Computational Biology Course

Paulien Hogeweg

An overview of the lectures on bioinformatic processes, by Dieter Stoker

"Elephant, I . . . "

Contents

Chapter 1

Introduction

About this reader

Welcome to the Computational Biology course. This course focuses on modeling biological systems as the information processing systems that they are. The subject matter is intriguing, exhilarating, and confounding all at the same time.

The purpose of this reader is to bundle the core subject matter of the course, so that you may test or supplement your understanding of the subject matter covered in the course. The lectures are information-dense, so repetition of some subjects might come in handy. To help you, this reader roughly follows the course outline (it might even be a bit more elaborate on certain details). Be advised that the field of computational biology is in flux, and new additions to the lectures may not (yet) be in the reader. The sheets are available for study on the course website, and if concepts or examples are unclear, please do not hesitate to ask: chances are that others are just as befuddled as you are.

Lastly, please make no mistake: this reader is *not* a replacement of the lectures and practicals, but can be used as a complementary source of information, allowing you to reiterate the subjects at hand. Together, the lectures, the practicals, and this written work by Dieter Stoker, should provide you with plenty opportunity for the deepest solace: understanding.

This reader is not perfect. If you find any mistakes, please notify one of the teaching assistants (missing figure legends, typo's, etc. Anything is welcome!)

Course introduction

Organisms are information networks. Living systems are complex information processing systems. Therefore, their behaviour will often be counter-intuitive and strange. By using computational modeling, the tangled mass of complexity can be unraveled and made partially understandable.

In this course, we therefore focus on **dynamic bioinformatics**. Bioinformatics is defined as the study of information processes in biotic systems (Hesper and Hogeweg, 1970). There are two types, which can be subdivided by their approach:

1. **static bioinformatics**: real data from life can contain patterns. In static bioinformatics, one tries to reveal these patterns through data analyses, and by studying these patterns, one infers and/or hypothesises about the biological mechanisms which might have generated the data. Think of large-scale phylogenetic studies, or GWAS studies to pinpoint disease-causing genes.

2. **dynamic bioinformatics**: biological processes generate patterns. In dynamic bioinformatics, one constructs a formal model in which basic biological assumptions (derived from observations) are represented by some processes, and one can then study what patterns or results emerge from these assumptions.

Thus, the two approaches of bioinformatics represent the two opposite orientations between patterns and processes: static bioinformatics is to discover patterns and infer processes, while dynamic bioinformatics is to generate patterns from assumed processes. Obviously, the two approaches are complementary in bioinformatics, and thus for understanding information processes in biotic systems. Phylogenetic bioinformatics work on extant species has shown the prevalence of whole-genome duplications (Edger and Pires, 2009; Eric Schranz et al., 2012), and dynamical evolutionary modeling can investigate why these are so prevalent(Cuypers and Hogeweg, 2012, 2014; Cuypers et al., 2017). Nevertheless, in the following, we focus on the dynamic approach of bioinformatics because the static approach of bioinformatics is taught in many other courses.

The main aim of the course is to answer the following two questions:

1. **What is modeling, and how can we use modeling to gain insight into complex biotic systems?**

2. **What exciting biological insights (theory) have been obtained from models?**

Two central ideas in the course will be **models++** and **results++**. *models++* is when certain behaviours of a system happen in many different modeling frameworks. For example, you will be introduced to models with emergent and with imposed higher levels of selection (do not fret, these concepts will become clear in time), and we will see how these concepts are independent of the modeling formalism. *Results ++* is somewhat harder to define. The gist is that throughout the course, we often work with minimal assumptions and see what happens in a model, with the intent of explaining a certain phenomenon. Surprisingly, however, we often find much more interesting phenomena in the model than we initially set out to achieve, and these findings often match or explain real-world phenomena. This is *results ++*: when a model shows surprising, broadly applicable, or deeper unforeseen results that turn out to be very relevant to the modeled system, but which are most often not preconceived. Perhaps, at the moment, these concepts will not mean much to you. Bear them in mind throughout the course, and when concepts and the models supporting them are discussed, ask yourself whether they hold in different model frameworks (models ++) and what the unexpected, exciting deeper findings were (results ++).

To appreciate the power of modeling, and the strengths and weaknesses of different modeling approaches, we will start the course by going over various **modeling formalisms**: archetypal systems that can be used to model biotic processes. These topics are abstract, but before we can meaningfully describe biological systems, we need to know what modeling choices we have, and how they compare to each other. Thereafter, we will use these modeling systems to investigate diverse biological systems and processes, from the origin and first steps of life to animal behaviour and ecosystem dynamics.

The intuitive work flow in biology is as follows: observation of an interesting phenomenon → formulation of a hypothesis → testing of this hypothesis. In most of the work discussed in this course, we take a

different approach: we build a model with minimal assumptions, and see whether these assumptions are sufficient to generate the behaviour. If not, we investigate what parameters/assumptions could. This allows for a process of **'debugging' of assumptions**. If we see an interesting pattern in nature, for example that a specific plant species only occurs at a specific locations, we intuitively hypothesise that there must be a reason for this. Perhaps it is better at retaining moisture or gathering key nutrients in such an environment. In a model, conversely, we might discover that if we initialise several plant species with equal fitness, and allow growth and random dispersal in space, plant species will dominate in specific areas solely due to this random dispersal and following growth: where the traditional approach immediately seeks to explain a pattern, modeling allows one to discover that *no explanation is necessary.*

What is biology about?

What is the defining characteristic of biology? That is its multi-level nature or **emergence**: structures, patterns, and/or behaviours arise which are *not predefined*, *persist for some time*, and *operate at different space-/time-scales*. In models, we will often refer to these phenomena as **mesoscale patterns**, having their own dynamics and needing new words and concepts to describe them.

Biology encompasses many levels. It starts from the level of nucleotides and goes up to genes, proteins, protein complexes, gene regulatory networks, metabolic networks, cells, tissues, organs, organisms, and ecosystems. An organism is highly complex, and the impact of a change in a lower level (a mutation on the DNA) is highly non-linear (one mutation can completely alter an organism's phenotype, or do nothing at all!). Additionally, the system itself and all its regulation is evolved. Evolution has made and optimised the mapping from the DNA, to the phenotype, to the organism. This is not easily incorporated into models, but this multi-level operation is a core concept in biology. A question that arises and we must aim to answer is: how is it that, despite this extreme upscaling of effects, this extreme sensitivity, there is still enough robustness in biological systems such that they generally function quite well?

We thus study biological systems as multi-level systems and see what happens in the interactions between multiple levels. We will happen upon some unexpected effects that we would not have seen or understood if it were not for this (multi-level) modeling. We focus on information and its patterns:

1. information processing
2. information storage
3. information transmission
4. information accumulation

All this over multiple levels of organisation (from genomes to cells to organisms to ecosystems) and over multiple timescales. Think of the information processing from DNA to RNA to proteins, and up to the whole organism. Or think of how the DNA of an extant organism is an accumulation of information of a huge amount of past selection pressures on that organism and its forebears. We will ask the following questions in our (multi-level) modeling:

1. **Given known (or assumed) interactions at the micro level, what are the (counterintuitive) consequences?**
2. **Given simple local interactions, what complex behaviour does this generate in the system?**

Other names for these types of approaches are *systems biology, biocomplexity studies, and theoretical biology.* We will focus on specific biological examples, and gain general insight into multi-level processes from them. We will see several examples during the course where researchers seeking general insights from the get-go, who therefore investigated very general models, actually reached wrong conclusions.

After introducing **modeling formalisms** and concepts that these generate such as **mesoscale patterns**, we will study how space and population dynamics generate **multi-level evolutionary processes**. In the process, we will study the **information threshold**, **spatial pattern formation** and (resultant)

new levels of selection, **genotype-phenotype mapping**, **evolutionary dynamics**, **neutrality and robustness**, **long-term information integration**, and **evolution of evolvability**. After all is said and done, we will discuss some special examples that show the (explanative) power of multi-level modeling and/or how failing to account for multiple levels occludes correct conclusions on the workings of biotic systems.

Chapter 2

Modeling formalisms

What is a model: finite-state machines (FSM)

Models are caricatures

Given the complexity of biotic systems, drastic simplification for modeling purposes is needed and desirable. However, it is important to know what can and cannot be learned from models with their limitations and simplifications.

To truly understand a process, one needs multiple points of view. Different model types (caricatures) can be combined to arrive at a more holistic understanding of the underlying biological process. If certain emergent behaviours pop up in different model formalisms, that is a large clue that this is a true and important behaviour. Modeling is thinking in the most interesting simplification(s) (Manjarrez, 2011).

What you should take away from the sections below is -besides an understanding of the different model formalisms- that every type of model makes some assumptions to make modeling feasible. The best model of the world *is* the world, but we model to gain understanding about specific parts of it, with simplifying assumptions. Hence, every type of model is but a caricature of reality, and one should use different approaches to see whether the results are not just a consequence of the particular caricature you are looking at and/or reason carefully about which simplifying assumptions are justified or not for your use case.

A finite state machine: a prototype model

A finite state machine (FSM) is a model that has a set of inputs (I), a set of states (which is everything that is necessary to generate the next state from the inputs) (S), a set of outputs (think of behaviours or signals based on inputs and internal feelings/motivation) (O), a next state function (given that you are in a certain state, and have certain inputs, what is your next state? The next state function defines this) (NSF), and a next output function (if you are fed, food input should give no output. If you are hungry, food input should give feeding behaviour as an output) (NOF).

We denote such an FSM by <I,S,O,NSF,NOF>. As the name suggests, the number of states N must be a finite number. The **state** of the system is defined as all internal information of the system that we need to uniquely determine the next state and the next output. The NSF specifies how a set of inputs and current state leads to the next state. A NSF can be defined by a transition table which specifies the next state for all possible combinations of inputs and states. If an NSF is defined, the NSF should uniquely determine which state an FSM takes in the "next time" given the current state and input of the FSM. Thus, an FSM is deterministic, and has a **unique next state function**: given a current state, and inputs, there is always one and only one next state to go to. Similarly, a NOF specifies the next output for all possible sets of input and state. This is also defined by a table, and therewith, one can infer which output an FSM emits in the next time given the current state and input of the FSM.

Not all FSMs have all five variables. **input-output systems** have only <I, O, NOF>. Thus, no internal state comes in to play, but rather, the system is only dependent on inputs for it output. FSMs can also be composed of <I, S, O, NOF>, which are **input-output systems with memory**. Let us model a pendulum system as a FSM to see how the components an FSM needs to have vary according to the system under inspection.

A pendulum is normally at rest (0). At that point, nothing happens. We call this state the **attractor**: a state or set of states that the system eventually goes to and from which it does not leave (without external input). If we swing the pendulum outwards by a certain amount initially (1) and then measure the deviation as a response to this input after a set time (2), we can use a simple input-output system. We don't need internal states, because the input leads (via a next-state function that invokes basic physics) to the output. (Figure 2.1, left). We can also think of a variation: we could deviate the pendulum outwards by a certain amount (1), wait a set time, then take that time point as the input (2), and calculate the output (3). Now, a simple input-output system does not suffice: we need the angular momentum of the pendulum at T = 0 together with the input to calculate the output. (Figure 2.1, right). Alternatively, you could think of deviating the pendulum and giving it a small push. You would then need to know both the deviation and the force of the initial push (input and state). Thus, the components an FSM needs are not intrinsic characteristics of the system under concern, but depend on what the experimental set-up is. Interestingly, most behavioural and learning experiments are set up as input-output systems in an attempt to control for variation between individuals, i.e. the role of the state (memory) is minimised.

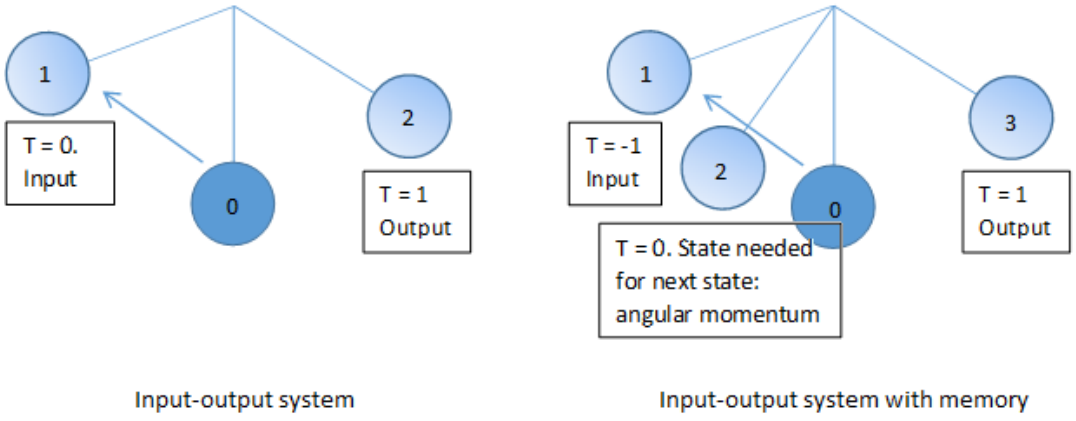


Figure 2.1: A pendulum illustrates an input-output system without (left) and with (right) memory. The internal state (angular momentum) is necessary to accurately model the movement of a pendulum, thus a plain input-output system does not suffice.

Autonomous FSM

Let us now look at an **autonomous system**: a system whose next-state function is solely dependent on its own state. Given that FSMs require *unique next states*, we can visualise this as a graph (Figure 2.2). In it, nodes are connected by arrows. In this scheme, each state has only one outgoing edge (remember, this is what it means to have a UNSF), but can have multiple incoming edges (multiple states can lead to the same state).

When we consider this simple visual representation of an autonomous system, we can derive several concepts that are important in modeling:

1. **attractors**, which we already defined as a subset of states that the system eventually goes to and from which it does not leave.

2. **multiple attractors**: there can be several subsets of states that the system can go to and not leave. (with a UNSF the system can converge, but not diverge)

3. **Garden of Eden states**: states in which the system can start, but to which it can never return.

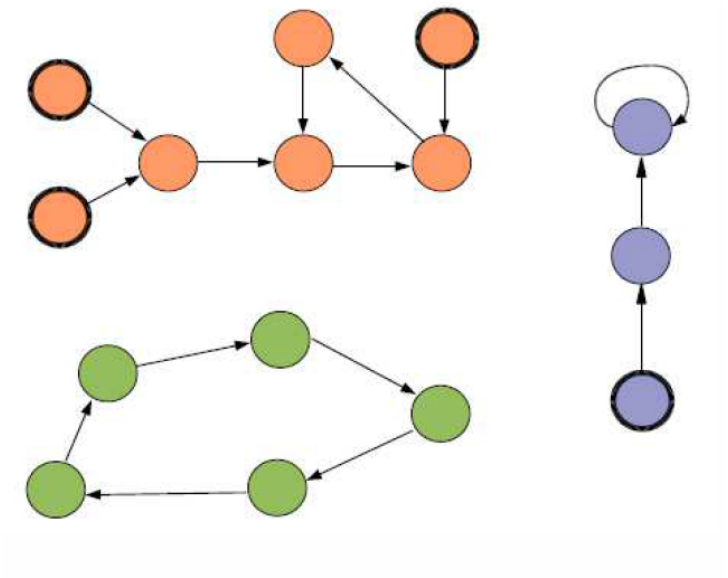


Figure 2.2: States of a system, visualised by nodes connected with arrows.

4. **domains of attraction**: the set of all states that eventually end up in a certain attractor.

FSMs must be **fully specified**: you must know for all possible combinations of states, inputs, and outputs what the next state function and next output function are. Otherwise you cannot compute the following time step. We use only such **fully specified** models in this course. In a complex system, which most biological systems are, assigning all possible transitions (called the transition table) is time-consuming and practically impossible. Just imagine making a huge table specifying for every combination of protein levels, mRNA levels, chromatin state of the DNA, somatic mutations in the DNA, protein phosphorylation levels, metabolite concentrations, etc. etc. what will happen. You would quickly need to define more combinations than there are atoms in the universe: it just isn't possible! Therefore, most models take short-cuts with respect to fully specifying a unique next state function so that modeling becomes a feasible endeavour. We will discuss some important short-cuts in the next section.

FSM short-cuts

Most or certainly many modeling in biology is done in terms of ODEs or MAPs. We assume some familiarity with the subject. Here follows a short overview of ODEs and MAPs (a possible shortcut to FSMs) to compare these classes of models to other model formalisms.

Ordered states (ODE, MAPs)

A possible shortcut to simplify the process of making a transition table for a FSM, is to use ordered states. Real numbers or integers can be ordered from small to large. These states can represent population densities, for example. Given these ordered states, one can define a next state function that maps numeric input to numeric output: a mathematical function. This function takes a variable, which is substituted by a state of the system, and must be valid for all numbers the model can take.

This is a huge short-cut. The transition table of a FSM is replaced by a (set of) numeric function(s). This short-cut imposes a lot of restrictions on what the model can do (we will see this later), and the number of variables is minimised. However, the short-cut allows us to relax the restriction that the number of states needs to be finite: the next-state function (an arithmetic function) can be valid for an infinite number of (real) values.

Two examples of this simplification are **MAPs** or difference equations and **ordinary differential equations (ODEs)**. Both these simplifications are frequently used to model *e.g.* population sizes of animals. They have a lot of assumptions in common. Both in MAPs and ODEs, the variables are all in terms of the total population: we look at the change in the population. Both the model input and the model

output are in terms of this population size. This assumes that all individuals in the population are identical, and that they all have exactly the same environment, where they interact with everyone and everything. Another way of saying this is that the system is **well-mixed**. These are rather important assumptions, and we shall see later on that they have major effects on model outcomes. ODEs and MAPs differ in their assumption on time. MAPs have discrete time steps, whereas ODEs use continuous time. One might reasonably wonder what kind of behaviour these discrete time steps might cause, and we will discuss this in the following section.

MAPs are of the form $X_{t+1} = f(X_t)$. As an example, consider yearly population growth of rabbits (N):

$$N_{nextyear} = r \cdot N_{thisyear} \cdot (1 - N_{thisyear}) \tag{2.1}$$

Here, r is the so-called **intrinsic growth rate**, and the population of rabbits is prevented from growing *ad infinitum* by having a negative effect on itself (*e.g.* resources are finite). As mentioned, this MAP has discrete time, which in this case, is in steps of one year. If we observe this system, how many rabbits do we find after sufficient time? For equation 2.1, depending on the value of r, the population of rabbits never reach a stable steady-state, and continuously over- and undershoots the equilibrium (Figure 2.4). In fact, for very high values of r, the system depicts **deterministic chaos**, and never visits the same state twice! In other words: the system is highly sensitive to initial conditions.

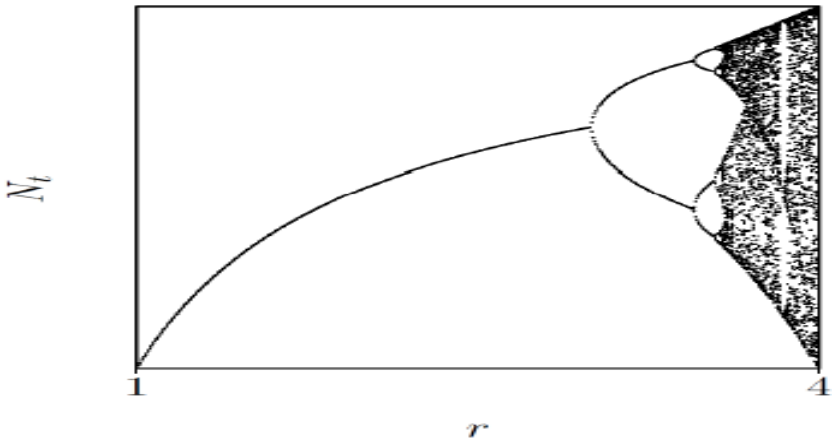


Figure 2.3: Bifurcation diagram for the steady state population size in the logic MAP. For increasing values of r, we first observe period doubling, followed by more period doubling, followed by deterministic chaos.

ODEs use continuous time, which can either be solved analytically or are numerically approximated. Consider again the population growth of rabbits:

$$\frac{dN}{dt} = r \cdot N \cdot (1 - N) \tag{2.2}$$

Do these ODEs, using the same equations as equation 2.1, also display *deterministic chaos*? The short answer is: no. By using continuous rather than discrete time, the system cannot "overshoot" the equilibrium without first visiting all the states in between. Since the system has a UNSF, there is no way to pass these in-between states and do something else the second time. In other words: trajectories cannot cross in ODEs. This argument extends if we add another variable (a 2D ODE like the one given below), since also in a 2D plane there is no way for trajectories to keep "missing" one

If you are unfamiliar with modeling ODEs for biological systems, we strongly recommend taking a look at the Systems Biology course reader (ten Tusscher, 2016), which introduces ODE analysis.

another. In 3D ODEs we *can* find *deterministic chaos* where trajectories follow non-periodic patterns, such that a small change of state at a certain time leads to an arbitrarily large deviation after some amount of time (see for example the Lorenz attractor).

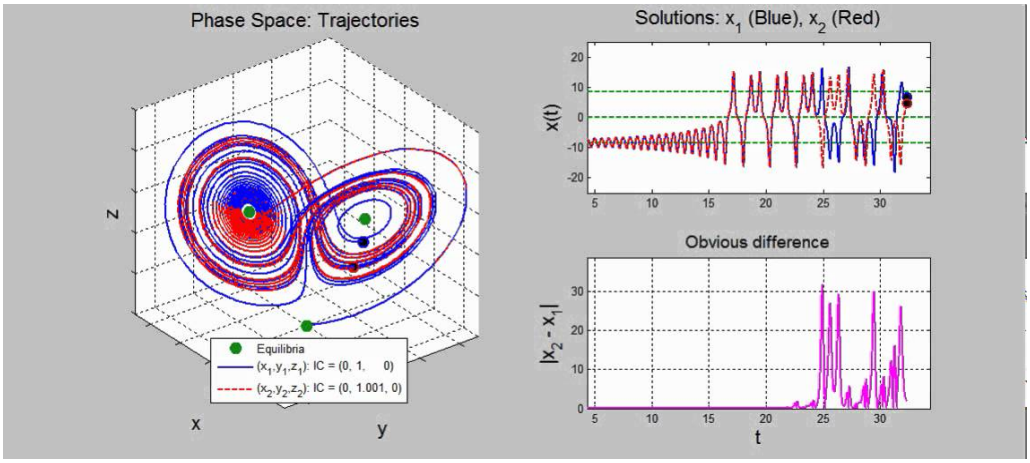


Figure 2.4: The Lorenz system has a beautiful butterfly shaped attractor in which, after some time, the two nearly identical initial conditions start to lead to very different outcomes. Also see animation on YouTube.)

Let us now briefly discuss 2-dimensional ODE systems. These are systems in which 2 variables interact. For example, consider a system of predators (N) and prey (R):

$$\frac{dR}{dt} = R \cdot a \cdot (1 - \frac{R}{K}) - N \cdot R \cdot b \tag{2.3}$$

$$\frac{dN}{dt} = N \cdot R \cdot b - N \cdot d \tag{2.4}$$

In this system, the prey are born according to a growth rate (a), and die according to the number of predators and the predator growth rate (b). It is assumed that predators grow equal to the amount of prey they eat. The predator population grows due to hunting prey, and predators die with a certain death rate (d) per individual. Lastly, K is the prey population size where prey birth rate is zero. In analysing this sytem, we can look at the **phase space** or **state space**: a 2-d projection where we can see, for every combination of R and N, what the system will do. In such a system one can draw **nullclines**: sets of states where the change in (time derivative of) either variable is 0. For example, if the amount of prey is equal to 0, the change in the amount of prey will always be 0. This is intuitive: both terms in the prey ODE above have the term R in them, so without prey, there is no change in the amount of prey. We often call these nullclines "trivial", as they are not very informative on the dynamics of the system we are interested in. When two of the non-trivial nullclines intersect, we might find more interesting behaviour, as they could describe a point in the system where both prey and predator no longer change. How do we go about determining this?

