elsevier, ed. *Ecological Economics* (1989-).

[37] D Fabian and Thomas Flatt. “The evolution of aging”. In: *Nature Education Knowledge* 3.9 (2011).

[38] Jacques Ferber. *Multi-agent systems: an introduction to distributed artificial intelligence*. Vol. 1. Addison-Wesley Reading, 1999.

[39] Christian Gerber, Jörg Siekmann, and Gero Vierke. “Holonic multi-agent systems”. In: (1999).

[40] Beverley Jane Glover. *Understanding flowers and flowering: an integrated approach*. Vol. 277. Oxford University Press Oxford, 2007.

[41] David E Goldberg. *Genetic algorithms*. Pearson Education India, 2006.

[42] L Gomila. *SFML–Simple and Fast Multimedia Library*. 2010.

[43] Stephen Jay Gould and Niles Eldredge. “Punctuated equilibria: the tempo and mode of evolution reconsidered”. In: *Paleobiology* (1977), pp. 115–151.

[44] Stephen Jay Gould and Elisabeth S Vrba. “Exaptation-a missing term in the science of form”. In: *Paleobiology* (1982), pp. 4–15.

[45] Couvet DF Bonnemaison P-H Gouyon. “The maintenance of females among hermaphrodites: the importance of nuclear-cytoplasmic interactions”. In: *Heredity* 57 (1986), pp. 325–330.

[46] Pierre-Henri Gouyon, Jean-Pierre Henry, and Jacques Arnould. “Les avatars du gène”. In: *La Théorie Néodarwinienne de l’Evolution. Belin, Paris* (1997).

[47] Peter Grassberger. “Information and complexity measures in dynamical systems”. In: *Information dynamics*. Springer, 1991, pp. 15–33.

[48] Susse Kirkelund Hansen et al. “Evolution of species interactions in a biofilm community”. In: *Nature* 445.7127 (2007), pp. 533–536.

[49] Garrett Hardin. “The tragedy of the commons”. In: *science* 162.3859 (1968), pp. 1243–1248.

[50] Charles B Heiser. “Introgression re-examined”. In: *The Botanical Review* 39.4 (1973), pp. 347–366.

[51] Josef Hofbauer and Karl Sigmund. *Evolutionary games and population dynamics*. Cambridge University Press, 1998.

[52] Rosmarie Honegger. “Developmental biology of lichens”. In: *New Phytologist* 125.4 (1993), pp. 659–677.

[53] Darren E Irwin, Jessica H Irwin, and Trevor D Price. “Ring species as bridges between microevolution and speciation”. In: *Genetica* 112.1 (2001), pp. 223–243.

[54] Charles A Janeway et al. *Immunobiology, 5th edition*. Garland Science, 2001.

[55] Morten Winkler Jorgensen, Esben Hallundbaek Ostergaard, and Henrik Hautop Lund. “Modular ATRON: Modules for a self-reconfigurable robot”. In: *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*. Vol. 2. IEEE. 2004, pp. 2068–2073.

[56] Stuart A. Kauffman. *The origins of order: Self organization and selection in evolution*. Oxford university press, 1993.

[57] Motoo Kimura. *The neutral theory of molecular evolution*. Cambridge University Press, 1984.

[58] Marc Kirschner and John Gerhart. “Evolvability”. In: *Proceedings of the National Academy of Sciences* 95.15 (1998), pp. 8420–8427.

[59] John R Koza. *Genetic programming: on the programming of computers by means of natural selection*. Vol. 1. MIT press, 1992.

[60] Christopher Langton. “SimLife from Maxis: Playing with Virtual Nature”. In: *The Bulletin of the Santa Fe Institute* 7 (1992), pp. 4–6.

[61] F William Lawvere and Stephen H Schanuel. *Conceptual mathematics: a first introduction to categories*. Cambridge University Press, 2009.

[62] Guy W Lecky-Thompson. *AI and artificial life in video games*. Cengage Learning, 2008.

[63] S Lembcke. *Chipmunk physics*. 2011.

[64] Nathan F Lepora, Paul Verschure, and Tony J Prescott. "The state of the art in biomimetics". In: *Bioinspiration & biomimetics* 8.1 (2013), p. 013001.

[65] Orgel Leslie E. "Prebiotic chemistry and the origin of the RNA world". In: *Critical reviews in biochemistry and molecular biology* 39.2 (2004), pp. 99–123.

[66] Ching Chun Li et al. "Population genetics." In: *Population genetics.* (1956).

[67] Helmut Lieth. *Phenology and seasonality modeling*. Vol. 8. Springer Science & Business Media, 2013.

[68] James Mallet. "Hybridization, ecological races and the nature of species: empirical evidence for the ease of speciation". In: *Philosophical Transactions of the Royal Society of London B: Biological Sciences* 363.1506 (2008), pp. 2971–2986.

[69] William Martin et al. "Modern endosymbiotic theory: Getting lateral gene transfer in-to the equation". In: *Journal of Endocytobiosis and Cell Research* 1 (2012), p. 5.

[70] Daniel W McShea. "Complexity and evolution: what everybody knows". In: *Biology and Philosophy* 6.3 (1991), pp. 303–324.