For every point on the phase space, we can look at the eigenvalues of the matrix of partial derivatives at equilibrium (the **Jacobian**) to find out where the system will move for the next state. The Systems Biology reader contains information on using a graphical representation of the Jacobian. Using this information, we can draw a **vector field**: this field gives the direction of change at selected states (Figure 2.5, left). We can also draw a **trajectory**: the set of states visited from a specific initial condition. **Attractors**, in this case, are states or sets of states visited after enough time. These can be either **fixed points** or **limit cycles**, where the system never reaches a stable point but cycles through the same set of states *ad infinitum*. You can see from Figure 2.5 (right) that the trajectories all converge to

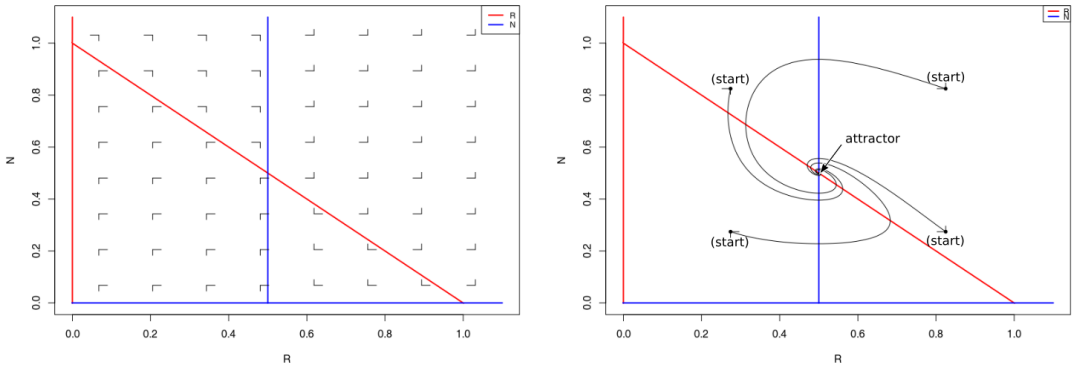


Figure 2.5: Phase space of a simple ODE system (Lotka-Volterra). On the left, the vector field is displayed, and on the right four trajectories are drawn towards the stable equilibrium.

the same attractor, independent of where they began. This is, in other words, a *fixed attrator*. Moreover, there is a clear spiraling into the attractor, which is why this specific attractor is called a **stable spiral**.

A general overview of all ODE attractor types is given below (Figure 2.6). The following are stable equilibria: stable node (3), stable spiral (6). Unstable equilibria also exist: unstable node (1), unstable spiral (4), saddle point (2). By calculating two special values of the 2-dimensional matrix of the Jacobian (the "trace" and the "determinant"), we can know a lot about the system.

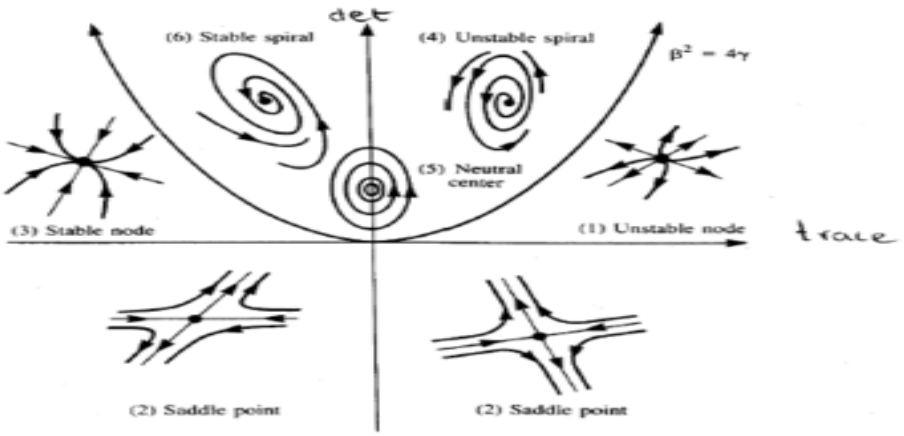


Figure 2.6: Different types of attractors in ODEs

MAPs and ODEs summarised

MAPs and ODEs both deal with the complexity of a FSM by making the states ordered, and changing the next state function into an arithmetic function. This also relaxes the need for finite states: in principle, the function can be valid for any real number (which means there is also output for biologically nonsensical states!). The difference between MAPs and ODEs is that MAPs have discrete time steps, whereas ODEs use continuous time. This results in the possibility of 1-dimensional chaos for MAPs, whereas ODEs only know chaos for 3 (or more) dimensions. The population is the primary variable (we saw a population of prey and predators). There is a fixed set of variables: you define populations as the model entities, and also observe what happens to these populations. All entities of such a population are therefore the same, and all are assumed to have the same interactions and environment. These are big assumptions, as we shall see presently.

The basics

We will now focus on an important type of short-cut to FMSs used in this course: Cellular Automata (CAs) (historical background references: (Burks, 1970; Von Neumann et al., 1966; Guy and Conway, 1982). Remember, as discussed in section 2.1, we use short-cuts in modeling because the transition tables for complex (biological) systems are near impossible to define. Here, the short-cut is in the decomposition of the single large FSM into smaller sub-FSMs whose transition tables are easier to define. Intuitively, you could think of a string of bits of length 50. Bits can be either 0 or 1. In that case, the possible states of the system are 2^{50} (Figure 2.7). Defining a transition rule for each of these states is a daunting task. However, what if you instead focus on local interactions: make a rule that says that if either of your neighbouring bits is 1, and you are 0, you also become 1. If you are 1, you become 0 if both neighbouring bits are 0. If neither of these is the case: stay what you are. By considering only a subsystem (a bit and its direct neighbour(s)), suddenly these rules are enough to define a simple behaviour for the whole system!

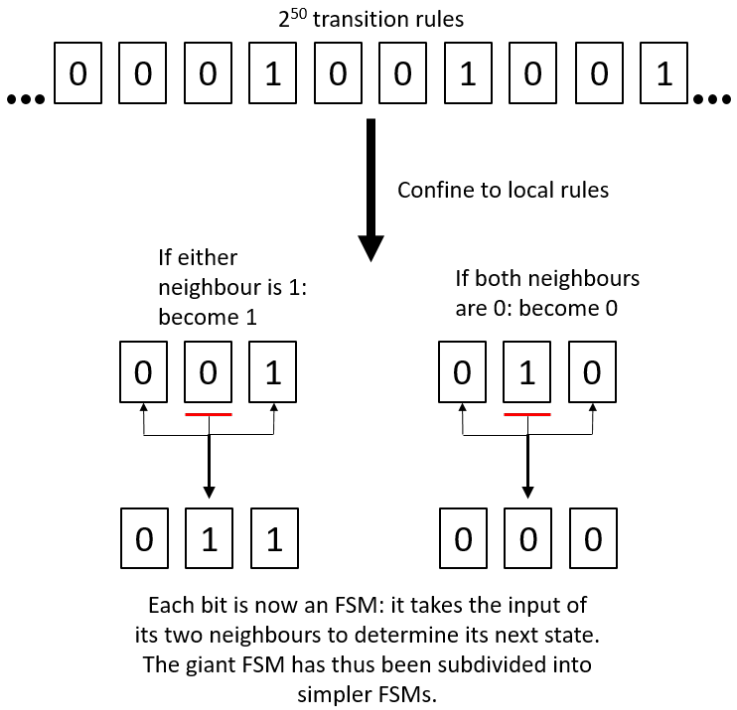


Figure 2.7: Decomposition of a FSM into local units whose transition rules are much easier to define. Made by Dieter Stoker.

We have now made a small FSM out of each bit in the bit string: every bit takes as its input the states of the two neighbouring bits, and we have defined transition rules for the full string. Because the smaller FSMs are connected, the result is a single large FSM. Thus, we have fully designed a FSM without explicitly stating all rules in the transition table. Note that this is a significant short-cut: we assume that local interactions are enough to define the behaviour of the system as a whole. If we think of biological systems, however, this short-cut might well hold: rabbits in one location might not be particularly affected by rabbits in a location 100 km away, but rather be affected mostly by their close neighbours.

The example above describes a 1D CA, where each cell has a neighbour to its right and to its left. Next, let us consider a 2D CA, a grid where all cells are influenced by the cells adjacent to them. We could take different local neighbourhoods in this case, and two often-used ones are displayed in Figure ??: the **von Neumann neigbourhood** (the 4 horizontally and vertically neighbouring grid points) and the **Moore neighbourhood** (all 8 surrounding grid points). For the von Neumann neighbourhood in a

binary CA (cells can be 0 or 1), the following holds: in one cell of the CA (one FSM) there are $2^5 = 32$ possible transition rules (because you can define a specific rule for every combination of FSM state and neighbouring state, there are 5 grid points under consideration (yourself and 4 neighbours) and every FSM can be either 1 or 0). There are thus 2^{2^5} possible transition tables (for every combination of states, you can have the next state function return either 1 or 0). Contrast that with a field of 1000 by 1000 grid points where you would *directly specify* the transition table for the whole FSM: there would be 2^{10^6} rules, which is a number much greater by far than the number of atoms in the universe!

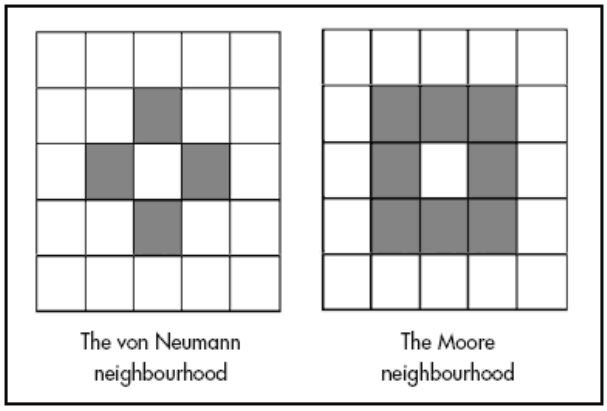


Figure 2.8: Commonly used neighbourhoods for 2D CAs.

Thus, the assumption that the transition rules are the same for all grid points drastically lowers the number of rules needed. Note that each grid point (FSM) requires input and is therefore non-autonomous. However, when the whole grid is updated (next state function applied) synchronously (all grid points updated at once) the full grid is an autonomous FSM (there are no inputs to the large-scale FSM). In this shift from global rules to local rules, we introduce **localness** or **the speed of light**: specifying the system based on local interactions ensures that perturbations need time to percolate through the field. A perturbance in the upper right corner of a grid will not immediately be reflected in the lower left corner of a grid. Why? Because interactions are only local, if an organism starts on the upper right of a field, it can only reproduce in adjacent squares. Thus, for the organism to reach the lower left corner requires some or many state updates, depending on the size of the field. There is therefore a maximum speed at which change can happen. In our own universe, that is the speed of light.

Note that a CA thus works by having each grid point take input from its direct neighbourhood, and on that basis going to a certain next state. In practice, this means that if you wish to define a system with reproducing organisms, empty squares take their neighbourhood, see whether there are organisms in that neighbourhood, and with a certain probability (growth rate of organisms) copy their neighbour's state. Thus, you always reason from the view of the grid point and how it itself changes, not how a grid point changes its neighbouring grid point. This might not make much sense intuitively, but you will practice this in the exercises.

We will now focus on three well-known examples of cellular automata. They have very simple local rules, but exhibit complex behaviour.

Modulo Prime CA

The rules of Modulo Prime can be defined for any prime number p. The state of each grid point is one of the numbers $\{0,1,2,...,(p\text{-}1)\}$. The CA is binary if $p = 2$, as it is for the following example. The next state function (NSF) of a cell is defined as follows:

1. Compute the sum of all neighbours in the von Neumann (4-cell) neighbourhood

2. The next state is this sum **modulo (%)** p (modulo is what is left over after the maximum amount of full divisions of a number by another, e.g. $10\%4 = 2$)

When this simple rule is implemented, and one starts with a specific initial state of the field (for example, in the shape of a dog like in Figure 2.9), after a given amount of time steps, the initial pattern is repeated and replicated at other positions in the field. This is a strange, funny behaviour, but drives home a very important point. This behaviour is not observable at the macro-scale (the level of the whole field): if you would plot the population of 1s and the population of 0s over time, there would be no indication that this happens. It is also not defined at the micro-scale: any random initial pattern you put in is replicated, all through a simple sum and the modulo. This behaviour occurs, but we can *only see it* when we look at the **mesoscale**: the scale between the micro- and the macro-scale. Only on this intermediate scale, that was *not predefined* in the model, something special happens and the pattern is visible. This mesoscale pattern (local configuration of cells) is not an attractor, but can be seen regularly on this non-predefined scale. Hence there is a clear predictability, but again, only on this **mesoscale**.

Said another way: you probably didn't guess that this pattern would emerge from these micro-scale local rules. If you look at what one grid point does over time, its sequence of states might be something like [0,0,0,0,0,0,0,1,0,1,0,0,0,1] etc. That is, it's zero for a long time, then the pattern passes over it, it gets activated a bit as the pattern is spreading in the vicinity, then it's zero again, etc. But if you would just draw a graph of any one cell's state over time, you really wouldn't know what's going on. The same holds for the macro scale: a graph counting 1 and 0 over time would be quite uninformative. However our evolved visual processing capabilities immediately pinpoint that some very special behaviour indeed is going on, an emergent pattern that we can see and describe in language other than how we defined our rules and counts of ones and zeros: repeated copying of an initial pattern.

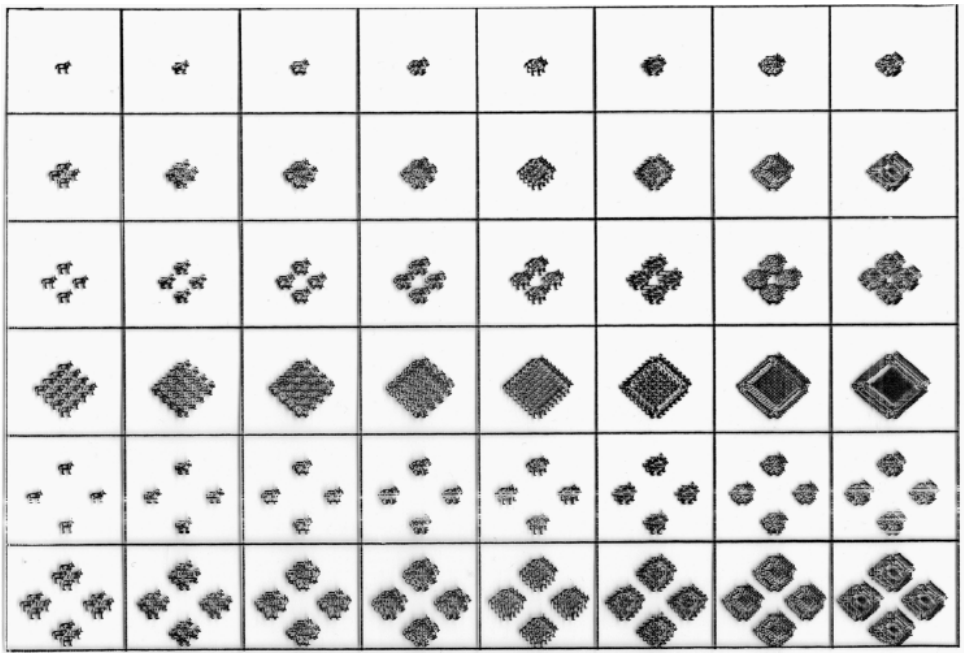


Figure 2.9: Modulo prime (p=2) produces a replicating doggy.

Game of Life CA

The rules for the Game of Life were originally defined by John Conway and first published in 1970. They are as follows:

1. A cell remains 1 when it has 2 or 3 neighbours (1s)
2. A cell becomes 1 when it has 3 neighbours (1s)
3. Otherwise, a cell becomes 0

This fully deterministic (arbitrary) rule, however, leads to completely *unpredictable* behaviour: the final

outcome of simulations starting with different initial conditions cannot be predicted. There is, to this day, no procedure that can predict whether 'life' will persevere or not for all possible initial conditions.

Many mesoscale entities are formed, each with different dynamical properties in time and space (blinker, glider, pulsar). Furthermore, this system displays long-range interactions that occur between traveling signals (e.g. the traveling gliders). Therefore, although the system is fully locally specified with simple rules, new levels of information processing emerge: signals from different locations can interact and form novel patterns. Even when all mesoscale patterns are defined, including their interactions, the system is still unpredictable. Nonetheless, those mesoscale patterns which have explanatory value are worth identifying, because they can have relevance to the behaviour of the system as a whole. In sum, we see here how very simple local rules generate (maximally) complex behaviour through the mesoscale patterns they generate. Note also that the *speed of light* is again very important here: signals can move about in the system, but they can only propagate with a certain maximum speed.

This example illustrates one very fundamental point: that *fully deterministic systems at a local level can generate unpredictable behaviour at a global level through emergence of novel entities at an intermediate level*. This means that in order to find out the particular behaviour of the system for a particular initial condition we can only "let life live its life". We cannot predict what will happen, so we need to let the simulation play out. While the rules of Game of Life are extremely simple, through its locality, it is as complex as can be, and no algorithm can predict what the outcome of all different initial conditions will be.

Voting rule or majority rule CA

This CA has a very simple rule set:

1. Choose the Moore neighbourhood (9 neighbouring grid points)
2. If <= 4 neighbours in state 1: next state = 0.
3. Else next state = 1.

This system is used as a model for many processes, perhaps even too many. It has applications in physics, social science, voting behaviour, and biology. Interestingly, this simple model can also give a profound insight. If you compare the two outcomes in Figure 2.10, which one would you call more ordered? Which one is more random?

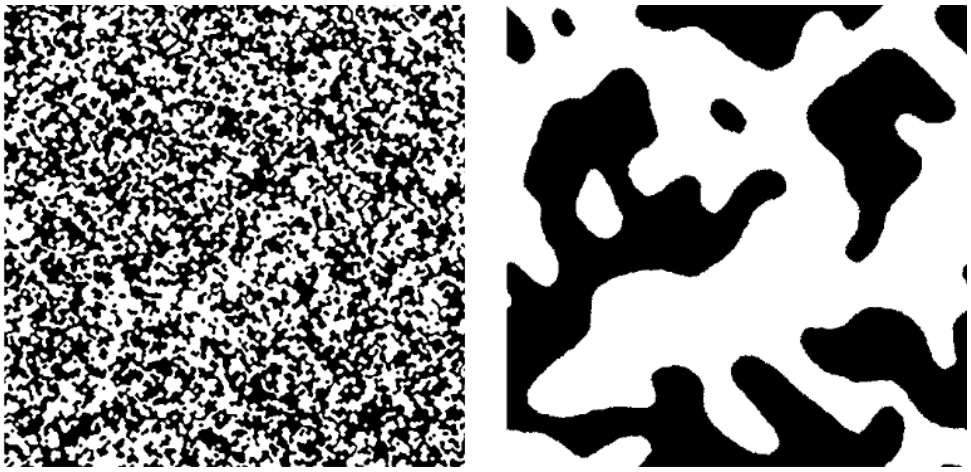


Figure 2.10: Vote produces spotty patterns, where introducing randomness of a *wobble* in the rules makes the pattern less like noise!

Chances are that you picked the left figure as the most random one, whereas the right one seems more ordered. This is not wrong, but what created them was the exact opposite: the left figure is a classical

voting rule CA, randomly initialised, but following the deterministic rules defined above. The right figure, conversely, is a voting rule CA with a 50% chance of the opposite behaviour from what is defined in the rules happening. Thus, *adding randomness can increase the pattern or order in a system.*

So what does this mean?

These are toy examples, no doubt about it. We don't contend that Game of Life models biology, say, or that modulo prime is secretly something that happens in bacteria. Instead, these three eye-catching examples immediately show that even with the simplifying assumption of only local interactions, you can get wildly unexpected behaviours and emergent patterns. So unpredictable in fact that for Game of Life there is no algorithm to predict the outcome for all initial states: you have to simulate and see. Fancy that for a system with 3 simple local rules and 2 states. As an aside, Game of Life is Turing-complete (although Microsoft Powerpoint and a host of other things are too). In practice, this means you can build a computer in Game of Life. People did that, and they were successful. Look it up! The voting rule quickly shows another counterintuitive result: add randomness, gain more pattern. We are trained to think order comes from specific causal factors. Here, in an exceedingly simple system with local rules, you see the opposite. Now imagine how infinitely more complex real biological systems are. Imagine what strange emergent patterns are hiding in *them*. Hopefully, you appreciate that to get a grip on what's happening we can't just go around hypothesising based on what we see, because we are hard-pressed to think about emergent behaviours of complex systems. Rather, we could take a step back with simple (local) models and see what happens when we turn the model's knobs.

Modeling biology with CAs

Cellular automata have been used to make models in various areas of expertise:

1. Experimental mathematics (Beyer et al., 1985; Burks, 1970)
2. Artificial life (Von Neumann et al., 1966; Braitenberg and Langton, 1988)
3. New physics (Wolfram, 2002; Margolus and Toffoli, 1987)
4. Local interactions in biology (Hogeweg, 1988)

In all cases the main defining point is the discrete nature of the formalism in combination with local interactions and complexity arising from the emergence of mesoscale patterns. For instance, Toffoli used a CA to explore the implications of discrete particles (particle concept) and contrast that to field theory (continuous variables) (Margolus and Toffoli, 1987). Wolfram even went so far as to consider the entire universe as a 3D CA in which one only had to find the transition table in order to gain full understanding (Wolfram, 2002).

PDEs (partial differential equations) are similar to ODEs, but on top of describing rates of change over time, they also describe rates of change in space, using differential equations. They thus have have continuous space as opposed to discrete grid points in CAs. If a droplet of water is added in the middle of a PDE, and we integrate over a small amount of time, there is immediately an infinitesimally small amount of water in all other grid points. In other words, PDEs are not subject to the aforementioned concept of **the speed of light** which CAs (and this universe!) have. PDEs are sometimes hailed as better CAs, but in the context of biological systems, 0.000000001 rabbits do not exist. If PDEs are bounded to contain true 0 values if the real values become too low, they behave much like CAs as far as spatial pattern formation is concerned. Oftentimes, CA are seen as a worse version of PDEs.

The "attofox" problem:
The attofox problem is named after the term atto, which indicates a number of $10^{(-18)}$ in the metric system, and the fox. Models were made for fox population dynamics surrounding the construction of the tunnel across the canal to Great Britain. A concern was that foxes with rabies might traverse it to Britain, and start an epidemic. There were some worrisome models that predicted that epidemics of such foxes would rage across Great Britain for years to come, starting anew every time from the same point of origin, and killing huge numbers of foxes. It turned out that the continuous waves of rabies depended on populations of affected foxes with rabies in the attorange surviving and starting the new spreads (Mollison, 1991). $10^{(-18)}$th of a fox is hardly likely to survive, much less start epidemics that sweeps the nation. Therefore, the term attofox problem is an evocative term to warn against some of the dangers of the ODE modeling formalism.

However, that does not do cellular automata justice. Instead, CA allow us to focus on discrete entities (such as cells) and focus on local interactions that, together, can form complex patterns.

To illustrate this point, we will focus on a model of B-cell nodule formation in lymph nodes. In lymph nodes, the B cells are clumped within a mass of T cells, forming B cell nodules. Both cell types enter the lymph node randomly. Two questions thus arise:

1. How are these nodules formed?

2. Why did the system evolve this pattern?

To answer these questions, Hogeweg constructed a CA model (Hogeweg, 1989). The 2D grid represents a cross-section of the lymph node. The state of each grid point represents the absence/presence of a T cell or a B cell. The next state function for a grid point is defined in terms of birth/death, and influx/efflux of cells. T cells and B cells continuously move through the lymph node. Therefore, there is influx and efflux on the cross-section. The model assumes *random* influx of T and B cells. T cells proliferate independently, whereas B cells only proliferate with local help of T cells (they need T helper cells for this, (Nutt et al., 2015)).

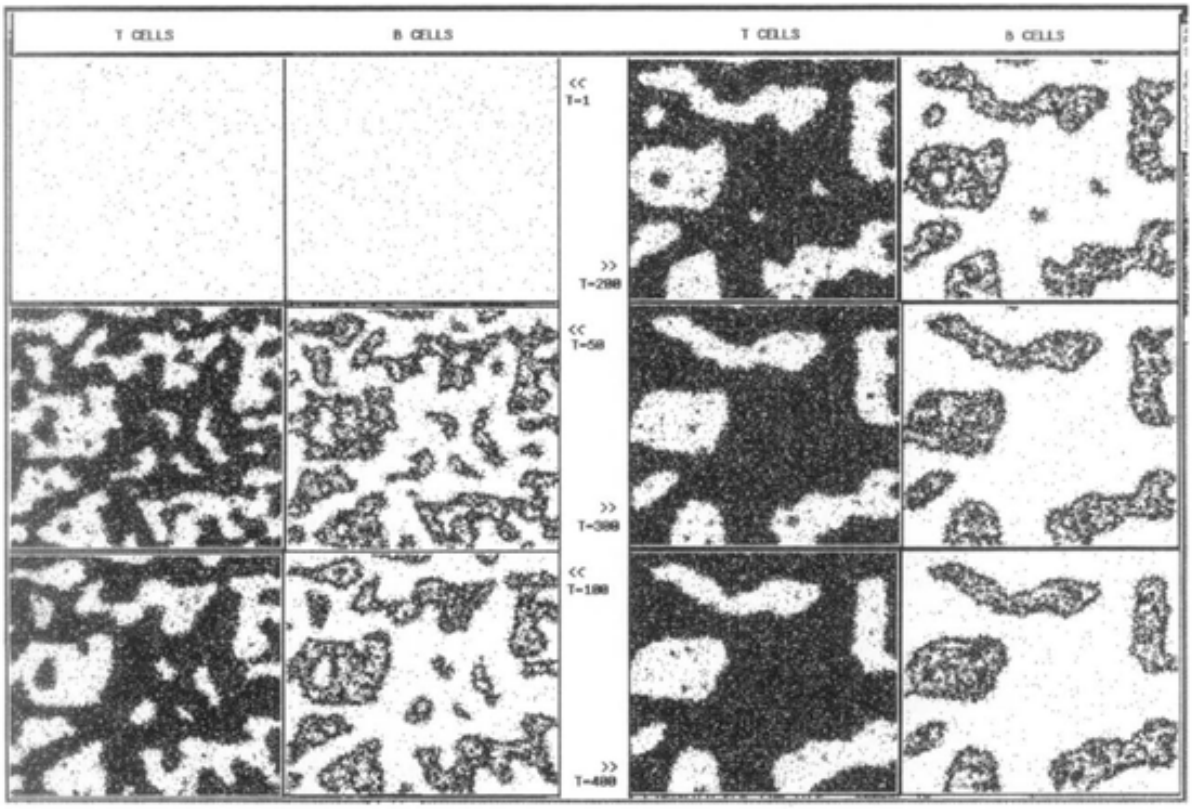


Figure 2.11: When implementing simple rules of T and B cell behaviour, the T and B cells segregate into the pattern observed in reality. I.o.w.: the pattern is the default, and might not need any further explanation.

This simple model leads to a striking observation: B cell nodules form, including a higher density of B cells at the edge of nodules (where they are stimulated to divide by T helper cells). This is similar to observations in real lymph nodes. This demonstrates an important point: *patterns need not be the result of an active process or mechanism and do not (always) require a functional explanation.* No such process was included in the model, and no benefit or function for the system was posited. In fact, in this case, this arrangement is *detrimental* to the system. Patterns with clumped B cells tend to slow the B cell-T

cell interactions needed for proliferation. Hence, proliferation would be optimised if the system would be well-mixed. It is, however, very hard to remain well-mixed when proliferation happens locally!

The example thus illustrates an important point: patterns need not be products of optimisation or come about by a specific mechanism. Patterns should, instead, be our default expectation: *in biology and CA models, non-homogeneity due to local interactions is the norm.*

Cellular automata thus provide a powerful modeling formalism to study the effects of local interactions in discrete entities, while allowing mesoscale patterns to arise and add to the dynamics in the system. As such they have been used as paradigm systems. Important is that such models can offer us *baseline expectations* for a given system: what happens given only the minimal rules needed for the system to work.

Generic behaviour of CA systems

We have seen what cellular automata are, how they work, and what we can do with them. For ODEs and MAPs, we can classify the behaviours into three large classes:

1. Fixed points (equilibria)
2. Limit cycles
3. Chaos (deterministic)

What behaviours might we expect from CA systems? What does an arbitrary CA do for arbitrary initial conditions? Wolfram studied this question in 1D CA systems using random transition rule tables with varying neighbourhood sizes and found different classes of CA behaviour:

1. Class I: move to a uniform state (either all 1 or all 0)
2. Class IIa: localised domain (limit cycle)
3. Class IIb: non-localised domains (limit cycles)
4. Class III: non-periodic, non-localised (high dimensional chaos)
5. Class IV: long transient, unpredictable (universal computation)

These classes were defined according to the effect that perturbations have on the behaviour of the system, i.e. what happens to disturbances? Do they percolate through the system, or do they have no effect? In Figure 2.12, we see slices of a 1D CA over time. The upper part is the initialisation with one disturbance, and the next slices show what happens to the 1D CA states over time. In Class I, the system returns to the fixed point. In Class II one can change from one attractor to another but the disturbance is limited to the attractor one is in. In Class III disturbances have a non-local impact and percolate throughout the field. This class shows high-dimensional chaos, but its statistical properties have a short transient. In Class IV disturbances can spread or not, die out or not, can be non-local and have a long transient. In other words they are highly unpredictable. Moreover, new entities with their own behaviour and interactions arise which lead to a new level of description and dynamics. Such entities have been suggested to have the capability of **universal computation**. This means that everything that can be calculated can be calculated in such a system. At the same time, this means that there is no algorithm that can perfectly predict, from all initial conditions, what the outcome is. Thus, though the system is simple to define, the behaviours CAs support in these systems are as complex as possible, i.e. all possible calculations can theoretically be done in them, and they are unpredictable.

To summarise: we are trying to see in a general way what can happen in 1D CA models, given different transition rules (example: If I am 0, and my neighbours are both 0, I become 1) and neighbourhood sizes (do you only look at direct neighbours? Or those beyond them as well?). You can try out all these systems and visualise what happens over time, as in the figure. So what kinds of things do you find? Well, some rules either all become 0, or all become 1. If you flip the state of one initial grid from 0 to 1 or

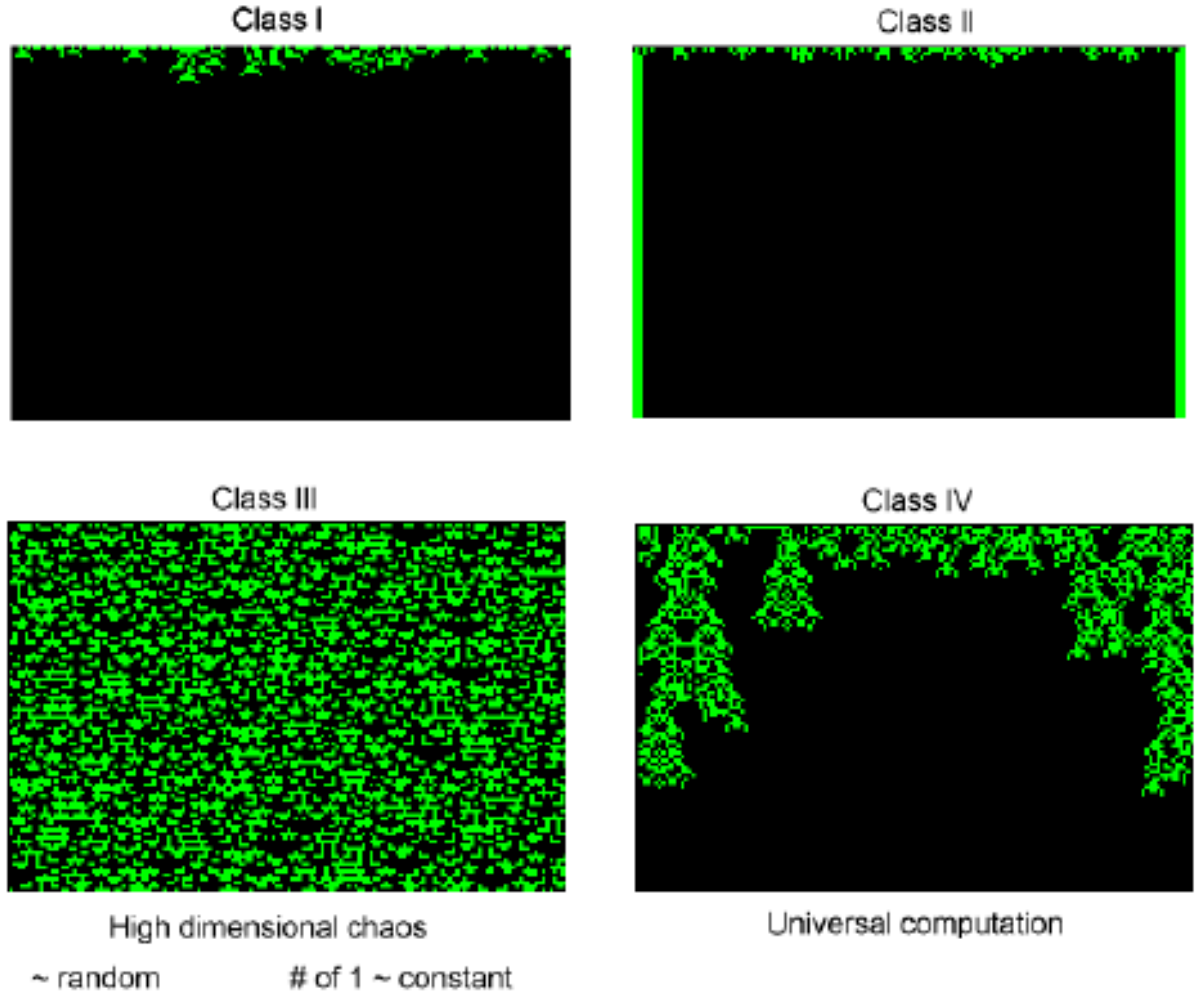


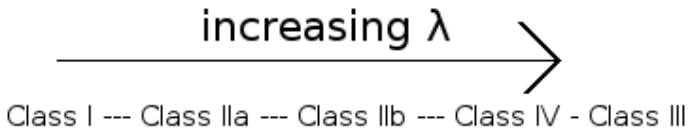
Figure 2.12: Wolfram's Classifications of CAs

1 to 0 that doesn't change anything. In class II you see that the system keeps a certain state (attractor). You can imagine that if you flip one bit in the beginning, this attractor might look slightly different. Say 3 cells on the left are green, rather than 2. But that's it. In class III you see the states are continuously changing and all over the place. For these rules, if you change one bit this effect will spread out over time and affect the whole thing. At the same time: the whole thing already is completely chaotic., and you do see that the number of 1s and 0s is relatively constant over time. Class IV is very unpredictable, patterns flare up locally and then die down again, or not. Here, the only way to know what happens if you change one bit is to watch and see.

In a strict sense, it is impossible to predict from the rules in which class a CA will fall. However, for almost all cases, we can predict the class by using Langton's ordering parameter lambda (λ): the number of rules which lead to the quiescent state (which is one of the CA states) (Langton, 1991; Seifter and Reggia, 2014). In a 1D CA, 0 is the quiescent (inactive) state, whereas 1 is the active state. He defined

$\lambda = (K^N - Nq)/K^N$, where K is the number of states, N is the number of neighbours and Nq is the number of rules leading to the quiescent state (i.e. the number of times the quiescent state is found in the transition vector: the vector of all possible outcomes of the next state function). The ordering parameter λ varies between 0 and (1 - 1/K). Hence, for binary CA the maximal value of lambda is 0.5.

Using this ordering parameter it becomes clear that as lambda increases one transverses from Class I – IIa – IIb- IV - III. Class IV is between the periodic and chaotic phase and occurs in a vanishingly



small parameter region (i.e. a vanishingly small region of lambda-values). This parameter regime clearly shows very interesting and unpredictable behaviour, but if it is so rare, is it important to consider? The answer is yes, because as we will see during the course, evolutionary/living systems tend to move towards behaviours that fall in this category, they go to the *border of order* or *edge of chaos*. These concepts will be treated more fully later on. Additionally, though this regime of Class IV almost never occurs, the fact that it can occur is already interesting. Most local rule sets give unsurprising or static behaviour, but unpredictability is possible. In that sense, it is an **existence proof**: it shows that with very simple rules, maximal complexity and unpredictability can result, which is an interesting finding in and of itself. This might sound like gibberish. Think of it like this: we define 1D CAs, basically as simple a model as you can get. We exhaustively look through all possible transition rules. We find that most resultant behaviours are boring: all 0 or all 1 pretty quickly, or simple limit cycles. Predictable. Then some look like madness (class III) and for a very few we really can't predict what happens next and changing 1 bit can change the behaviour of the system completely. That's pretty amazing. As we start using this model formalism to model biology we should keep this in mind.

Let us look at some examples to see what we can do with λ. In Modulo Prime (with $p = 2), K = 2, N = 4$ and $Nq = 8$ (since half of the 16 possible combinations of the four neighbour states will lead to a next state of 0 (quiescent state), $\lambda = (16 - 8)/16 = 0.5$. Hence, according to the ordering, Modulo Prime is class III, and is therefore highly disordered and chaotic. This is true, and can be illustrated by tracking the percolation of a single pixel change in a given random initial condition. If such a change is tracked in time it displays a fractal pattern which spreads throughout the field (Figure 2.13). On the left are two CA fields. One has only 1 state different than the other. On the right there are snapshots over time of the difference in the two fields, and on the far right, there is a space-time plot (i.e. how the change percolates through the surface over time). The figure is from unpublished work by Paulien.

The Voting rule also has $\lambda = 0.5$, again because half of the possible neighbour combinations will lead to a next state of 0. However, this case is not chaotic, although it is maximally disordered. It is therefore Class II. Hence, this is one example of a CA which is an *exception* to the classification by the ordering parameter λ.

An important point shown here is that generalisations, or model outcomes, are based on *almost all cases* (i.e. there are exceptions to the rule).

Mean field approximation/assumption (MFA)

One analytical drawback of CA models is their unpredictability and therefore the need to let them live their lives (i.e. run the simulation and observe the outcome). This can be very time-consuming. Moreover, each simulation is a specific case (parameter condition) because it is necessary to fully specify the model. In order to do **parameter sweeps** one must therefore do many simulations. This can be quite cumbersome which has led to attempts to obtain **mean field approximations (MFA)**: equations that strive to capture CA dynamics in an ODE format.

Mean field approximations are done by short-cutting on localness and therefore assuming *well-mixed* and *continuous variables*. Implicitly one therefore assumes that the *local pattern* do not matter and that *stochasticity* and *discreteness* are also irrelevant. Given what we have seen of the behaviour of CA models, this might not be wise. Nevertheless, let us construct an example model and see what happens.

We construct the MFA of a simple birth-death CA: Consider a binary CA where the state of a grid point represents the absence/presence of an individual. If a grid point is empty, it draws a random neighbour,

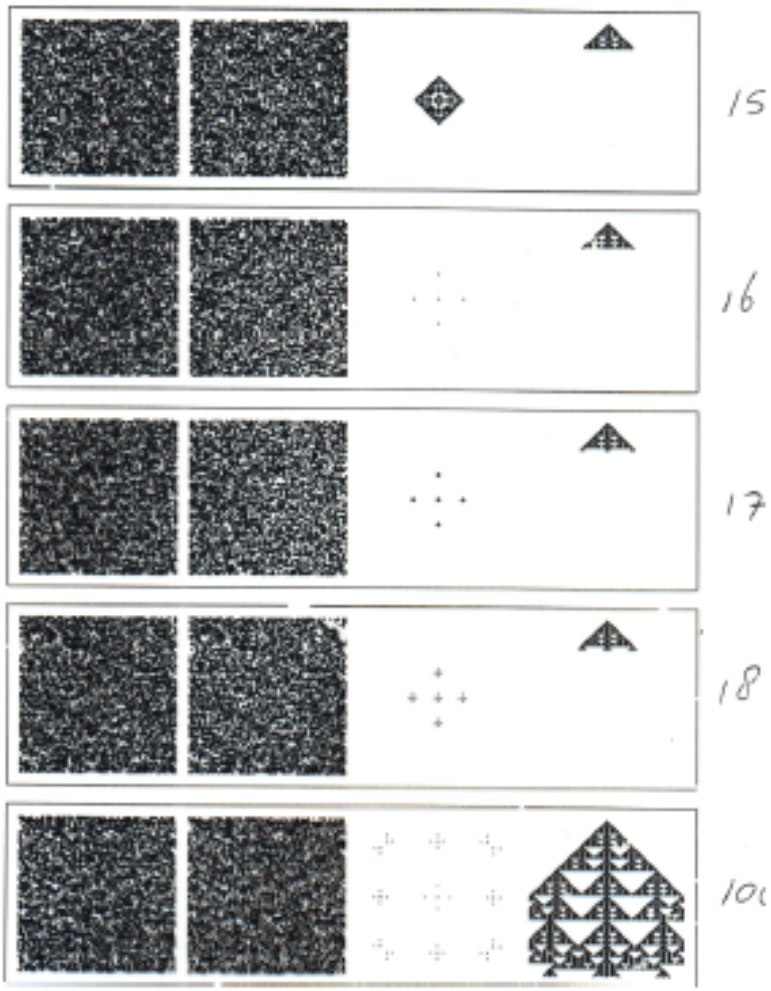


Figure 2.13: Disturbances in modulo prime take time to propagate. When a single pixel is changed in the middle, and the difference between prime with and without the disturbance is depicted, we can see how the distubrance propagates slowly over the entire CA.

and if this neighbour is an individual, it reproduces into the empty square with probability ("birth rate") b. Furthermore, filled grid points "die" with probability d. Let N be the total population size. Birth events take place if an empty square "meets" an individual. Hence, the MFA is given by:

$$\frac{dN}{dt} = b * N * \frac{E}{T} - d * N$$

Here, E is the number of empty patches, and T is the total number of patches in the grid. If we scale $T = 1$ and let N be the population density, the MFA simplifies to:

$$dN/dt = b * N * (1 - N) - d * N =$$

$$(b - d)N - b * N^2$$

This is the standard logistic growth formula.

One obvious question is whether the behaviour of such an approximation is similar to the original CA. The short answer is *no*, and why this is the case will be handled in-depth during the exercises. In ODE approximations of a CA, all individuals are assumed to see the same, "average" neighbourhood, while

in a CA, individuals vary in the neighbourhoods they are surrounded by. In order to compensate for this shortcoming MFAs have been constructed which try to incorporate some information from the local neighbourhood. These are called *pseudo-spatial models* (for example, models by Tilman (Tilman, 1994)). Such models are sometimes erroneously referred to as spatial models, but they only incorporate the first order effects of local neighbourhoods. This means they still ignore pattern formation and growth at edges which is often crucial in determining dynamics. Even higher-order approximations (that include more of the "local" neighbourhood) do not help to alleviate this problem since they still only at best incorporate local neighbourhood effects.

Simply put, the importance of local interactions that are at the core of the CA formalism cannot be fully translated into ODEs. However, one can instead use ODEs as a **mean field assumption**. In that case, we do not try to approximate the CA by means of a MFA, but instead ask: *What happens in the extreme well-mixed case, in contrast to the local case (the actual CA)?*. In this way, one uses both formalisms as paradigm systems and can learn more about the system under consideration. How important is locality to the observed behaviour? What would happen if the system is assumed to be well-mixed? Using different model formalisms to study extreme cases can be very useful. Expressing the dynamics of one model formalism in another, however, cannot work perfectly.

Using ODEs as MFAs will therefore never be a good replacement of CA models. The **mean field assumption** is however a good tool to discover what happens to a system if it is assumed to be well-mixed and can serve to inform your baseline expectations.

Emergent mesoscale patterns

So far we have looked at how CA models allow for mesoscale patterns in the sense that simple local interactions lead to complex patterns. The main points that we made were:

1. Patterns are often the default expectation
2. Large-scale patterns form from local interactions (or disturbances)
3. Unpredictability arises despite deterministic rules. Thus, the simulation has to 'live its life' before one knows what will happen.

We have discussed how mean field approximations can be seen as a descriptor of a system. It is important to note that unique next state functions (UNSF) depend on the level of description of a system. A CA has an unique next state function. However, when we use a MFA to describe what the population of entitities in a CA does, there is no unique next state function. That is because the local environments and mesoscale patterns matter, and the population can thus change in multiple ways depending on those patterns. You will practice this in the exercises. On the level of the variable "population size" (which is used in ODEs and MAPs), a CA does not have a unique next state function.

CAs vs IBMs
We can contrast the CA with individual-based models (IBMs). There, the focus is on individuals, who can be moving around in space, and exhibit certain behaviours (move towards food, feed, reproduce). In a CA, the basal unit is a discrete grid point (space) that can be occupied by an individual. Space-based models, like the CA, need a *predetermined* number of states and variables to be a **dynamic system**. In IBMs there can be a changing number of states (more individuals can be born, which influence the behaviour, and add extra possible states to the system on the go) and space is often continuous, therefore this is an undefined system in terms of dynamic systems. Just to be clear: what characterises a dynamic system is that it is a description of pre-defined entities (variables, which can be individuals or concentrations of molecules), whose interactions over time are defined (either via rates of change or finite next state functions). A CA has a predefined number of spaces that interact in defined ways. In an individual-based model, the number of individuals can vary, and hence the interactions vary, so this is not what we call a dynamic system. If this is all specific modeling mumbo jumbo to you, that is fine, just make sure you know that CAs and IBMs work from different assumptions, and that there might therefor also be fundamentally different behaviour in both.

Crutchfield and Mitchell conducted a groundbreaking study on mesoscale patterns in 1D binary CA by using filtering techniques to discover complex behaviour in seemingly undynamic CA models (Das et al., 1995; Crutchfield and Mitchell, 1995; Hanson and Crutchfield, 1997). A typical example is the study of Elementary CA 54. This 1D CA is named after the

binary code for its next state function. Every point only considered its two neighbours (e.g. if the CA was 10111, the middle bit's state is 1, and its neighbours are 0 on the left and 1 on the right). The rule set was:

1. if local state one of (111; 110; 011; 000) $\rightarrow$ become 0
2. if local state one of (101; 100; 010; 001) $\rightarrow$ become 1

In its space-time plot, rule 54 shows a seemingly uninteresting triangle pattern, with a predominant pattern interspersed with larger triangles (see Figure 2.14). Note that this is a 1D CA, so the figure is different time points of the system stacked on top of one another, with time moving from the top to the bottom of this figure. The dynamics of this system were studied by filtering out the predominant background pattern of small triangles, and instead showing the deviations of the pattern and marking them in black. What results is the emergence of a non-regular pattern showing mesoscale patterns that travel through space and time (see right figure, this is reminiscent of Class IV behaviour). This is interesting, but how does it help us understand the behaviour of the system?