[71] Alexander Meduna. *Formal Languages and Computation: Models and Their Applications*. CRC Press, 2014.

[72] Julian F Miller. "Evolving developmental programs for adaptation, morphogenesis, and self-repair". In: *Advances in Artificial Life*. Springer, 2003, pp. 256–265.

[73] Julian Francis Miller. "Evolving a self-repairing, self-regulating, french flag organism". In: *Genetic and Evolutionary Computation–GECCO 2004*. Springer. 2004, pp. 129–139.

[74] Stanley L Miller et al. "A production of amino acids under possible primitive earth conditions". In: *Science* 117.3046 (1953), pp. 528–529.

[75] Tom M Mitchell. "Machine learning. 1997". In: *Burr Ridge, IL: McGraw Hill* 45 (1997).

[76] Jeffry B Mitton and Michael C Grant. "Genetic variation and the natural history of quaking aspen". In: *Bioscience* (1996), pp. 25–31.

[77] Alfredo Morabia. *A history of epidemiologic methods and concepts*. Birkhäuser, 2013.

[78] Martin A Nowak, Corina E Tarnita, and Edward O Wilson. "The evolution of eusociality". In: *Nature* 466.7310 (2010), pp. 1057–1062.

[79] Isabelle Olivieri and Pierre-Henri Gouyon. "Evolution of migration rate and other traits: the metapopulation effect". In: *Metapopulation biology: ecology, genetics and evolution* (1997), pp. 293–323.

[80] MIT Press, ed. *Artificial Life* (1993-).

[81] Trevor D Price, Anna Qvarnström, and Darren E Irwin. "The role of phenotypic plasticity in driving genetic evolution". In: *Proceedings of the Royal Society of London B: Biological Sciences* 270.1523 (2003), pp. 1433–1440.

[82] Przemyslaw Prusinkiewicz and Aristid Lindenmayer. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012.

[83] Paul B Rainey. "Unity from conflict". In: *Nature* 446.7136 (2007), pp. 616–616.

[84] Thomas S Ray. "Evolution, ecology and optimization of digital organisms". In: *Santa Fe* (1992).

[85] Luis Rene Rivas. “A Reinterpretation or the Concepts" Sympatric" and" Allopatric" with Proposal or the Additional Terms" Syntopic" and" Allotopic”. In: *Systematic Zoology* 13.1 (1964), pp. 42–43.

[86] Amir Sapir et al. “Viral and developmental cell fusion mechanisms: conservation and divergence”. In: *Developmental cell* 14.1 (2008), pp. 11–21.

[87] Winfried Schiebel. “The Cell Cycle of Physarum polycephalum1”. In: *Berichte der Deutschen Botanischen Gesellschaft* 86.1-4 (1973), pp. 11–38.

[88] PM Sheppard. “The evolution of mimicry; a problem in ecology and genetics”. In: *Cold Spring Harbor Symposia on Quantitative Biology*. Vol. 24. Cold Spring Harbor Laboratory Press. 1959, pp. 131–140.

[89] Karl Sims. “Evolving virtual creatures”. In: *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. ACM. 1994, pp. 15–22.

[90] JMW Slack. “Morphogenetic gradients—past and present”. In: *Trends in Biochemical Sciences* 12 (1987), pp. 200–204.

[91] J Maynard Smith. “Group selection and kin selection”. In: *Nature* 201 (1964), pp. 1145–1147.

[92] John Maynard Smith and John Haigh. “The hitch-hiking effect of a favourable gene”. In: *Genetical research* 23.01 (1974), pp. 23–35.

[93] John Maynard Smith and Eors Szathmary. *The major transitions in evolution*. Oxford University Press, 1997.

[94] Kenneth O Stanley and Risto Miikkulainen. “A taxonomy for artificial embryogeny”. In: *Artificial Life* 9.2 (2003), pp. 93–130.

[95] Michiel Van de Panne and Eugene Fiume. “Sensor-actuator networks”. In: *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. ACM. 1993, pp. 335–342.

[96] Thomas L Vincent and Joel S Brown. *Evolutionary game theory, natural selection, and Darwinian dynamics*. Cambridge University Press, 2005.

[97] Mitchell M Waldrop. *Complexity: The emerging science at the edge of order and chaos*. Simon and Schuster, 1993.

[98] Mitchell Whitelaw. *Metacreation: art and artificial life*. Mit Press, 2004.

[99] Neil Willetts. “The genetics of transmissible plasmids”. In: *Annual review of genetics* 6.1 (1972), pp. 257–268.

[100] Dennis Wilson et al. “A continuous developmental model for wind farm layout optimization”. In: *Proceedings of the 2014 conference on Genetic and evolutionary computation*. ACM. 2014, pp. 745–752. URL: http://oatao.univ-toulouse.fr/13059/.

[101] Mark Yim et al. “Modular self-reconfigurable robot systems [grand challenges of robotics]”. In: *Robotics & Automation Magazine, IEEE* 14.1 (2007), pp. 43–52.

[102] Bernard P Zeigler, Herbert Praehofer, and Tag Gon Kim. *Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems*. Academic press, 2000.