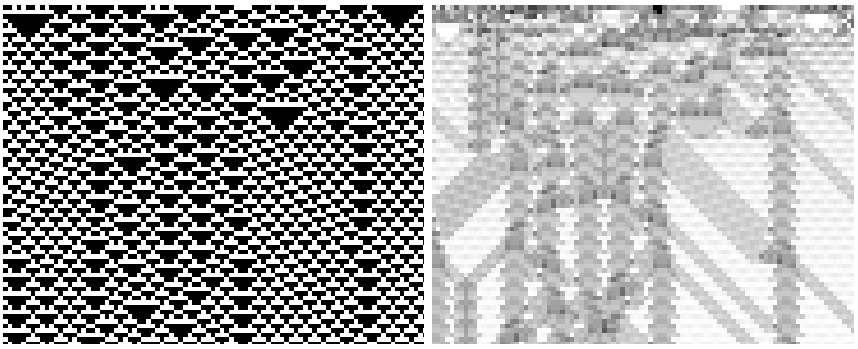


Figure 2.14: Interactions between mesoscale patterns in Crutchfield and Mitchell's CA 54. Left: space time plot of Rule 54. The right panel is the same, but the predominant patterns have been filtered out so we mostly see everything that is NOT a part if this typical pattern. It is a bit blurry, but notice how the changes in the left panel are propagated through the diagonal interactions in the right panel.

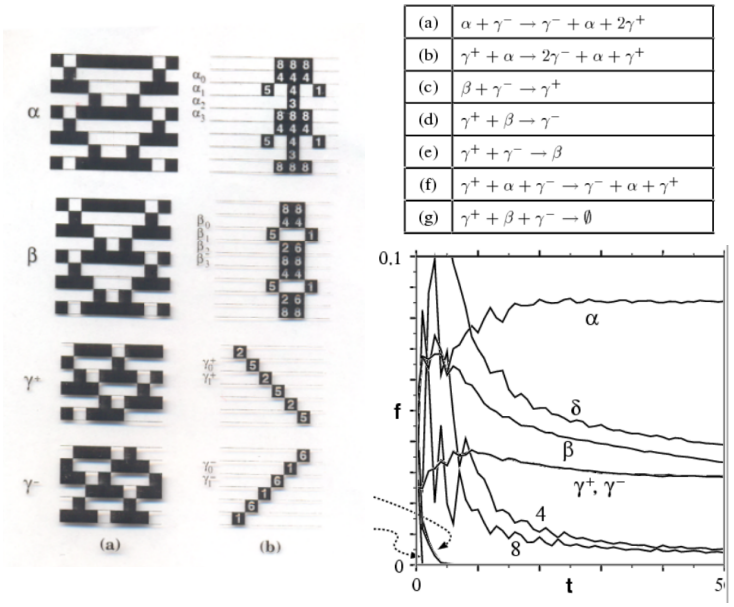


Figure 2.15: Mesoscale patterns in the work of Crutchfield and Mitchell are 4 types of "particles" with very particular rules.

We know the micro-scale rules of the system (transition rules). But can we classify the behaviour of the mesoscale patterns? It turns out that we can. In this case, we can classify the mesoscale patterns as 4 types of "particles" $(\alpha, \beta, \gamma + and \gamma -)$ which interact in particular ways.

We can then derive a description of the system at this higher level using ODEs to describe the particle "concentrations", which is not much different to what is done in chemistry (using mass-action terms to describe reactions). Note, however, that in these reactions within the CA system, there is no conservation of mass (mesoscale patterns can obliterate each other fully). Moreover, in time one can track particles

and observe what happens: first, there are remnants of the initial conditions, which die out quickly, after which particles continue in time. At another higher level one can plot the particles themselves (a further abstraction) and view the CA as particle interactions over time.

Such analyses have been done for all 256 elementary CA, i.e. all 1D CA with random transition tables. Although it may not be biologically relevant, it is conceptually important. This demonstrates that given a fully defined deterministic universe, there can be mesoscale entities with behaviour of their own which are best described beyond the "full description" modeling framework. In this case, the particles can be described in a reaction framework. We can now look at the system in this different way. While this does not help predictability given random initial conditions, it can yield deeper insight into the mesoscale patterns and how they interact to form the overall behaviour of the system.

Thus, mesoscale patterns can arise (though they may at first be difficult to recognise) that can be classified into different subsets (particles above). By studying these subsets, we can get a better idea of what is happening within the system, and even contrast it with what we might expect given well-mixed conditions (the ODE framework used here)

Predefined multiple levels (mesoscale entities): the CPM

In some cases, models are formulated with predefined multi-level properties in order to study the impact of such multiple scales or levels. In that case, the emergence of higher level entities is assumed and it is not the point of study to determine how they arise. Instead, the question is what impact they have given that they *are* there. In terms of CA this means that *local rules* are defined to be dependent on *predefined higher level entities*, such as cells.

The most famous multi-scale CA model is the Cellular-Potts Model (Graner and Glazier, 1992). In this model, single biological cells are defined in terms of a local group of grid points (lattice sites) with the same state (representing that they are a certain cell, see Figure 2.16). Such an individual cell can be given properties that make it act like specific (existing) cell types.

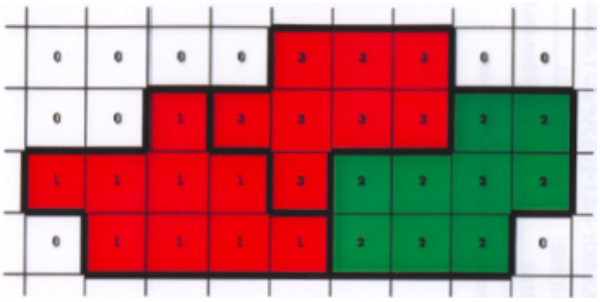


Figure 2.16: The Cellular Pots Model is a CA in which groups of grid points with the same state share a over-arching state (e.g. volume) that determines their behaviour.

In the model, biological cells are defined in terms of a volume (V) and a cell type. The cells are assumed to conserve their volume (actual volume) relative to some ideal volume (target volume, v). The cell membrane is assumed to bind to other cells (J_{ij}) or the medium (J_{im}) according to energy bonds. The ingenious part of the model is therefore how the local rules depend on the mesoscale properties of volume and surface energy minimalisation.

How is the behaviour of the local scale made dependant on the mesoscale? The free energy is minimised with volume conservation. This is done by considering the change in energy in the system for every change at the edges of cells in terms of changes of bonds and changes in cell volume. The free energy formula that is minimised is:

$$H = \sum \frac{J_{ij}}{2} + \sum J_{im} + \lambda(v - V)^2$$

Here, the first term describes the free energy at borders between CA grid points of different types (i.e. between cells). It is divided by two because this is done for all cell types, and you would otherwise count borders double (once from cell a to cell b, once from cell b to cell a). The second term describes the free energy of all borders with the medium. The last term is a penalty for the deviation of the actual volume v from the target volume V. Without this term, given that you strive for energy minimalisation, having no cells would be a viable proposition (no cells = no cell borders = total energy minimisation). That is not what the model should do.

To determine whether a cell should expand into a neighbouring grid point, the following equation is used:

$$P(i \to j) = 1 \; if \Delta H < -\beta; \qquad P(i \to j) = e^{-(\Delta H + \beta)/M} \; if \Delta H >= -\beta$$

In words, this does the following: if a certain change in cell type from i to j has a certain negative free energy (more than threshold β), it happens. If it does not have that, it can still happen with a certain probability, dependent on how much it goes against minimisation of free energy (ΔH), and on the parameter M. This is known as 'temperature' and can be interpreted as the propensity to do something that is contra to minimalisation of free energy.

In this way, the free energy for the whole system is minimised, but it is implemented on a local level. The higher level entity wants to have a certain target volume, and its type affects interactions of individual CA grid points. Note that for this model, to add anything to normal CA dynamics, the higher level cells need to be large enough: if a cell is only 3 grid points in size, local behaviours dominate over the larger scale. If this is not immediately clear to you, do not fret: you will work with these models in the exercises.

Let's apply the model to cell sorting. If one presses a sea cucumber through a gauze, the cells will apparently reaggregate. The question: is such reorganisation much different from oil and water separating after being mixed? In other words: perhaps certain cells just like to clump together and this is no magic effect.

In the Cellular Potts model it is possible for cells to squeeze past each other. Using different settings for differential cell adhesion, various forms of cell sorting can be represented in the model (Figure 2.17).

Note that, here, J_{ij} describes the free energy for a bond between cells of type i and j (e.g. $J_{ii} < J_{ij}$ means that bonding to type i is favoured over bonding to type j, since the free energy should be minimised).

This two-scale formalism is interesting because it can capture these sorting behaviours of cells. It is appropriate to study such sorting behaviours, but is not formulated to be informative about the emergence of higher-level entities like cells.

Moving against the flow in a CPM

Käfer *et al.* described cell movement caused by a gradient in a tissue with two different cell types, that sort through differential cell adhesion (see Figure 2.18). All cells are chemotactically attracted to the source of the gradient (which is on the **right**). Hence, we would expect all cells to move in the direction of the gradient. However, we observe that some cells move against the flow, in the direction of the red arrows. This movement is caused by these cells being pushed away by other cells. This can happen if this cell type is larger, has different adhesion settings, or is in the minority to start with. In this sequence of images, you see that the black cells first make their way to the front of the gradient, but then once they reach a large enough blob, they get pushed back. This happens because the pressure is large at the front: all cells want to expand towards this gradient, so the Hamiltonian is quite positive. The black cells want to adhere to each other more than the white cells. Hence, relatively, as long as black cells are still stuck together this is good for their Hamiltonian, while white cells have less of this positive bonus of sticking together. At the front, then, black cells get less of a penalty for expanding to the back (to regain some of their target volume as they are crushed by the pressure of all the other cells who want to be there) as they are still touching other black cells. This happens at the front, but then the cells just behind that lack some volume, and they also relatively more easily expand to the back than white cells. This causes the whole mass to move backwards.

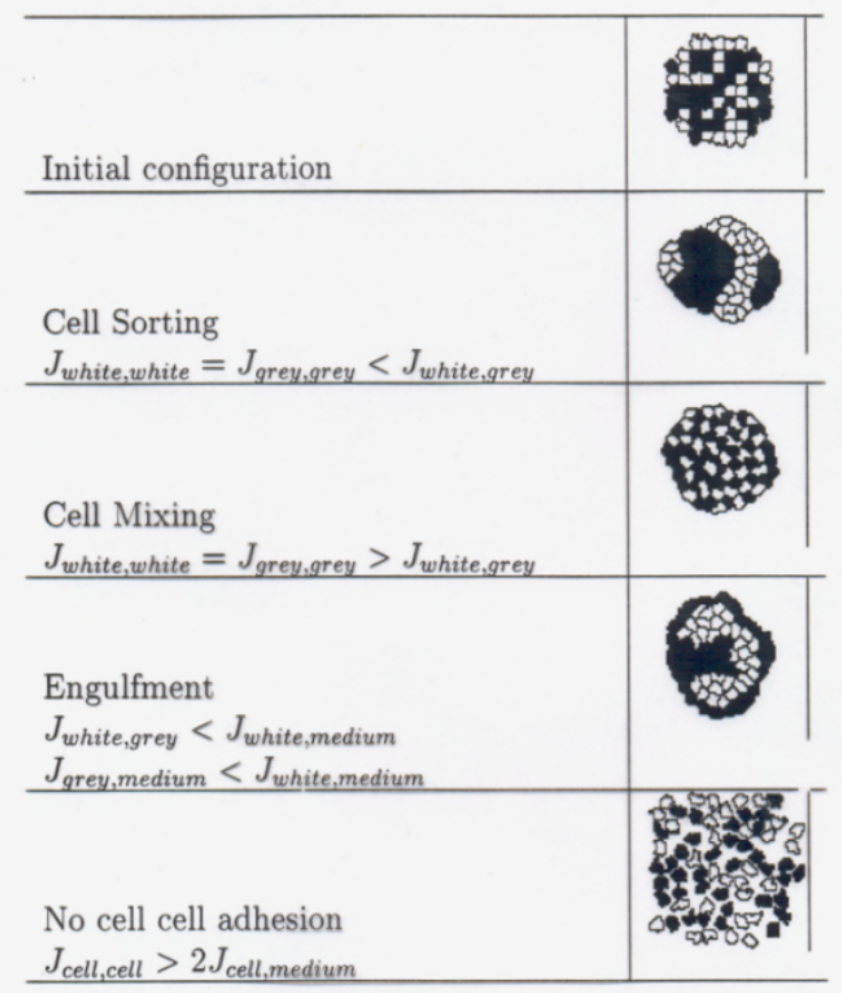


Figure 2.17: The Cellular Pots Model and cell sorting depending on differential cell adhesion (Graner and Glazier, 1992)

Movement and chemotaxis in a lymph node

We now revisit the lymph node, and consider the movement of T cells using the new model formalism. In experimental cell tracking studies, it was shown that T cells in lymph nodes move according to "stop-and-go" patterns, where cells move in a certain direction for a short amount of time, stop, change direction, move for a short amount of time, and repeat (Miller et al., 2003). The experimental set-up was such that, in a crowded lymph node, only a subset of cells was made fluorescent and their movement recorded.

One question that arises is whether this is an internal mechanism, or if it arises due to the spatial environment in the system: the crowded lymph node. To study this particular moving mechanism, Beltman *et al.* used a simple 3D CPM in which one extra feature was implemented: cells had a persistence of motion, i.e. a preference to keep moving in the same direction (see Figure 2.19). Biologically, you could translate this as the make-up of the cytoskeleton being optimised to move in a certain direction (with actin filaments pushing the cell membrane outwards there: this would take some time to change). An explicit stop-and-go mechanism was *not* implemented. Movement was entirely determined by interactions with the environment. The visualisation was the same as in the experimental study: a small subset of cells was coloured and followed over time. Despite not coding for it explicitly, the stop-and-go behaviour emerged.

Why did it emerge? The leftmost figure shows the time-course of movement of a single cell in a 3D system. Darker colours are later time points. No stop-and-go motion is evident: movement is continuous

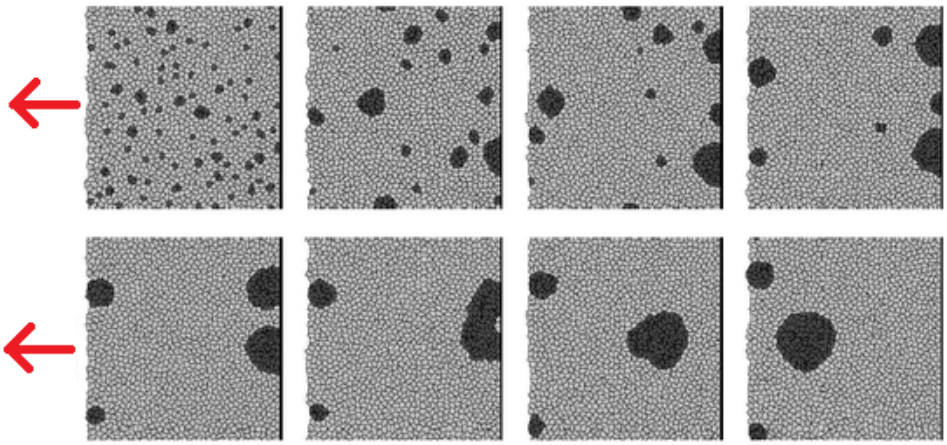


Figure 2.18: Cells can "move against the flow" in CPMs (Käfer et al., 2006)

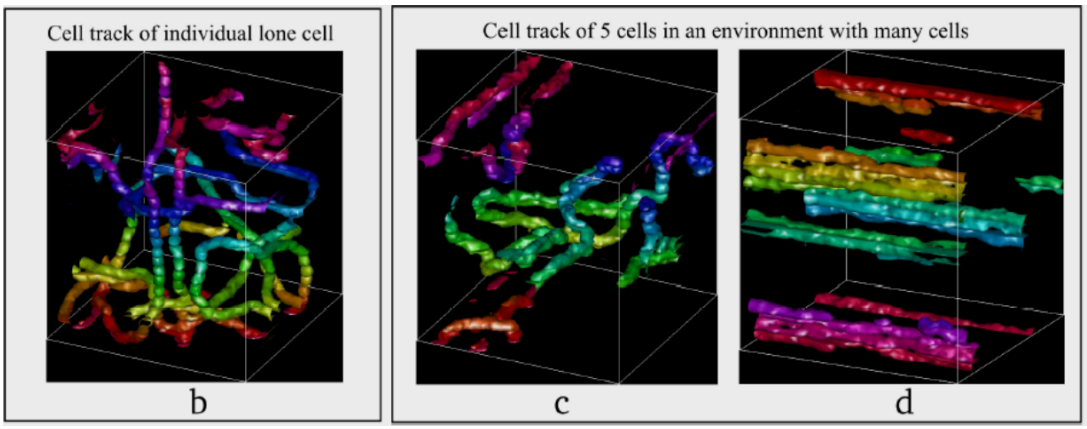


Figure 2.19: 3D CPM to study T-cell migration gives a stop-and-go mechanism, for free! Possibly, there is no further explanation needed. (Beltman et al., 2007)

and smooth. If you now put in many lymphocytes, and colour a subset of cells, the two rightmost pictures emerge: there is early disorder (left) and later ordering of movement in a single direction (right).

If cells bump into each other, they still wish to go in the same direction, but they need to go in a slightly different direction because they can not go *through* one another. They will try again, bump again, and turn a tiny bit. This repeats. This is the stop-and-go component. Eventually, a sort of alignment between the movement directions of cells becomes evident. In real lymph nodes, small groups of cells align their paths, bump into obstacles and stop, and then move in a somewhat different direction. Here, there are no other obstacles than deformable cells than can be squeezed past, and thus, all cells can align. In the real lymph node, there are obstacles, so no global alignment occurs, but groups of cells do align, and the stop-and-go phenomenon is simply explained by bumping into each other and trying again with a slight rotation.

Thus, no intrinsic stop-and-go programming of the cells caused this behaviour. Instead, if cells want to go in a certain direction, bump into something, and turn a tiny bit away while still striving to move in their original direction this behaviour manifests. Interactions then align themselves by moving in a particular direction. Again we find that out default expectation sometimes needs to be adjusted, because we have observed that certain behaviour arises from our models for free!

We can also include chemotaxis in the model, for instance to attract T cells to antigen-presenting dendritic cells (DCs). We might expect chemotaxis to improve the scanning of DCs by T cells, because the T cells can more easily find the DCs. However, when Riggs *et al.* made a CA-like model of this process,

they found that a random search strategy is optimal, and that chemotaxis actually hinders the scanning process(Riggs et al., 2008).

Why does this happen? If T cells are attracted to DCs, after a short period of time clumps of T cells form around the DCs, that hinder the movement of new T cells towards these DCs. However, this turned out to be an artifact of the T cells being modeled as fixed, non-deformable blocks. In reality, a T cell is deformable, and cells can squeeze past one another. Vroomans *et al.* later showed that if T cells are modeled as the deformable objects that they are (i.e. by using CPM in stead of a CA-like model), the "old" T cells are pushed away by the new ones, and chemotaxis actually enhances the scanning process(Vroomans et al., 2012). This example warns us to beware of modeling artifacts, and illustrates the importance of using different modeling formalisms. Here, the CPM is more appropriate, because T cells don't truly block other T cells that wish to move past them: cells are deformable and can push and squeeze.

In conclusion, cells are deformable, highly viscous objects that can wiggle past each other, and modeling them as such actually makes a difference. Furthermore, cells usually live in a cramped environment, with many other cells. The examples show the importance of including these cell characteristics (using CPM) if we want to understand cell movement.

Timing regimes in CA

So far we have considered CAs where updating over the field each time step is done synchronously: **synchronous updating**. The rules we instated were checked simultaneously for each grid point, and the whole field was then updated at the same time. However, this is a particular choice of the timing regime. Everything updates at once. An alternative would be asynchronous updating. Synchronicity was chosen for simplicity, but it can be considered an unfair short cut relative to finding out about how local rules lead to higher order complexity. It introduces a universal tick of time that governs when local interactions take place. At that point, all local interactions happen at once, so they are not independent. One might call this a special case and it could be considered artificial.

As an example, take a cell that has 8 empty spaces next to it and a chance of 1 to divide. In a synchronously updated CA without randomness, it can grow into all 8 neighbouring spaces within one time step. That might not be what you intend, but if everything is updated simultaneously, there is no way of knowing that the cell has already divided. This might seem esoteric, but the simple message is this: the very fact that you update your CA synchronously introduces an assumption that there is a universal time tick and that you can model things well in this way. As we have seen with the ODE framework and its (implicit) assumptions of well-mixedness and infinite population size, strange things can happen due to assumptions.

Asynchronous updating is different: rather than the universal tick of time for all locations, you can update the grid points one at a time. However, which pixel updates first? The upper left one? This would of course give rise to unintended behaviour, so we should either randomly pick a pixel to update, or randomize the update order of every pixel so that every pixel at least gets updated once. The latter is common practice, and is also implemented in the CA software (Cash2.1) you will use in the practicals. Note that by doing so, the unique next state function is gone: which CA grid point updates first, second, third, ..., will determine what the next state of the whole field becomes. Since that is determined randomly, there is no unique next state function. The example of the dividing pixel (like above) would play out differently: if 2 empty grid points update while the other cell still has a probability of 1 to divide, but thereafter it updates and does not have that probability anymore, the result is different. Besides *randomised order of updating*, asynchronicity can also be introduced by using *reaction rates* such that reactions at the fastest time step happen first (i.e. certain states have more chance of being updated, so this introduces more timescales).

It is important to get a feel for the timescales of certain processes, and how these might interact. In a synchronous CA, time works differently than in an asynchronously updated CA, and this might be important. Another avenue of influencing the timing of events relative to another which can be used in CA models is that of *claim mechanics*. You will practice using them in the exercises, but an introduction follows below.

Claim mechanics versus random neighbour in a CA

Claim mechanics refer to the fact that whether or not an organism colonises a spot is probably not solely dependent on spot availability. If we wish to model a forest with oak, pine, and yew trees, we could take a 2D CA, where state 0 is an empty spot, and 1, 2, and 3 are the tree types. In the exercises of the course, we often model reproduction by choosing a *random neighbour*, and if this is empty nothing happens, while if there is a tree there, it replicates into the empty space. That works, but it assumes that other neighbours have no influence on the process. In other words, this might correspond to looking on a day-to-day basis: chances of trees replicating are very small, and in time steps of a day, the fact that there are other trees growing nearby probably do not affect matters much.

If we take a step back, we should consider that tree saplings take at the very least a year to grow. We might thus wish to model the growth of the forest in years. In this case, time steps are large relative to the modeled system. During a year, trees make and disperse seeds, which grow. If two trees are next to the same grid point in a CA, it is not fair to have that empty grid point pick a random tree from its Moore neighbourhood and have that tree replicate. In truth, both trees would probably deposit seed in the same empty spot, and the saplings would compete during the year. Thus, there is an interaction time scale within a neighbourhood and a time scale over which multi-interactions are integrated: the local competition of saplings over the course of a year, versus the larger time scale where the victors of local competition can spread new seeds. In a CA, such an extra time scale can be incorporated by using a claim-mechanic. In that case, a square chooses, for example, two random neighbours, and they compete for the space (by weighing their probability of reproduction against each other).

In that way, you account for the effect that two seeds are dropped in a spot over the year, and one will eventually grow. However, this still gives a lot of noise in the system: if you choose two empty squares nothing happens, and if you choose an empty square and a yew, for example, the yew will always colonise, even if there are 5 pines also surrounding the square. If yews have reproductive advantage over pines, selecting only two random neighbours dilutes the strength of this effect. However, if you test the reproductive propensity of all 8 neighbours of a square against each other, then you give greater importance to competition during the time step of a year: all nearby squares (trees) compete for space.

Thus, it is all a matter of timescales. If you are modeling a bacterial population where there is a cheater that does not produce a common good and therefore has a reproductive advantage, it matters greatly whether you use random neighbour mechanics, or total claim mechanics with "all-against-all' -competition, or something in between. If you compete all vs all, the cheater is locally at full strength: it will win out over all non-cheaters, and since everyone is sampled, it will almost always colonise empty squares and take over the population. However, if you use just two random neighbours, and compete those, there is a larger chance of not randomly selecting a cheater, or selecting a cheater and a non-helper, thereby lessening the cheater's advantage and allowing non-cheaters to better survive. You will practice this during the exercises and/or a mini-project. The important thing is to think about how you set up competition, and how "fast" this competition is compared the other updates of the CA (*e.g.* mixing, diffusion, random events). Assumptions, assumptions, assumptions.

Diffusion in a CA

Interestingly, timing regime issues play a role in modeling diffusion: synchronicity generates a problem with respect to modeling particle conservation. Why does this happen? Diffusion is a random walk process, but in a CA particles are modeled by turning non-particles into particles without the option to simultaneously remove the particle (neighbour) that was the source of the new particle. Grid points only update their states based on their neighbours, they cannot affect what happens in neighbouring cells. Moreover, there is a conflict problem in the synchronous case when there are several particles as neighbours: which particle gets to diffuse to a certain spot? Attempts to solve this problem are as follows:

1. Using approximations. Here the average number of particles is maintained. This, however, leads to a clumped pattern of diffusion which is non-local (Figure 2.20)

2. Margolus diffusion. Here, alternate 4-square tile contra-rotation is used. This means the field is

divided up into squares of 4 particles, which are then rotated around. The next time step, different squares are selected, and the particles there are rotated. In this way particles are conserved and can pass each other(Margolus and Toffoli, 1987) (Figure 2.20).

3. Hexagonal diffusion. A 6-layered hexagonal grid is used where each layer is a direction which determines a next step position (REF). In the vertical axis, particles can react (bounce etc). For this model it is possible to show that the analytical limit quite well approaches continuous diffusion algorithms.

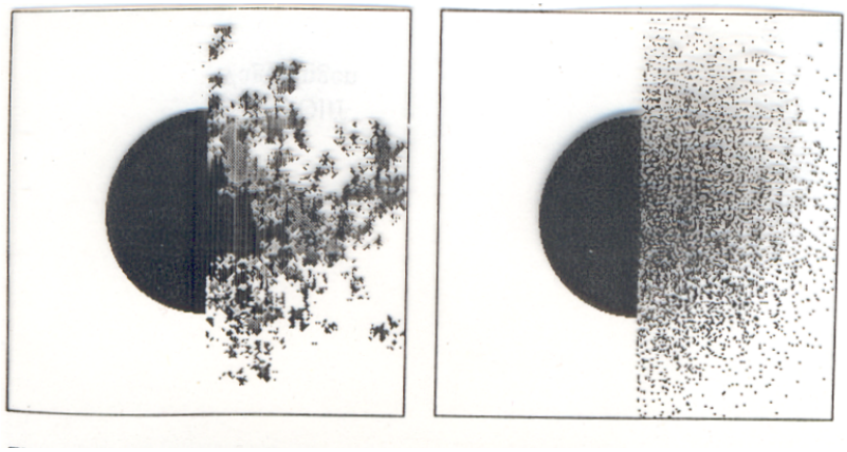


Figure 2.20: Left: psuedo-diffusion by "copying" from a random neighbour. Right: diffusion with the Margolus Diffusion algorithm. (Margolus and Toffoli, 1987)

The figure above shows a side-by-side comparison of clumped diffusion (using the approximation) and Margolus diffusion. Clearly, Margolus diffusion much better approximates actual diffusion. Interestingly, however, gene invasions are often conceptualized as diffusion. Since the invasion of a gene in a population is dependent on birth-death processes, this is in fact much better conceptualized as a clumped pattern, as one can see on the left (with clumps of individuals that have a gene and it thereby spreading locally, not as a free diffusion process).

The role of timing regimes in particle conservation makes an important point of how choosing a modeling formalism can push conceptualisation in certain directions. In synchronous updating, diffusion becomes a problem. We also discovered an interesting difference between particle conservation (as in actual diffusion) and birth-death diffusion (useful to model gene invasion/spread).

Of course conventionally diffusion is modelled in lattice maps or partial differential equations (PDEs) where lattice maps are the discrete approximations of PDEs and space and variables are considered to be continuous.

Boolean networks

We will now discuss another type of model formalism: Boolean networks. Kauffman conducted studies on random Boolean networks as a paradigm for gene regulation. Unlike a CA, that has a fixed connectivity (every pixel has 4 or 8 neighbours), every node in a network can have a different number of inputs, thus its own transition rule. A node (which can represent a gene for example) can be either on (1) or off (0). Each node can have inputs and connect to various other nodes, and there is no notion of "locality". Every node then integrates signals with its next state function, deciding whether to become 1 or 0 based on its inputs. Thus, Boolean networks have:

1. a specific network structure
2. specific interactions (not local like a CA)
3. specific transition rules per node (a Boolean function with k inputs)

A very simple Boolean network of three nodes is shown below (Figure REF).

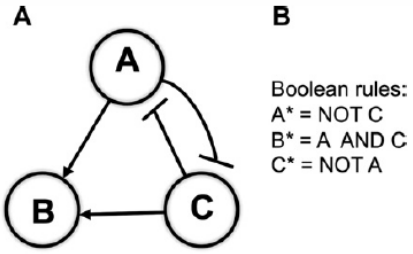


Fig. 1. Illustration of the Boolean rules corresponding to a simple regulatory network. (A) Network representation. A, B, and C are the network nodes. The directed edge → or —| denotes activation or inhibition, respectively. (B) Boolean rules governing the nodes' states for the network given in (A). For simplicity the state of a node is represented by the node name. The asterisk denotes the future state of the labeled node.

What do Boolean networks do? They consist of nodes (for us biologists, these can be genes, proteins, or transcripts, for instance) which can be either on or off. This means that dosage effects for an individual gene do not exist, it is either there, or it isn't. They can model, for example, the cell cycle. Often, there are checkpoints, and certain genes that need to be produced before the cell cycle is allowed to go to the next stage. As a simplification, these systems can often be modeled surprisingly well using Boolean networks with on-off states. Nodes depend on the inputs of other nodes to determine their state.

As you can imagine, much larger gene networks can also be modeled in this way. Sometimes, such a Boolean gene network can be directly mapped to a spatial CA. This is the case if the nodes can be structured such that they need only input from direct neighbours. One example is a 10-position 1D CA with a rule layer and a state layer, with different rules for specific positions. If a cell in a 1D CA takes 2 inputs, there are $2^4 = 16$ different possible rules. As an example:

1. if input $1 = 0$, input $2 = 0 \rightarrow 1$
2. if input $1 = 1$, input $2 = 0 \rightarrow 0$
3. if input $1 = 0$, input $2 = 1 \rightarrow 0$
4. if input $1 = 1$, input $2 = 1 \rightarrow 1$

As you can see, two inputs yields 4 combinations of inputs, which can be mapped to outputs in 4 different ways, thus leading to 16 possible rules. There are $2^{10=1024}$ initial configurations (ten nodes that can be either 1 or 0). Depending on the initial configuration, there are four **attractors** in this system. Half of initial configurations go to a specific attractor, so the attractors have different **domains of attraction**. (Figure **??**)

This is not trivial so let's explain this more deeply. Figure **??** shows two separate things. These are actually 2 1D CAs that interact. There is a *rule layer* that specifies, for each CA cell on the top layer what transition rule it will use. That is to say, 12 could stand for 'if your neighbours are 1, become 1, otherwise stay what you are'. Thus, on the 1D CA below, every grid point uses different transition rules. This is different from a normal CA: there, the transition rules are the same for every space (though what happens does, of course, depend on the state). By using this rule layer that determines the interactions and taking different initial conditions, network behaviours can be translated into a CA.

Thus, in the lower part of a figure, you see what a **trajectory** and **attractor** look like in a CA. These are all examples of cyclic attractors that can arise, and the CA visualisation of network dynamics allows you to see the different trajectories. Read on to find out the specifics about network trajectories and attractors.

Properties of networks

Boolean networks are very useful with respect to understanding the gene regulation networks that we are uncovering from data. For instance, we now know the full (or think we know the full) transcription

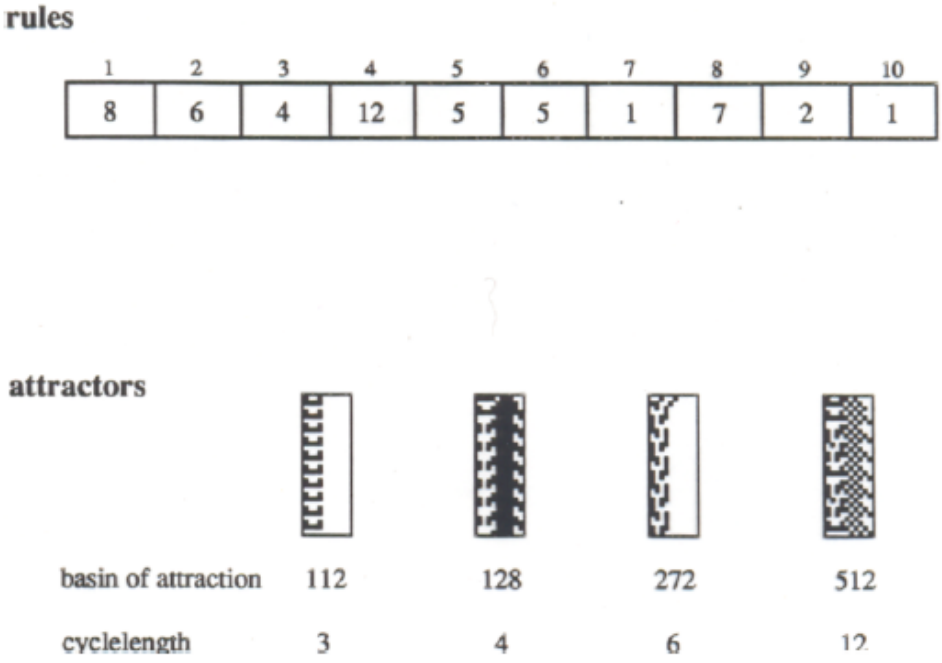


Figure 2.21: A boolean network represented in a CA. An extra "rule"-layer defines for every grid point which of the many possible transition rules it uses. Then, a space-time-plot can depict the trajectory towards the different attractors (bottom).

network of yeast. It is a complex and messy network when visualised. Moreover, this represents a very specific case, possibly with specific properties, that raises several questions:

1. How does this system behave?
2. How special is this system, and in what way is it special?
3. How did this system evolve to be the way it is?

In order to make some sense of this we can compare the observed specific network to a whole class of random networks, of which the behaviour has been characterised. Kauffman's approach to studying properties of gene regulation networks was based on studying the properties of random networks. This is somewhat similar to the transition rules in CAs: all 256 possible 1D CAs were tested to see their behaviour, and that is how the different classes of behaviour came about. He looked at many random Boolean gene networks, with differing amounts of inputs, connectedness, and transition rules, and derived several rules or common behaviours. He thereby had a baseline for such systems and could derive how that depended on parameters. These systems were tested with both synchronous and asynchronous timing regimes (remember, in CAs we saw it is important to test these different ways of updating), and they held in both cases (though patterns were stronger in the synchronous than asynchronous regime). Very generally, his findings were the following:

1. Multiple attractors: There are multiple domains of attraction, all leading to a different attractor. You could think of these attractors as different cell types that have different gene expression patterns.
2. Multiple causes: There are multiple ways to get to the same attractor. There is no *one* cause for a certain state of the gene network: there can be many with the same result. One example of this in neutrophils is presented below.
3. Robustness: A small change in the network (e.g. the deletion of a node, similar to a knockout) does not significantly alter the behaviour of the network.

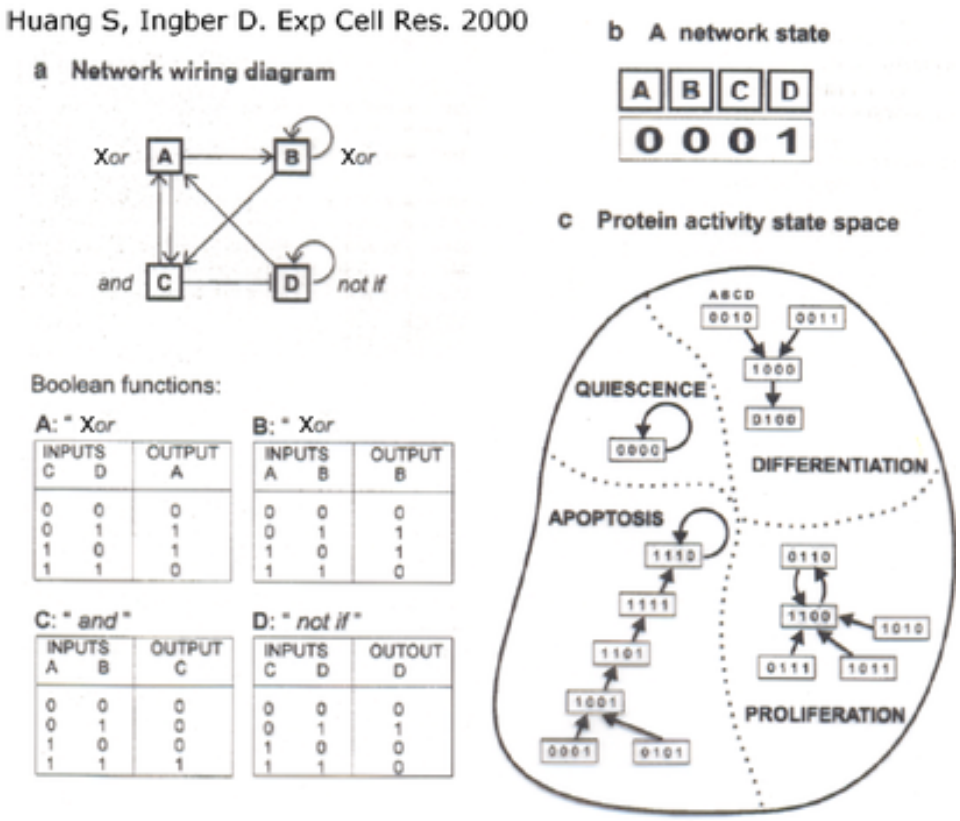


Figure 2.22: Modeling cell fate using boolean networks. (Huang and Ingber, 2000)

This means that random networks often harbour multiple attractors, each fed by domains of attraction. Moreover, the same signal can lead to different outcomes dependent on the state of the network, and several pathways can lead to the same attractor (cell state). In Huang and Ingberg the role of gene expression and the resulting signals is studied with respect to outcomes of cell states. By assigning boolean functions to a simple system of just four genes, different pathways leading to different cell states are modeled (Figure ??). This paper really brought attention to the potential of Boolean networks in the biological community. In another example, Huang *et al.* show two alternative routes of neutrophil differentiation in a high-dimensional state space(Huang et al., 2005) (Figure ??). This space consisted of 2773 dimensions (nodes), for 2773 proteins that could be involved. This meant n^{2773} different possible network states! This example shows that there are two completely different transients (regarding gene expression patterns) that lead to the same attractor: a differentiated neutrophil. This means that different signals can lead to the same eventual state. Importantly, it also means that a cell differentiating along one of the paths can respond much differently to a signal than a cell that is differentiating along the other path: these are very different cells in the transient period. This all goes some way towards guiding our intuitions. Since we know the generic properties of Boolean networks, we can have an expectation to find different attractors and domains of attraction, i.e. that different signals can lead to the same result.

We will now focus on **forcing structures**: Boolean functions that propagate an 'on-state' through the network. Some types of Boolean functions are insensitive to multiple inputs: if even one of the inputs is 1, the output is 1, regardless of the other inputs. Such functions are thus *forcing*: a single input forces the node to produce an output. The *OR* function is one such function. If only one of the inputs is 1, the output will be 1. Because such functions are insensitive to several inputs, these structures lead to **redundancy** and **robustness**: not all input information is needed for the output, and deletions (i.e. losing an input node) might not change the final result.

Interestingly, nearly 80% of yeast genes can be knocked out without any observable effects and even double

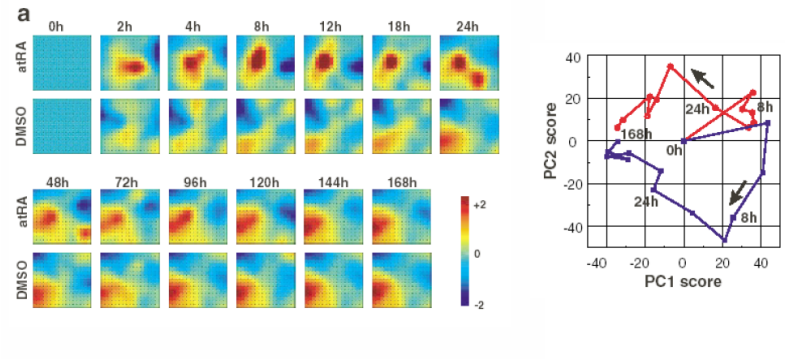


Figure 2.23: Neutrophils differentiation shows multiple trajectories to the same attrator

knock-outs can be invisible. Does this mean that yeast gene networks contain many forcing functions which are insensitive to changes via knock-outs? Hogeweg showed that in random Boolean networks only a small fraction of nodes tends to be functional, i.e. the network can be highly reduced(Hogeweg, 2000). This reduced subset is dominated by forcing functions. Such results are augmented by gene interaction studies which seem to find many false positives which are not actually functional. Given the role of forcing functions we should expect that most functional links are in fact non-functional with respect to the behaviour of the network as a whole. Moreover, research on the domain of attraction of the cell cycle shows that 85% of states goes to one mega-attractor(Li et al., 2004).

Kauffman's approach to Boolean networks was powerful in the sense that it revealed properties of a whole ensemble of cases. He was general. He did this by looking at changes in rules (similar to parameter sweeps in ODE systems) by generating networks with random connections where all nodes have the same connectivity or number of incoming connections (N = number of nodes, K = in-degree: the number of incoming connections for each node). In general, he found that random Boolean networks have many different attractors with a very long cycle length. Furthermore, these networks have a *low homeostatic stability*: small disturbances in the initial conditions can easily lead to convergence to a different attractor, and a high *reachability*: different attractors are easily reached by a small disturbance from a given attractor. However, his analysis also showed that the number of attractors depends on the connectivity, the least being found for $K = 2$. Moreover, cycle length, stability and reachability between domains also vary with connectivity, with shortest cycle lengths, lowest reachability and highest stability also being found for $K = 2$. The properties found for $K = 2$ most resemble what we would expect from biological networks. For this reason, Kauffman concluded that was the most likely number of incoming connections per node in biological networks.

Is it true that, as Kauffman thought, *connectivity* is the main determinant in these properties? In fact, Kauffman made special choices with respect to network types and finding $K = 2$ for this optimum is an artifact of his sampling. For binary Boolean functions, K=2 leads to the highest proportion of forcing functions in all possible functions. For K=2, the only two possible functions that are non-forcing are of the form 'exclusive or' or XOR, namely XOR→0 and XOR→1, which in turn leads to the results on robustness. With more connections a higher proportion of functions becomes non-forcing. What can XOR mean biologically? This means for example that A activates something, B activates something, but A and B together do not activate something. An analogue would be monomers that activate a gene, while dimers inhibit it. Because of the higher proportion of non-forcing functions like XOR, we find more domains of attractions and less stability. In fact Hogeweg (REF) studied the effect of the proportion of non-forcing (XOR) functions for $K = 2$ and showed that if all function are non-forcing this results in chaotic behaviour or long state cycles. If there were only forcing functions, however, one obtains single strong attractors. Stability is therefore not a result of a connectivity of 2 ($K = 2$), but the proportion of forcing structures, and such reasoning can be extended to biological systems.

This is an example of studying behaviour in a general system gone awry: Kauffman wanted to know general behaviour of Boolean networks, so he kept his networks as general as possible. However, his sampling inadvertently introduced a bias. Further on in the course, most models will be specific **paradigm**

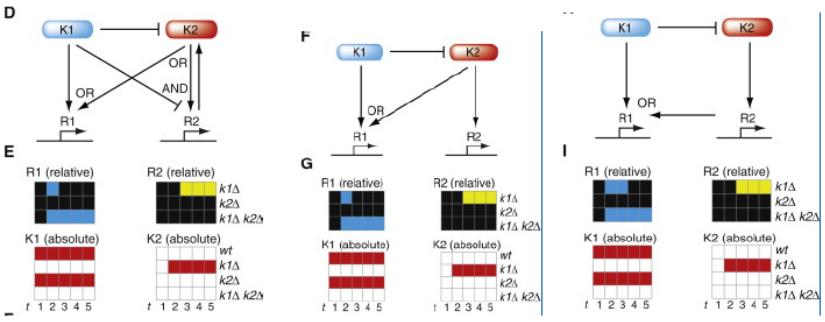


Figure 2.24: Backwards engineering a regulatory network on the basis of gene expression data.

systems, with general ground truths extrapolated from them.

Reconstructing a gene network

We have now studied what to expect from Boolean networks. With the influx of data in biology, we might wonder how to construct a network from data. As it turns out, translating gene expression data into a Boolean regulation network is not trivial. As an example: for 141 kinases and 38 phosphatases in yeast, the effects on gene expression of single and double knock-outs were measured (Wageningen et al., 2011). 60% of the single knock-outs did not lead to a different phenotype (a different phenotype was classified as >= 8 genes changing expression because of the mutation). Even in the double knock-outs, many buffering effects were found. These results show that there is a high level of redundancy (as we would expect) and that there are many **epistatic interactions**. Based on the knock-out data, the researchers tried to reconstruct the underlying regulatory network. However, even for small subsets of the data they found that there are several networks that can explain the same expression patterns (Figure 2.24). Hence, backward engineering of networks is *non-unique*! It is highly unlikely that you could reconstruct the "true" underlying regulatory network from a single set of gene expression data.

Conclusions (gene regulatory) networks

Taken together, we expect the following of gene regulatory networks:

1. Even simple networks have multiple attractors
2. There are alternative trajectories to the same attractor: there can be multiple causes for the same end point.
3. Domains of attraction: important for **state-change robustness**, i.e. how many states go to one attractor, and whether a change in state will change the final attractor. Different sets of nodes can lead to different attractors, it depends how you start.
4. Forcing functions are important for robustness in a different sense, namely **node removal robustness**. The more forcing functions, the less deletions impact a network's behaviour.
5. They are difficult to extrapolate from data: many different models can fit specific data.

Event-based models (Gillespie algorithm)

Until now, FSMs have taken center-stage, and we have shown how other modeling formalisms can be derived from them using short-cuts on conventions used in FMSs. We will now introduce **event-based models** and compare them to ODEs. As an example to see how they compare we use the logistic growth equation:

$$\frac{dN}{dt} = aN - bN^2$$

Or, alternatively, if we do not go by real population numbers but rather by a carrying capacity (see ODE introduction):

$$\frac{dN}{dt} = rN(1 - \frac{N}{K})$$

These equations result, just like in ODEs and MAPs, in simple logistic growth, where growth is fast until competition starts to dominate, and then levels off until it stops completely. We now consider the history of the interpretation of the terms of this classic equation. For instance the N^2 term can be interpreted as representing competition. Alternatively in the case of dN/dt = rN(1-N/K), K is the carrying capacity, which is a concept which was derived from the population concept. These interpretations were in fact derived later, while the original interpretation was much more arbitrary and stems from a conference called by king Leopold I of Belgium (the nice man who was personally responsible for slaughtering half of the Congo's inhabitants between 1885 and 1908(Twain, 1905)) to discuss the implications of Malthus' prediction about the human overpopulation (population doubling every 30 years!) as derived from British parish records and North American population growth numbers(Malthus, 1888). At the conference, Verhulst, a physicist, put forward a simple solution to the problem and was so able to calm the poor king's fears. Verhulst used the argument that any function (including a growth function) can be approximated by a Taylor expansion , and with a little reasoning a reasonable human population growth function can be derived from $\frac{dN}{dt} = a + bN + cN^2 + ... + nN^y$. The first term can obviously be dropped because there is no external influx into the human population. Moreover there should be at least one additional term other than aN and it should be negative, otherwise the population would grow to infinity, which is obviously a physical impossiblity. And so, obviously there was no reason to fear for infinite overpopulation! Notwithstanding this reasoning, it still depends on the *coefficients* whether overpopulation is alarming or not! The main point here however, is that the interpretations of model terms can be very minimalistic indeed.

We can model a stochastic birth/death process using a stochastic ODE:

$$\frac{dN}{dt} = aN - bN^2 + noise$$

In the ODE-formalism, the variables change continuously, i.e. something is happening at every time point. However, we can also assume that things only happen at certain times, as events. In this event-based formalism, each event is modeled using probabilities which determine which event occurs when in continuous time. This is called the Gillespie algorithm(Gillespie, 1977). Since we use probabilities for the events, this automatically gives a *stochastic description* of the system.

In this formalism, each term needs to be explicitly interpreted:

1. aN = birth + death
2. bN^2 = extra death + reduced birth due to competition

Now, assume that $a1$ is the per capita birth rate, $a2$ the per capita death rate, $b1$ the reduction in births due to competition, and $b2$ the extra deaths due to competition (hence $a = a1 + a2$, and $b = b1 + b2$). There are two possible events that can take place: birth events and death events. Then, the frequency of all events (birth and death) is:

$$E0 = (a1 + a2)N - b1N^2 + b2N^2$$

The time until the next event is stochastically determined as:

$$\tau = 1/E0 * ln(1/rand1)$$

where $rand1$ is a random number between 0 and 1. Hence, on average there is a waiting time of $1/E0$, but this varies stochastically. Lastly, we have to determine whether the next event at time $T + \tau$ is a birth or a death:

1. $N \to N+1$ (birth) if $(a1N - b1N^2) < rand2 * E0$ where $rand2$ is another random number between 0 and 1.
2. $N \to N - 1$ (death) otherwise.

The behaviour in this model allows for stochasticity which is comparable to the chaotic regime in MAPs (in ODEs such behaviour never occurs). However in the MAP the direction of change in the chaotic regime is always to the other side of the equilibrium (there is an overshoot to the other side every time). In the event-based formalism the direction is not necessarily relative to the equilibrium. This type of noise can be considered to be more realistic for stochastic simulations where once in a while something happens.

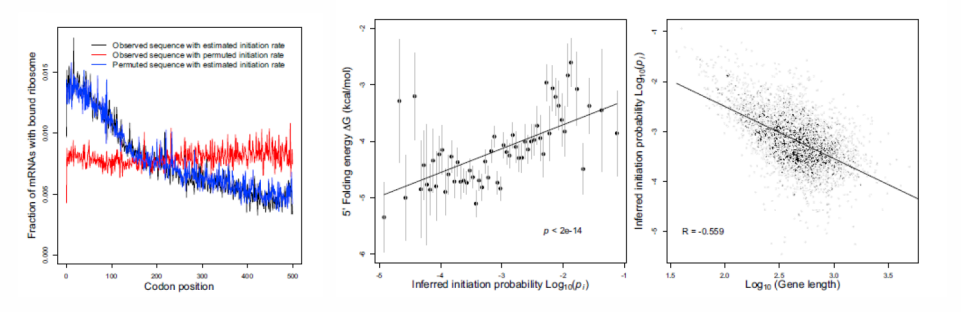
This all might sound complex and -who knows- even interesting at times, but how can we use this? Why would we use such a stochastic model? The Gillespie algorithm was used to investigate what the rate-limiting step in yeast protein translation is(Shah et al., 2013). The question was whether protein production was initiation- or elongation-limited in exponentially growing yeast populations. To investigate this question, all the known data was brought to bear upon the issue: the fasta file containing all yeast mRNAs and the number of RNA molecules per cell, the number of yeast tRNAs in the cell and their wobble behaviour, the number of ribosomes, the initiation probabilities of all mRNA types, the sizes of ribosomes and tRNA's, the diffusion constant of ribosomes and tRNAs (called the *characteristic time*), etc.

The states in the model were manifold: the free numbers of ribosomes and tRNAs of every type and the position of bound ribosomes and tRNAs on the individual mRNAs. The events in the model were inititation (binding of a ribosome at the 5' end of mRNA) and elongation (change position of the ribosome if the next one was free, bind a new tRNA).

In pseudocode, the algorithm worked as follows, for each time step:

1. calculate the fraction of initiatable mRNA (no ribosome attached to first 10 codons)
2. calculate the number of elongatable ribosomes waiting at position j (RB(j)): ribosomes on the mRNA where the next 10 codons were not bound by another ribosome.
3. Rates of all possible events:
   (a) total initiation rate, normalised to the diffusion rate
   (b) total elongation rate: keeping the wobble behaviour and a sort of proofreading in mind, normalised to the diffusion rate
4. calculate the probability of each possible event
5. select an event based on its probability of occurence
6. update the changes in the state of the cell
7. increment time until the next event
8. update the number of free ribosomes
9. update the number of free tRNAs

The experimental observation was that there were more ribosomes at the 5' end of mRNAs. Apparently, the ribosomes could not progress, so initiation seemed easy, whereas elongation must be difficult. In the model, the same observation was made, but it was clear that this was not due to elongation (Figure REF). Why? If they randomised the mRNA *initiation probability* they lost the signature of elongation dependence. However, it was *initiation probability* that they altered, so what gives? If the nucleotide code of all mRNAs was randomised, the signature was still there. As it turned out, the curve from experimental data that led researchers to believe it was elongation-limited was averaged over all mRNAs. If many ribosomes are on short mRNAs, then you will see them often at the 5' end, because they have relatively more of it. Short mRNAs are also less likely to fold in on themselves, and are therefore more easily accessible (though this is not so by default, and might have been selected for). Thus, the model debugged the assumption that protein production was elongation-limited.



Thus, this event-based modeling technique is an interesting addition to the model family. You only deal with time points at which events actually happen, lowering computational load and making such a data-intensive example as the above possible. There are no problems with infinitely small or large populations (discrete variables). However, there are fixed conditions, a fixed number of molecules, and, most importantly: there is space, but no spatial structure. Space is implemented by diffusion and molecule concentrations, but everything experiences an 'average' space. This is problematic, as we shall see further on.

Conclusion and overview of modeling formalisms

We have now spent quite some time talking about modeling frameworks, using the FSM as an archetypal model and branching out from there. We have considered many different combinations of discrete/continuous time, space, and variables (see table 2.1). Let us finalize by zooming in on the behaviours, properties, and peculiarities we have seen.

Table 2.1: Modeling formalisms and their assumptions on space, time, and the modeled variable(s)

Formalism	Space	Time	Variable
ODEs	-	continuous	continuous
MAPs	-	discreet	continuous
CAs	d	discrete	discrete
PDEs	c	continuous	continuous
Map lattices	d	discrete	continous
Meta-pop models	d	continous	continuous
IBMs	either	continuous	discrete
Event-based	either	continuous	discrete

FSMs, CA models, mesoscale entities, and timescales

We have looked at FSMs as a prototype modeling system. They need fully defined inputs, outputs, and a unique next state function for every possible combination of states. We have seen various short-cuts to this completely defined formalism. ODEs solve the problem of completely defining all interactions by making the next state function a mathematical function, and defining continuous populations and continuous time. The need for a unique next state function is relaxed, because the function is valid for all real numbers. MAPs are similar, but use discrete time.

The CA solves the arduous task of defining a full transition table by restricting itself to FSMs that can be modeled by local interactions, a situation that is often valid for biological systems. Small FSMs that take into account only the states of their direct neighbour and themselves have the same next state function, but outcomes differ due to different local states. Non-predefined *mesoscale patterns* can arise that have behaviours that can only be described in terms other than those used to define the model and those that describe the macroscale. They have a dynamics of their own, and are often important to the behaviour of the system. Cataloging the different types of mesoscale patterns can give greater insight, as with the CA with rule 54 and the particles observed.

We have seen that pattern is not unexpected, but rather is the expectation: it is hard to be well-mixed (think of the lymph node)! We have also seen that randomness or noise can lead to clearer patterns

(majority vote). Additionally, simple rules can lead to very intricate patterns and completely unexpected (modulo prime) and/or unpredictable (game of life) results.

Besides naturally emerging higher levels, a higher level can also be explicitly defined. This does not give insight into how a system could produce the higher level, but does shed light on what happens once it is in place. The Cellular Potts CA model works by defining a higher level cell entity that occupies multiple (sufficient) grid points. It works by free energy minimisation, and different cell types having different interaction energies and target volumes. In this formalism, cells can squeeze past one another, and cell sorting behaviours can be modeled. The importance of cell grouping in moving towards chemotactic gradients, and the effect of space and crowdedness on T cell movement was discussed in relation to this formalism.

The relative timescales of events in a CA are of great importance, and should be thought about when modeling. If the time step modeled is small, picking random neighbours and having the neighbour picked move into a spot is fine. If, however, the time steps are larger, one might wish to use a claim mechanic, whereby all local neighbours compete each time a grid point is updated. This more accurately models, for example, what might happen in a forest, where saplings in an open area all compete, and over a few years, a tree grows. One can also introduce the chance of nothing at all happening. If species A has twice the fitness of species B, but there is a chance of nothing happening, the theoretical benefit of species A over B is diminished: if something always happened it could exert its fitness advantage 100% of the time. Scaling of the timing in a CA is thus very important, and there are many implicit assumptions in what mechanisms you use for the next state functions. It is important to consider this.

Boolean networks, event-based modeling

Besides the FSM formalism, we have looked at Boolean networks. We have learnt to expect that networks, even simple ones, have multiple attractors, with differing domains of attraction. We also know that, due to forcing functions, networks are often resistant to knock-outs, because not all inputs are needed to get to certain attractors. Often, there are different paths to reach a certain attractor. There is no easy one-to-one cause-effect relation. Multiple network topologies can explain the same experimental data, so reconstructing a network from data is difficult.

In event-based modeling, we considered a probabilistic framework, where we only model events that actually happen. One categorises all possible events, relative chances of them happening, and then advances time until an event happens. There is thus discrete time. Taken together, this makes whole-cell modeling feasible. As we have seen, space can be included, but only *average space*, not locality.

What are models?

In the previous sections, we have discussed various modeling formalisms. So, what is a "good" model? *A good is a model of B, when by studying A we can (or hope to) learn something about B.*

What does this mean? Firstly, to make a model of B, one needs a mapping of features of B into variables/states in A, such that it satisfies (or partly satisfies, given short-cuts) the unique next state function one needs to define in a dynamical system. Different model formalisms can play different roles: CA have patches of space that can be filled with individuals. MAPs and ODEs have a fixed set of continuous variables and populations (concentrations) of organisms (molecules).

Furthermore, in this course, we use the following requirements for or characteristics of a good model:

1. A model should **not** be as general as possible. Ashby wrote a book on cybernetics, in which he asked the question: 'To what degree is the Rock of Gibraltar a model of the brain?'(Ashby, 1999). The answer might come as a let-down: 'It persists, so does the brain; they are isomorphic at the lowest level'. The crux is, of course, that the Rock of Gibraltar is not a good model for the brain at all, though *in an extremely general sense* it is.

2. A model should **not** be as similar as possible. In that case, A would be the best model for A. We build models to gain insights into core/partial workings of a system, or to debug certain

(experimental) inferences. The lymph node model that investigated B-cell clumping does not purport to accurately model the complexity of a real lymph node, yet it showed us that clumping is suboptimal and the natural/simple state (see CA as a paradigm system).

3. A good model considers first: what needs to be modeled? What are the patterns that need explaining? For that, two important questions are how generic or special the patterns are. You can investigate whether something is a special case by checking its robustness, and seeing whether it pops up in other modeling formalisms. If a behaviour shows up in different modeling formalisms and is robust to changes in timing, tweaks to spatial scales, or (as in networks) deletions or knockouts it might be very important. Generic properties can be characterised by checking behaviour under many different parameter regimes and then classifying these behaviours. The order parameter (λ) is an example of this.

Chapter 3

Behaviour and doing what there is TODO

Introduction

Herbert Simon, the founding father of artificial intelligence said (Simon, 1969):

"An ant as a behavioural system is quite simple. The apparent complexity of its behaviour is a reflection of the environment in which it finds itself."

which we can paraphrase to:

"A human as a behavioural system is quite simple. The apparent complexity of its behaviour is a reflection of the environment in which it finds itself."

The basic idea here is that given enough individual-based diversity, individual versatility becomes expressed when individuals meet different circumstances. Thus while an ant may want to walk straight to get back to its anthill, its actual meandering route home is a reflection of the obstacles it encounters in the environment. In this light it is also interesting to consider apes that grow up in environments with people where they can do all sorts of human-like things, but if you go into the field one observes them mainly sleeping and eating. This suggests that apes have excess capacity that is not used often in their natural environment. How should we understand this in an evolutionary sense? On the one hand, finding food may be harder than we think. On the other hand, doing easy things *well* may allow you to do hard things *a bit*. In any case, the environmental structuring plays an important role.

The themes we address in this section are:

1. Self-organisation
2. How simple rules can generate complex behaviour

So far, we have looked mainly at a CAs, which have space as their core entity. We now take a look at full individual-based models.

Individual-based models

Individual-based models (IBMs) are based on individuals in space, rather than space with individuals as in the CA. Their properties are as follows:

1. Individuals are simple (in)finite state machines
2. Individuals are located in space. This space is not necessarily discretised like in a CA: space can also be a continuous variable.

3. Individuals interact with their (possibly complex) environment, and with other individuals in the environment.

4. Individuals, in principle, keep the same behavioural rules. However, they can change their environment, and thus the input they experience.

The TODO principle

Inspired by Herbet Simons' "reflection of the environment", Hogeweg and Hesper conceptualised the TODO-principle (**?**Hogeweg and Hesper, 1991). Herein, they placed the focus on:

1. "Do what there is to do!": this principle emphasises that behaviour is steered by local information: only when food is observed will food-gathering behaviour actually be triggered.

2. "Do based on what is done!": the local information the organism perceives (its environment) is shaped by its doings. Food that is eaten is no longer there, prompting a halt to feeding behaviour and return to foraging/roaming. There is thus external memory for behaviour, which is the environment that is created by that behaviour (collective memory for all individuals in a shared environment). Changes in the environment brought about by organisms will inform their behaviour: they do based on what is done.

Using this framework, we can observe that flexible behaviour arises from rigid rules. Furthermore, we might expect to see **automatic adaptation**: behaviour will not be adapted to something that is not there, i.e. no nonsense behaviours will arise due to this if-then coding of behaviour. This stands in contrast with common assumptions in behavioural models about optimal behaviour where the environment is often not taken into account. This can lead to rather ridiculous analyses since assumptions are made that individuals do things without cues from the environment.

Rodney Brooks' robots

A first example of opportunity- vs optimality-based behaviour are the robots of Rodney Brooks. Rodney Brooks designed state of the art robots which were clever and could make plans. Such robots would think a long time to make a plan for an optimal path through a room. However, the robots would take so long that, in the mean time, the room might have changed, in which case the robot would have to rethink everything. In other words, if one needs the best model (i.e. to plan a path), the best model is the world itself. In that case, it is better to generalise and solve problems on the way (i.e. start walking and change direction if you bump into something). Thus using continuous feedback from the environment is a good planning tool. This is what he used in later work, producing robots that could efficiently navigate office environments(Brooks, 1991). In this way we can see TODO as an alternative explanation for behavioural patterns which are normally explained in terms of evolutionary optimisation. The outside world directly informs what is good behaviour.

Self-organisation in moving groups

An often-used example of how simple individual rules can lead to complex behaviour is self-organisation in groups of animals. Much studied prototypes include the flocking of birds (e.g. Boids, (Reynolds, 1987)), schooling fish (e.g. (Hemelrijk and Hildenbrandt, 2012)) and migrating herds (Couzin et al., 2005). These models usually consider a simple inert (unchangeable) environment, in which at most the physics of the environment are included.

Boids (bird-like objects)

The Boids are a classical example of individuals with simple grouping rules in a simple (initially empty) environment(Reynolds, 1987). The Boids follow three behavioural rules, which are depicted below:

1. Repulsion: If other individuals are too close to you (within repulsion zone), you move away from them.

2. Alignment: Adjust your direction to match the average of your neighbours (within alignment zone).

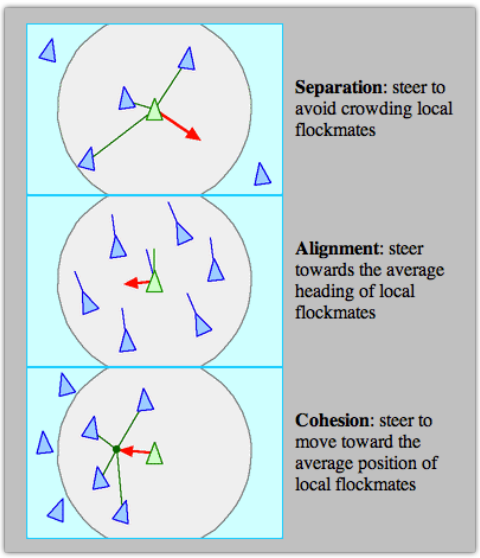


Figure 3.1: Three simple rules cause flocking very similar to *e.g.* flocking of sparrows

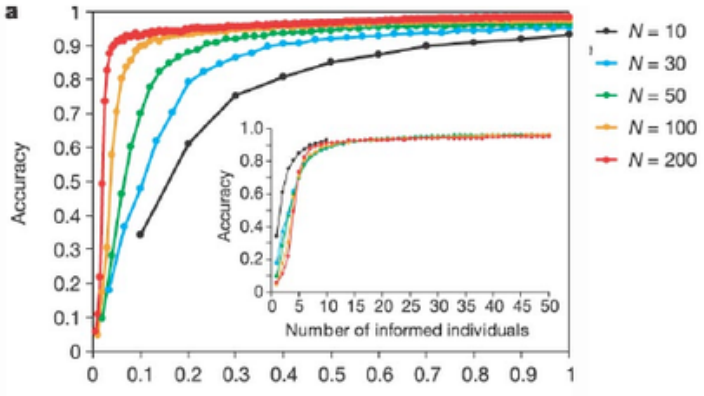


Figure 3.2: Couzin et al. (2005)

3. Attraction: Move towards other individuals, as long as they are not too close (within attraction zone).

These three simple rules cause a flock in an empty environment to all move in a certain direction (see Figure ??). Moreover, if obstacles are added to the environment, the flock can change direction and break up and reform. Hence, these simple rules can cause quite complex behaviour when combined with a non-trivial environment.

Decision making in flocks and schools

Couzin *et al.* also use the rules of Boids to describe flocks of animals (Couzin et al., 2005). They then study how well these flocks can move to a certain target, e.g. a food source, if only a small amount of the individuals knows where to find this target. These informed individuals have directed movement towards the target, while the other individuals just follow the regular Boids rules. Surprisingly, they find that you only need a very low fraction of informed individuals to accurately find the target (Figure ??). Moreover, the larger the group, the lower the fraction that you need. Hence, especially in large groups, you only need a small amount of information to find a target.

Lastly, they studied the effect of two groups of informed individuals that have different targets (e.g. half of the informed individuals preferentially moves to the left, the other half to the right). Depending on

the parameter conditions, this leads to one of two outcomes:

1. Averaging: The flock ends up somewhere half-way between the two targets. This happens if the directional movement of the informed individuals is relatively weak, and/or if the two targets are sufficiently close to each other.

2. Winner takes all: The flock ends up in either one of the targets, i.e. one group of informed individuals wins. This happens if the informed individuals have strong directional movement, and/or if the two targets are far apart.

Flocking with physics of the environment

In the example above, the individuals are basically assumed to float in space, without forces acting on them. However, animals are subjected to many physical forces that constrain their movement possibilities. For instance, flying birds have to maintain a certain speed to avoid falling from the air because of gravity, and animals moving at speed in a certain direction cannot immediately turn around because of their inertia. Hemelrijk and Hildebrandt developed a model of flocking starlings that incorporates these constraints(Hemelrijk and Hildenbrandt, 2012). However, other than these physical constraints they only implemented Boids-like simple grouping rules. The behaviour of the starling flocks in their model closely resembles observations of real flocks of starlings, again illustrating that a simple set of rules and obvious physical constraints can explain behaviours thar are perceived to be (very) complex. Furthermore, Hemelrijk and Hildebrandt have used their model to test the influence and relative importance of model parameters such as locality of the interactions and speed variability, "experiments" that could never be done in nature. The latter is an obvious example of the utility of models.

Subconclusion

All these models deliver a wow-factor upon first impression. They show the power of what simple behavioural rules can do, given a relatively simple environment. Next, we will study what happens when we combine simple rules and complex environments, which can be changed by individuals.

Social structure and grouping in simulated chimpanzee-like entities

Here we take a look at social structure. In particular we will look at primates (chimpanzees), in which the following things have been observed by Harcourt (Harcourt et al., 1992):

1. There are all-male groups, but (almost) no all-female groups.

2. Females live solitarily much more often than males.

3. Males travel further than females.

We will look at a model to study this social structure. However, we will not pre-specify that chimps want these types of behaviours. Instead, we ask whether the patterns we observe could be *side-effects of things that individuals have to do anyway*. Thus, our modeling strategy is one where we do not implement the behaviour we are interested in, but rather a minimal structure of the system that we are interested in, and then *observe* which behaviour emerges.

To this end, Boekhorst and Hogeweg made a model of CHIMPs(Boekhorst and Hogeweg, 1994), where individuals:

1. Go to the nearest fruit tree and eat until they are satisfied or food has been exhausted.

2. May take other CHIMPs into account when looking for food.

3. Want to mate: males check for receptive females.

4. If they are female, need more protein (for reproduction), so they eat protein-food that is not eaten by males

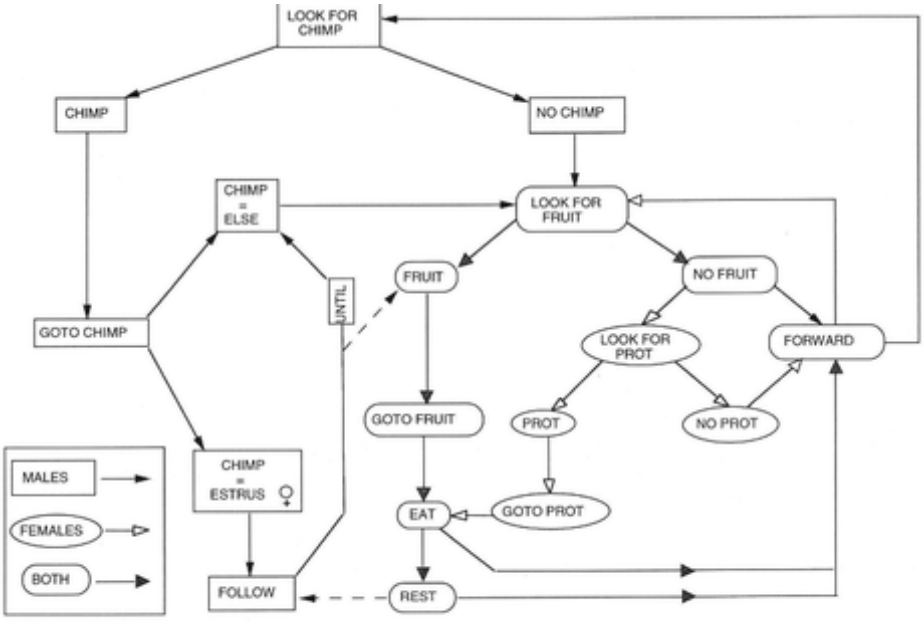


Fig. 1. The behaviour of CHIMPs. The dashed arrows indicate that a MALE companying an estrus FEMALE feeds from the FRUITING TREEs he encounters while following the her. After having rested, he joins the FEMALE again (if she is still in estrus). For further explanations, see the text.

All this takes place in an environment with FRUITs and PROTs: estimates were made on how much fruit and protein is available in real habitats, and in which chunks the food is available (in terms of chimpanzee-hours: the amount of time a single chimpanzee could feed on it) (Figure ??).

So, what were the results? They were striking:

1. There were random paths of movement and individuals ate the nearest food.
2. Various groups formed with interesting compositions.
3. There were clearly more lone females than lone males.
4. There were more all-male groups than all-female groups.

These observations are very similar to the observations in real chimpanzees!

Originally, the occurrence of all-male groups had been explained by the idea that males band together to defend their territory and engage in warfare with neighbouring groups. This was thought to be both functional and adaptive, and and advanced feature (i.e. closer to humans than to other animals). However, we see the same in the simple CHIMP model, with even more exaggerated male grouping (Figure ??).

However, we can now see that chimpanzees need not want to be in groups (males) or solitary (females). Rather, the social structure is an **epiphenomenon**: this phenomenon arises as a side-effect from other rules. In terms of speculation about the purpose of grouping, it is nonsensical to think that this structure is advantageous and should give a higher fitness; it just happens. Note that once it arises such an epiphenomenon might be further exploited by evolutionary forces: maybe in real chimps there are extra factors involved, but we don't require it to see the same behaviours arise in CHIMPs.

Why does this epiphenomenon happen? Well, the food occurs in clumps and individuals move towards the nearest food source, which can generate grouping. Moreover, males want to check for receptive females, and may move to the *same* female, which enhances male grouping. Females, on the other hand, need PROTs that are more easily depleted: they thus split up more easily in search of protein food sources (similar groupings occur in orangutans for much the same reason, see the article).

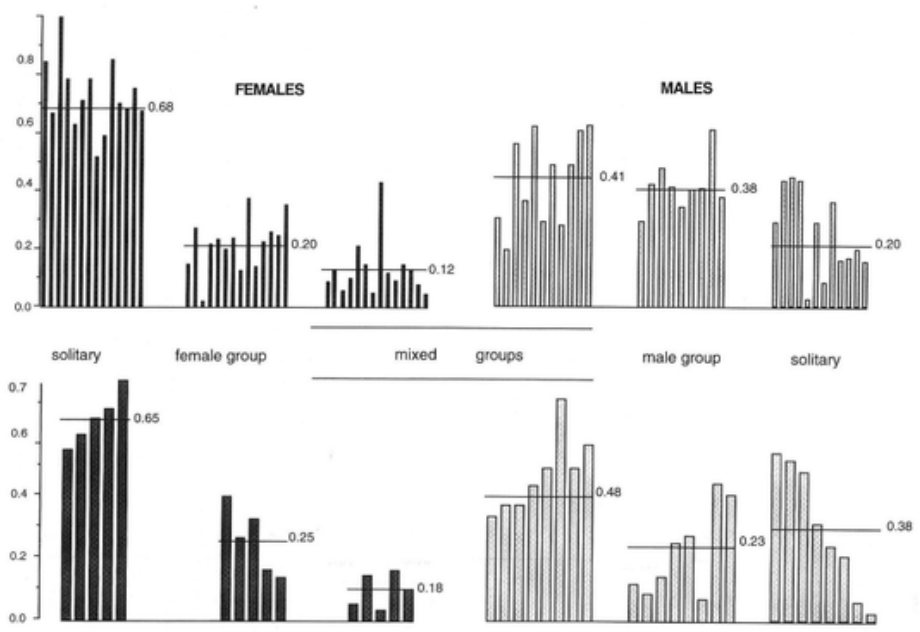


Fig. 4. Proportion of times female and male CHIMPs (above) and chimpanzees (below) spend in parties of different composition. The chimpanzee data are from Halperin (1979). Mean values are indicated by the horizontal lines.

So far, we have shown *one* matching pattern (*model observable*) between simulations and real life, but that is of course quite minimal. However, several other observables also show a good correspondence between the model and real chimpanzees, making a much stronger case:

1. In terms of walking distances, males walk much further than females in the model. This is because males are more often in groups, which deplete food sources faster, and move on faster.

2. In real chimps, males walk about 4km/day while females on average 2.7 km/day(Goodall, 1986).

The model is actually able to *quantitatively* reproduce these results. The moral of the story is *not* that chimpanzees are stupid and only act based on simple programming. However, some observables are so basic that one cannot infer from them that chimpanzees are not stupid. Given that you perform certain very basic behaviours (eat, follow mates), other behaviours will emerge as side-effects or *epiphenomena*. In that light, taking an observed feature and then trying to find out how it might be optimally evolved is misleading: it might just be a side-effect. In this example, taking the observed grouping of chimpanzes and trying to explain why this is great for them wasn't necessary. We can use models to study how different behaviours are related to each other, and how some may emerge as side-effects of others.

Subconclusion

1. We have seen that the TODO formalism means that behaviour is determined by local information (opportunities) and that behavioural patterns can arise as side effects

2. We have seen a contrast between opportunity-based and optimality-based approaches. The latter largely ignores the opportunities, whereas many behaviours could be explained as side-effects, i.e. they are a consequence. In that sense, there is *automatic adaptation* (there is no need for an adaptive process to not eat what is not there!)

Now, we will look at TODO in combination with internal change within individuals (memory/learning).

DODOM interactions in bumble bees

In social insects, there is a clear division of labour between the queen and the workers. Looking at the differentiation of labour in honey bee colonies, Whitfield *et al.* found the following by studying genome data(Whitfield et al., 2006):

1. Gene expression is age-dependent
2. Variance in expression is already seen at low dimensions (as found in a **Principle Component Analysis (PCA is explained in the Glossary section in the back of this reader)**)
3. Only the 3rd PC showed differentiation between species (i.o.w. differences in gene expression between species were smaller than those within species!)
4. There were similar patterns across species
5. The newborns differ the most from the other individuals
6. There is differentiation, but some bees become foragers sooner than others: could this be TODO dependent?
7. There is no clear one-to-one mapping of gene expression, age and behavioural patterns

This example illustrates that there are many factors - both internal and external - involved in differentiation and division of labour. Thus, even if we can explain things with TODO mechanims, there is probably a lot of feedback from what is done: reinforcement of the behaviour. In the next example we study the effects of reinforcement on behaviour.

Intro and experimental work

Here, we look at a model of bumble bees and their social interactions, made to study the life-cycle of bumble bee colonies. Some remarkable features of this life cycle are:

1. Bumble bees are not truly eusocial, in the sense that the queen can and under some circumstances does still leave the nest to forage herself.
2. Only the queen survives the winter and starts a new nest on her own early in spring.
3. The first brood gives rise to workers.
4. When there are enough workers, the queen stays in the nest and only lays eggs.
5. At the end of the season, the queen lays special eggs that when fertilised produce queens, while unfertilised eggs produce drones (males).
6. Shortly after, the workers rebel and expel or kill the queen.
7. After this rebellion, some workers lay eggs (the so-called elite workers).
8. These eggs are unfertilised and produce drones.
9. New queens and drones mate to start the cycle again.

This cycle of course raises many questions, for instance: *who are these elite workers which lay eggs*? It would seem that if this were a heritable trait, all workers should lay eggs! To study these issues van Honk and Hogeweg, and Hogeweg and Hesper, studied the social interactions patterns in real (in the lab!) and artificial (see next section) bumble bee nests(van Honk and Hogeweg, 1981; Hogeweg and Hesper, 1983, 1985). In the nest, bumble bees have pair-wise interactions. When 2 bumble bees meet, they antennate (make contact via their antennae). After the interaction, one goes straight (dominant) and one gives way (subordinate). The queen always goes straight. Hence, there is some assessment of other bumble bees through a dominance interaction. But how to study how this works exactly? This is difficult, since the interaction matrix of all bumble bees will only be sparsely filled after experimental observation: not all bumble bees will interact with all others many times, so that clear patterns are visible. Also, the question was whether these interactions would change over time.

As a solution, van Honk and Hogeweg used a cluster analysis in order to reveal the underlying social structure. They obtained a matrix of interactions containing how an individual behaved towards all

others: how often it won in total, and how often it lost in total. The reasoning was that the elite would probably behave differently towards most others. They considered two bumble bees to be similar if they interacted with all other bumble bees in a similar way. Furthermore, they subdivided their experimental data set into periods (while the composition of the nest stayed more or less similar).

The results of this meticulous experimental work on bumble bee interactions were as follows (Figure **??**):

1. In the early period, the colony is populated by the queen and some workers.
2. In the middle time frame, a group of elite workers has emerged, that becomes more accentuated over time:
   (a) Once an individual is in the elite cluster, it remains there.
   (b) These are not simply the oldest workers: being elite is not age-dependent.
   (c) Individuals can join the elite quite late in life.

   When the queen is killed: a pseudo-queen emerges (highest in dominance among all elite workers) which, together with other elites, starts to lay eggs.

Thus, we see a social distinction in bumble bees: there is a distinct group of elite workers that cluster with the queen (red rectangle). Furthermore, it is this elite which lays eggs at the end of the season. However, a weak point of the study was that it relied on data of only one (lab) nest. When this experiment was repeated in Germany, the division between elite and common workers was less clear and less consistent over time(van Doorn and Heringa, 1986). These nests also grew faster and became bigger. We could repeat the experiment *in vivo*, but we can also use simulations to study how this social structure can arise.

DODOM in bumble bees: model

Hogeweg and Hesper formulated a BUMBLE model to study how the social structure in the bumble bees can arise (Hogeweg and Hesper, 1983). They modeled the interactions between BUMBLEs via the hypothesised **DODOM** rule:

- There is a winner-loser effect in interactions between bumble bees: once an individual wins an encounter this increases its propensity to win following encounters.
- This works via a damped positive feedback (once you are winning a lot, your propensity to win will rise less quickly than when you first start winning), which maintains sensitivity to external conditions.
- Individuals have an internal dominance parameter D. When two BUMBLEs meet, they can observe the value of the other's D parameter.
- They then have a dominance interaction (Do Dominance; **DODOM**) where the probability to win is: $p\_win = D\_me/(D\_me + D\_you)$
- If you win an encounter ($randomnumber < p\_win$) then $K = 1$, else $K = 0$
- Your updated dominance parameter is then: $D\_me_{new} = D_{me} + a(K - D\_me/(D\_me + D\_you))$
- Thus the update depends on the difference in dominance (like chess rating or ELO you might know from online games): you get a high boost from an unlikely win, while only a small boost if you were likely to win. This is a damped positive feedback loop.

The other assumptions of the model were as follows:

1. The population dynamics of bumble bees was taken as a given.
2. All workers were *created equal* with no propensity to become elite.

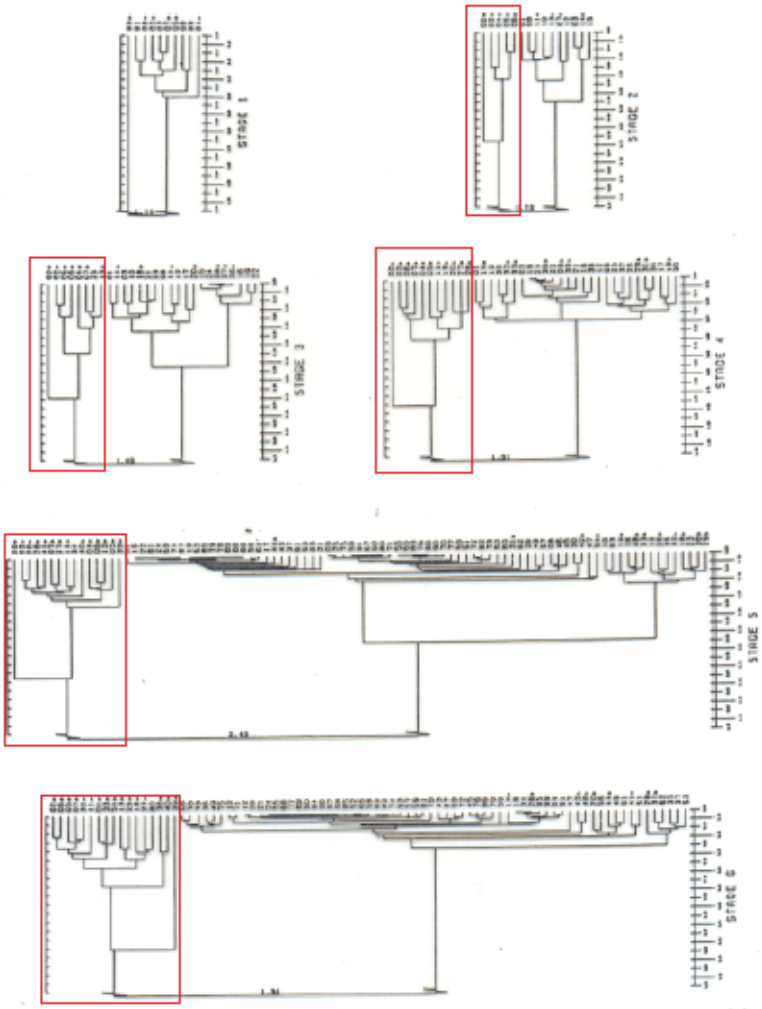


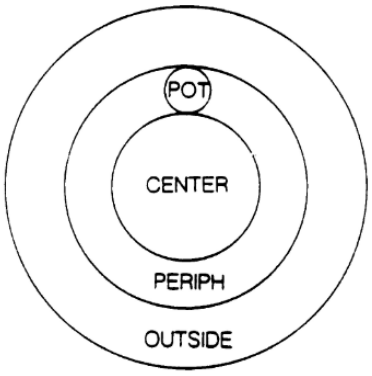
Fig. 3. Cluster analysis. The vertical scaling is identical (0–1.1) for each of the stages. The level at which the elite group and the common workers are finally linked is indicated along the connection line. The elite group recognised by this method contains only a subset of the egg-laying workers

Experimental Data

Figure 3.3: Cluster analysis of bumble bee dominance hierarchy

3. The TODO was as follows: feeding, foraging, egg-laying on nest.

4. The environment was given by the nest: it had a center and a periphery, including a food pot (Note that there was no real space in the model, but only compartments).



5. The model was event-based and actions took time.

6. Social interaction between random bees within compartments via DODOM (queen versus workers, and workers versus workers).

7. Initial dominance:

   (a) Queen initially gets D = 7.5

   (b) Workers initially get D = 1

   (c) The D parameter determines the location and activity of the BUMBLE (periphery vs center).

   (d) The queen is killed when she looses a number of dominance interactions in a row.

   (e) The model was then observed in the same way as the real nests.

An important point from above was that the main analysis was carried out exactly as in the real nests: counting the number of won interactions against all BUMBLEs, and clustering over time. The results were interesting (Figure **??**. Elite workers arise in the model, and are relatively constant in time. All workers try to lay eggs. The queen is expelled at a certain moment, and the timing of this event is the same in all replicate simulations. Hence, our first conclusion should be that a heritable predisposition to become an elite worker is *not necessary* to explain the social structure in bumble bees. Rather, we observe **emergent division of labour**: labour division arises out of simple behaviours. That still leaves the confounding results from the study of a nest in Germany mentioned earlier. There, the division between elite and common workers was less clear over time. It seems that the consistency of the elites is thus dependent on group (nest) size. In the model, we can study nests with different size by varying the *egg laying speed*. This yields the following results:

1. For differently sized nests the queen is still killed on about the same day!

2. Slow small nests have more differentiated D parameters (and thus a clearer division between elites and normal workers), fast nests have less differentiated D values. To understand this: in larger nests, there are more workers with low/average D values. If a to-be elite worker consistently meets workers with low/average D values, its highly likely win only allows it to gain a slight bit of dominance. In small nests, the difference quickly becomes more pronounced.

3. This result depends on how workers develop dominance.

4. This self-stabilises on nests of a certain age.

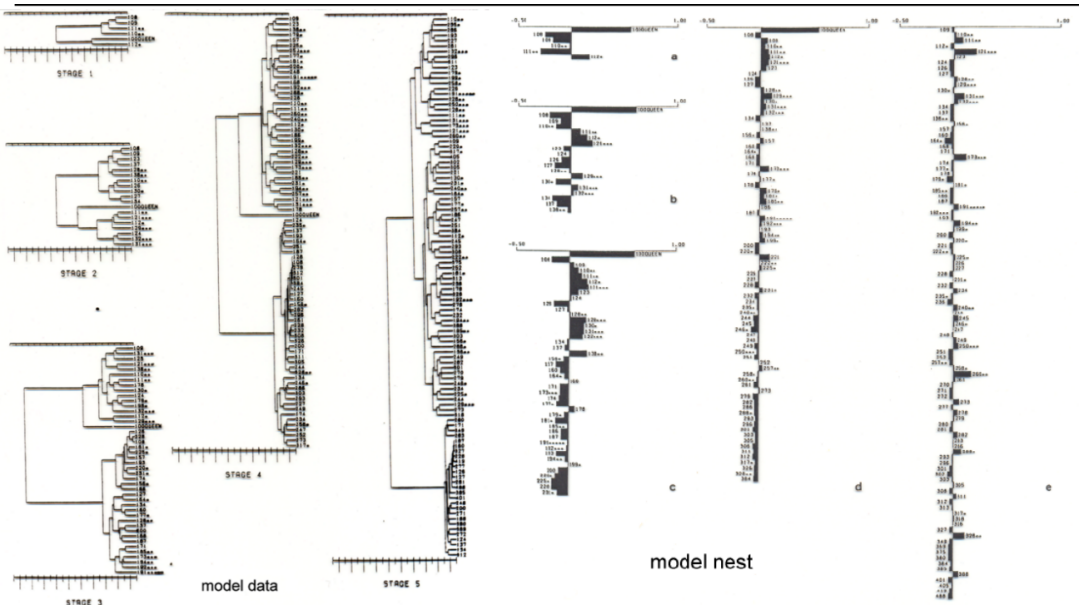


Figure 3.4: Cluster analysis of BUMBLEs (modeled bumble bees) dominance hierarchy. Differentiation into elite and common working as observed in the data.

Next to explaining why the results were less clear in the bigger nests in Germany(van Doorn and Heringa, 1986), this also gives us insight into another question: how does a bumble bee nests tune itself relative to the season? In terms of producing generative offspring (queens and drones) it shouldn't be too soon or too late. Do they use an external cue for this? Probably not, because the same process occurs in artificial labs without cues. How then to tune the timing of killing the queen? Even queen age is unlikely to be used as a cue, because old queens can still be successfully introduced into a young nest. The results of the BUMBLE model suggest that there could be a **socially regulated clock**. Nests that grow slowly switch (kill queen) soon in the model and *in vivo*! This also explains the bumble bee nests in Germany, where there is fast growth, and less social differentiation due to influx of new individuals which disrupt the system, which keeps it "out of balance".

DODOM, TODO, and side-effects

What we observe in the BUMBLE model is social differentiation. This is dependent on nest structure and growth rate (without the periphery for inactive bees to settle the results are not obtained). Thus, we obtain elite workers that lay eggs, but these properties are not heritable: all start equally. The problem of the elites and their reproduction is a pseudo-problem. Instead we see that they play an integral part in a *socially regulated clock*, which is needed for the life-cycle of bumble bee colonies. Something that seemed puzzling from an evolutionary optimisation standpoint (why don't all workers lay eggs?) is logical and explained by an opportunity-based standpoint (TODO behaviour).

Finally, we note that there were some extra observations in the BUMBLE model in terms of adaptability, namely compensatory feeding. This was observed *in vivo* in nests where, if workers were taken away, the remaining workers would spend more time feeding. In the model, this happens simply through TODO, because there is more feeding to do! Because workers are taken away, other workers are more likely to come across unfed larvae and then initiate feeding. This is an example of **automatic adaptation**: behaviours that emerge simply because external input changes.

Learning what to eat

We will now look at the combination of TODO behaviour and learning with culture and diet within groups.

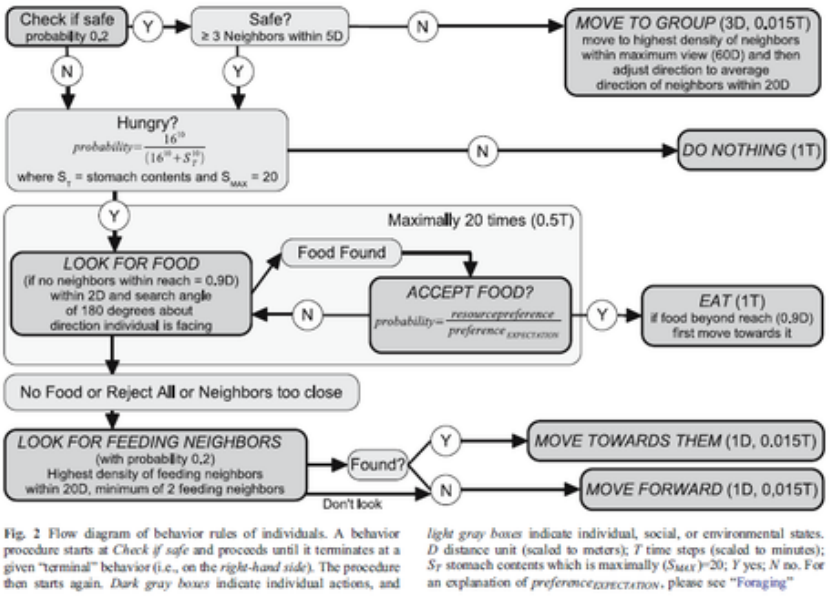


Fig. 2 Flow diagram of behavior rules of individuals. A behavior procedure starts at *Check if safe* and proceeds until it terminates at a given "terminal" behavior (i.e., on the *right-hand side*). The procedure then starts again. *Dark gray boxes* indicate individual actions, and *light gray boxes* indicate individual, social, or environmental states. *D* distance unit (scaled to meters); *T* time steps (scaled to minutes); S_T stomach contents which is maximally (S_{MAX})=20; *Y* yes; *N* no. For an explanation of $preference_{EXPECTATION}$, please see "Foraging"

Figure 3.5: Behavioural rules for individuals from (van der Post and Hogeweg, 2004)

TODO and learning: Grouping makes diet and diet makes grouping

In this example we combine TODO and learning. In most ecological models predators (or foragers) are assumed to know what to eat, but in reality that is not clearly predefined. Rather, we could expect the diet to depend on the availability of food in the environment. Furthermore, we can expect that you can learn what to eat. The most basic form of learning is probably *trial-and-error*. On the other hand individuals could mimick conspecifics (social learning). Here we will consider the impact of the environment on these processes and contrast opportunity and optimality. Moreover, we will look at how diet cultures can arise.

Van der Post and Hogeweg formulated a model(Van Der Post and Hogeweg, 2006; van der Post and Hogeweg, 2004), where:

1. Individuals seek out and select resources to eat. These resources give varying amounts of nutrients, and also have bulk and some value for toxicity. The resources are renewed each 'year'.

2. These individuals live in an environment with many different types of food.

3. They select resources depending on their (learned) preferences and on preference expectation: if they have not seen their favourite foods for a while, they are more likely to accept less-preferred foods.

4. Unknown (new) resources are always tried (sampled).

Given these behaviours, they wound up with a model where TODO behaviour occurs relative to the environment, *and an internal state* (the expectation of finding your favourite food). In the model, trial-and-error learning in solitary and grouping (gregarious) individuals is then compared. Learning is also compared in patchy (clustered food sources) and uniform (a little food everywhere) environments.

In the uniform environment, the diet was heterogeneous in groups: *minimisation of competition* (Figure ??). This is quite sensible: if everyone in the group eats something different, you can make the most of uniformly available resources. This type of diet, however, is not generally observed in nature. In the patchy environment, there are homogeneous diets in groups: *group diets*. These opposite results indicate that what is being done (which food is eaten) is partly an effect of what you see in the environment (which food you find). In mixed environments, where food is uniformly distributed and food patches

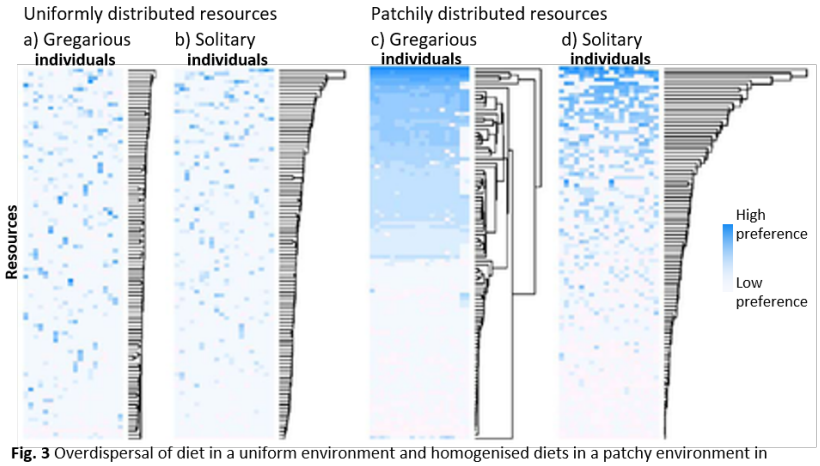


Fig. 3 Overdispersal of diet in a uniform environment and homogenised diets in a patchy environment in gregarious individuals. Shown are dendrograms and data matrices of diet preferences. **a,b** Uniform environment; **c,d** patchy environment; **a,c** gregarious individuals; **b,d** solitary individuals. Hierarchical clustering (Manhattan distance, single linkage) was conducted on resources according to how individuals prefer them. *Dendrogram terminals* represent resources, and *distances* between resources reflect the difference in magnitude in how preferred they are. The *colored data matrices* illustrate the resource preferences. *Columns* represent individuals, *rows* represent resources, and *colour darkness* indicates preference magnitude. Parameters: 100 resources, TR = 0.01, 10^{th} year.

Figure 3.6: Van Der Post and Hogeweg (2006)

are then superimposed as well, both effects can be observed at the same time. This gives a good search image for experimentalists and field workers, since that is perhaps the situation that occurs most often.

In the patchy worlds, different groups have different *diet cultures*: certain groups eat only certain foods (Figure ??). Specifically, this happens in the category of foods that are average to good: the very good foods are eaten by everyone, but the less energy-dense foods are only eaten by certain groups. We thus observe that *grouping makes diet and diet makes grouping*. Compare this to the CHIMP model, where a similar phenomenon is observed. For details, see (van der Post and Hogeweg, 2004).

Inheritance of dietary cultures

Now, we turn to culture and how we can use models to learn something about it. First, we point out that culture is a fuzzy concept. On the one hand, there are *traditions*, which represent stagnation. For example, a group can learn that certain foods are inedible and pass on that knowledge, while the foods are actually edible but they consumed a rotten cache. This works against what might be optimal food consumption. On the other hand, there is *cumulative change*: standing on the shoulders of giants. Behaviours learned over a life time can be passed on and improved to, for example, access more difficult food sources.

So, how should we get a better view of the concept culture? Well, we could start with inheritance through learning as a basis and see how far that goes towards culture. To do this, van der Post and Hogeweg extended their trial-and-error model from above, by adding primitive population dynamics to conduct transmission experiments in patchy environments(van der Post and Hogeweg, 2008)

They introduced a new, naive individual into each group every year, while removing one of the older individuals. The results of the model were that diet traits were inherited over generations, giving rise to *diet traditions*. Over the generations, there was also an increase in diet quality: the *cumulative change* aspect of culture (Figure ??).

We observe cultural phenomena here as side-effect: grouping + environmental conditions gives a semblance of culture for free. Of course, the parameters still need to be right: this doesn't happen under every parameter regime. Transmission of learned behaviour and learning need to be of certain strengths for this to work. However, these parameters can evolve due to all manner of reasons that have nothing to do with culture!

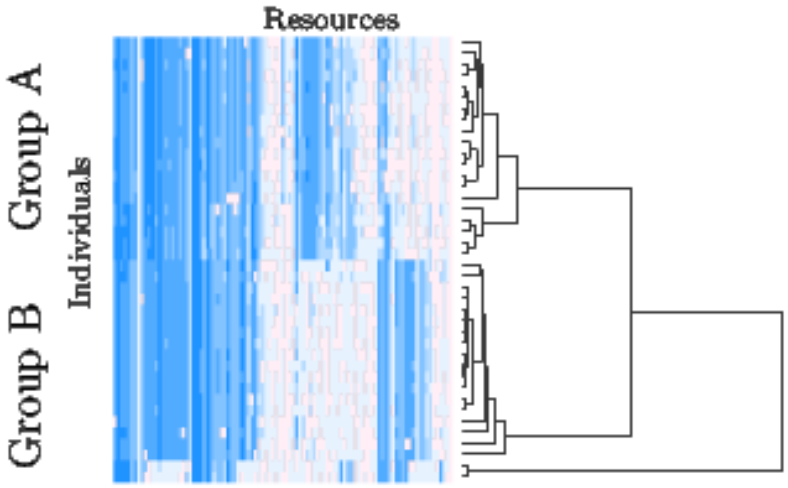


Figure 3.7: van der Post and Hogeweg (2004)

Conclusion optimality-based versus opportunity-based approaches

We observe that social learning can arise as a side-effect, not a strategy. Importantly, we get environmental-based memory, e.g. in a uniform environment, diet divergence is seen. We also obtain coherence between sets of behaviour: between grouping and social learning, and how this translates into traditional or progressive cultural phenomena. Moreover, we obtain alternative explanations for behaviours. We see *automatic adaptation*: behaviour automatically adapts due to the environment. The BUMBLE *socially regulated clock* (large nests versus small nests, self-regulation for correct reproduction timing), and compensatory feeding are two examples. There is also stagnation by doing what you did before: if you won previous dominance interactions, you will likely do that again. If you have learned to eat some foods, you will probably eat them again. Lastly, there are long-term effects and what you can call *long-term information integration*: over the generations, diet traits turn into diet traditions, but they are still amenable to cumulative change.

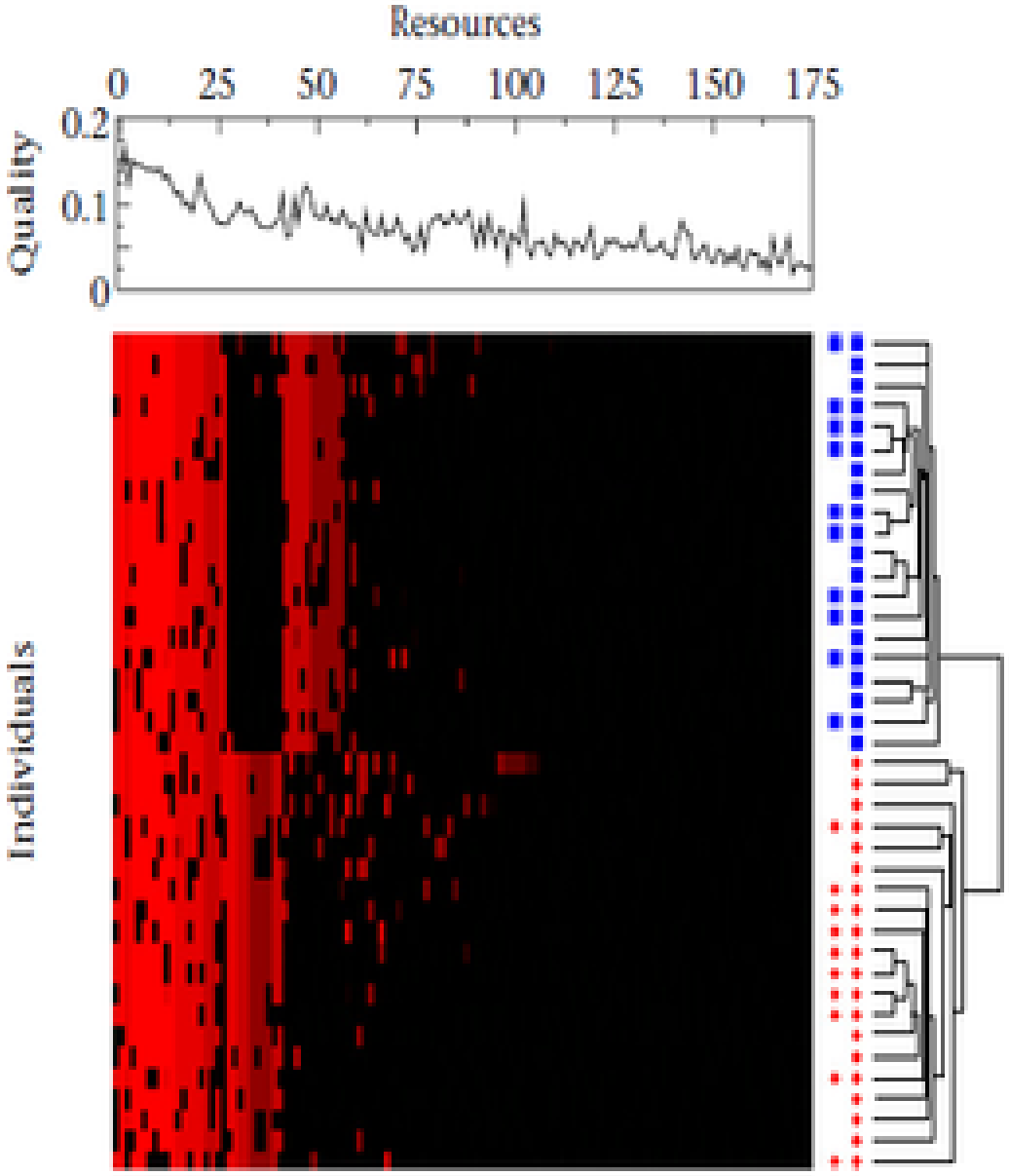


Figure 2. Diet traditions by trial-and-error learning ($N = 3$, $M = 10$). Individuals from two independent groups (square and diamond markers) at year 40 and year 80 (one and two markers, respectively) clustered according to similarity in diet preferences (average linkage, uncentred correlation distances). Dendrogram terminals and data matrix rows: individuals, data matrix columns: resources (red colour intensity indicates preference magnitude), top graph: resource qualities as ordered in the data matrix.

Chapter 4

Prebiotic evolution and the information threshold

We now come to a new part of the course, where we investigate how the evolutionary process led to the complexity that we can observe today. What are the prerequisites for Darwinian evolution, how does evolution lead to more complexity, and what problems are encountered along the way? That is what we will focus on in this part of the course.

Introduction to prebiotic evolution

One of the most fundamental issues in biology is the question of the origin of life. How did life arise from molecules in the environment of a pre-biotic world? Not surprisingly, there are many hypotheses and theories regarding this issue. We will address some of the main ones here. So far, research has mainly focussed on three aspects of life:

1. Vesicles: how do vesicles emerge from the properties of lipids and how do they grow and divide? (Schrum et al., 2010; Mansy and Szostak, 2009; Luisi et al., 1999; Ruiz-Mirazo et al., 2014)
2. Catalysis: how do we get self-sustaining networks of catalytic peptides (as we see in cells)? (Falkowski et al., 2008; Ruiz-Mirazo et al., 2014)
3. Replicators: how do we get replication of information-coding molecules (as we see in DNA and RNA)? (Ruiz-Mirazo et al., 2014; Orgel, 2004)

In each case there is a focus on building blocks (i.e. a basic unit from which complexity can be assembled) and some form of replication process. Assuming the availability of these simple building blocks, we can summarise these theories as:

Life is energy/nutrient cycling (catalysis first)

All living organisms have catalytic networks that convert certain nutrients into building blocks or energy. Furthermore, the core machinery in these networks seems to be conserved over many species. We could therefore speculate that (auto)catalytic networks stood at the basis of life. In this scenario, there has been a focus on hydrothermal vents, where we get energy (temperature) gradients and compartments (in the rock) for free, and in this rock there are potential catalytic substances (metal sulphides) (Lane and Martin, 2012; Lang et al., 2010). This would create an optimal environment for the emergence of autocatalytic sets (of e.g. peptides). However, the main question in this scenario remains: are these autocatalytic sets evolvable? Can selection act and complexity arise from them?

Life is cells (compartments first)

As suggested by, for example, Szostak, life requires compartmentalisation (protocells) (Schrum et al., 2010; **?**). These compartments ensure some kind of organisation, and furthermore allow for competition

between different compartments (and hence selection). In wetlab experiments, Szostak et al. have shown that they can get large, heterogeneous, mutlilamellar vesicles (i.e. micells/protocells with fatty acid "membranes") that grow and divide into multiple daughter cells. Furthermore, these protocells can absorb fatty acids from other protocells, leading to competitive growth (Adamala and Szostak, 2013). One interesting point from Szostak's experiments is the following: initially, they worked in a very "clean" system, studying uniform, unilamellar vesicles. This led to problems with divisions: during each division the volume decreased until eventually the vesicles were too small and disappeared. When they switched to more heterogenenous, multilamellar vesicles, these problems disappeared and they found protocells that could keep dividing without volume loss. Actually, this heterogeneity is more likely to occur in a highly heterogeneous, messy environment, which is what we should expect from a prebiotic soup. This example shows that, while in experiments we always try to be as "clean" as possible, the heterogeneity that we should expect in real life might sometimes make things easier!

Life is evolution (replication first)

We can also state that the main characteristic of life is replication of heritable information that can be selected. According to Gerald Joyce (Joyce, 2012), living systems:

1. Have a molecular memory (genotype) which is shaped by experience (selection)
2. Are maintained by self-replication

If we consider this starting point, RNA is a good candidate molecule because RNA can both store information (template, nucleotide sequence) and catalyse reactions such as its own replication (i.e. ribozymes). The current *in vitro* state-of-the-art of an RNA system that is capable of ongoing Darwinian evolution is an evolving, self-replicating system. It is ligation based: $A + B + E \rightarrow E.AB \rightarrow E.E \rightarrow E + E$. This system grows exponentially and evolves. However, only a few nucleotides are prone to evolution, most nucleotides in A and B are fixed. These fixed nucleotides are "borrowed" from a current ribozyme, that is known to have a ligation function (Samanta and Joyce, 2017).

RNA world

We will focus on the replication-first scenario. RNAs act as replicators as well as information carriers. Our focus is on whether RNA molecules, with some form of self-organisation, can evolve to resemble living systems. Recent findings show that many RNAs can act as enzymes called ribozymes (Doudna and Cech, 2002): think of the ribozymes in the ribosome, for instance. This means that RNAs are molecules that have special properties and behaviour (catalysis), and are capable of carrying the information coding for these properties. They can, in principle, also replicate this information (by replicating themselves). Hence, RNA molecules are ideally suited as a minimal model for the evolution of complex life from simple building blocks.

In our approach to the origin of life we focus first on replicators in the form of RNA molecules. Unlike peptides, in which self-replication is always highly dependent on particular other peptides (non-generic), RNAs can always self-replicate (in principle) by binding of the correct complementary nucleotides. In this way an RNA-world provides the minimal requirements for evolutionary optimisation (Darwinian selection):

1. Generic replicators
2. Independent synthesis and decay
3. Mutation
4. Competition

The difficult problem is the synthesis and stability of RNA. One of the biggest questions in prebiotic evolution is how early RNA replicators managed to evolve into larger complexes. Given that an RNA world would suffice in terms of the minimal requirements for evolutionary optimisation, the most immediate question is whether Darwinian selection would actually occur, and what its consequences would

be. Can an early population of self-replicating, rapidly mutating RNA molecules exist and give rise to complexity? This issue was studied by Eigen *et al.* when they developed their *quasispecies theory* (Eigen, 1971; Eigen et al., 1988).

Quasispecies theory

Eigen and Schuster formulated the quasi-species model in order to study whether some key ingredients of evolution would allow for selection processes to arise in non-living RNA molecules and lead to a bootstrapping process which could amplify complexity. In other words: whether simple, short RNA molecules could evolve into longer, complex RNA molecules (Eigen and Schuster, 1977, 1978). By using RNAs, they took the best-case auto-replicator. Furthermore, they assumed a sequence-independent replication rate. In this way, they had a pre-biotic system that fulfilled the four requirements listed in the last section.

Eigen tried to capture these requirements in his replicator equations. These equations describe the abundances of different replicators (e.g. strains, indexed with i) in a chemostat. A chemostat is a type of bioreactor where new medium and/or micro-organisms are continuously added, while an equal volume of medium and/or micro-organisms is removed, such that the total volume of the system is constant. The equations are:

$$\frac{dX_i}{dt} = A_i * Q_i * X_i - d_i X_i + Sum(W_{ij} * X_j) - \Omega_i \text{ with } \Omega_i = (X_i / Sum(X_j)) * Sum((A_j - d_j) X_j)$$

Here, X_i is the abundance of replicator strain i, Q_i the quality of replication of strain i (i.e. how faithfully the replicator can reproduce itself), A_i its growth or replication rate and d_i its death or decay rate. The first term thus represents successful replication of X_i, again forming individuals of strain i, while the second term is the decay of X_i. Unsuccessful replications (i.e. non-perfect replications) of X_i mutate and form other strains j. This happens at a rate $A_i * (1 - Q_i) * X_i$. The third term in the equation represents the formation of X_i by the replication and mutation of other types X_j. Here, W_{ij} is the amount of j mutants that are equal to the i genotype. This takes into account the rate and quality of replication (i.e. A_j and Q_j) and the mutational distance between i and j. Lastly, Ω_i is a dilution term: it leads to a decrease in X_i-concentration at a rate proportional to the frequency of X_i in the population (first factor) and the total population growth (second factor). This term represents the chemostat assumption: it assures that the total sum ($Sum(X_j)$) remains constant, i.e. that the total concentration of replicators does not change.

For the more mathematically trained people: the main result of this equation is that, over evolutionary time, the system will converge to the normalised eigenvector corresponding to the largest eigenvalue of matrix W (mutation interactions). This eigenvector describes the final frequency distribution of all strains. If you are a not sure what that means, in layman's terms we can say that instead of a single "fittest" replicator, we get a *distribution* of replicators that is "fittest". This is what we call a **quasispecies**. The evolutionary process maximises the total growth rate, and hence selects for the quasispecies (i.e. a collection of genotypes) that grows fastest. This means that it does not necessarily go to the fittest RNA, but to the fittest combination of RNAs which is a cloud of mutants that are (genotypically) related.

The first surprising insight from this model is that certain replicators are present in the population not because they are optimal (i.e. have a high growth rate), but because they are mutationally close to a replicator with a high replication rate. This illustrates an important point: just observing a certain phenotype in the population does not mean that this phenotype is evolutionary optimised *itself*!

A second important result is that whether evolution leads to the fittest RNA in this model depends on the mutation rate. When the mutation rate is too high the fittest replicator (in terms of replication rate) might no longer be able to survive. This can best be illustrated in a simpler model, that is described below.

Master-mutant simplification

In the original quasispecies model, fitness can be defined *a priori* as the rate of replication. Let's assume a **Dirac-delta** like fitness function: one particular replicator has a high replication rate (a1), while all

other replicators have the same, lower replication rate (a2). We can then simplify the model to two equations: a **master equation** (the fittest) and a collective **mutant equation** (all others):

$$\frac{dx}{dt} = a_1 * x * Q - d * x - x * ((a_1 - d_1)x + (a_2 - d_2) * y)$$
$$\frac{dy}{dt} = a_2 * y + a_1 * (1 - Q) * x - d_2 * y - y * ((a_1 - d_1)x + (a_2 - d_2) * y)$$

Here, x is the fittest strain (i.e. the **master sequence**) and y represents all the less fit mutants. x + y = 1. Because x is only a specific "strain", and y represents all other mutants, we neglect back-mutations from y to x. Both x and y have a negative term to keep x+y equal to 1 (textbfthe chemostat assumption). We can use this set of equations to investigate under which conditions the master sequence, x, will be present in the system. For this we will use the invasion criterion, which looks at whether a small amount of x is able to invade a population of y. In effect, this allows us to research when selection works: if there are conditions under which x is competed out of the system, that means there are fundamental restrictions to when natural selection can work.

For x close to zero, we can simplifie into:

$\frac{dx}{dt} = a_1 * x * Q - d * x - (a_2 - d_2) * x * y = x * (a_1 * Q - d_1 - (a_2 - d_2) * y)$.

Since x+y = 1 and x 0, y 1. Hence, we see that x can invade ($\frac{dx}{dt} > 0$) if:

$a_1 * Q - d_1 > a_2 - d_2$

Assuming $d_1 = d_2$, i.e. the difference in fitness is only given by the growth rates a_1 and a_2, this simplifies into:

$Q > \frac{a_2}{a_1}$

and since $\frac{a_1}{a_2}$ is the **selection coefficient**: the relative fitness advantage of a_1 over a_2, which is denoted as σ, we can simply write:

$Q > 1/\sigma$

This is the **error threshold** condition. If the replication quality Q is not large enough, x cannot invade into a population of y. This means that the master cannot sustain itself in the population; and hence that we do not see survival of the fittest: if one master sequence would mutate into existence into a large population of less fit sequences, it *would not prevail*. Therefore, these equations lead us to believe that natural selection cannot operate if quality of replication is too low. In other words: for too low replication quality, fitter sequences cannot be selected for, and as such the system is outside the grasp of natural selection. Would this have posed a problem during evolution as it occurred on Earth? We should wonder whether the error threshold poses a problem at realistic parameter values. For this, we need to determine what sequence lengths can be maintained at particular mutation rates.

Information threshold

RNA is a polymer, and hence mutations at many locations in the sequence are possible. Hence, Q, the quality of replication per genotype, can be calculated as q^L, where q is the quality of replication *per residue* and L is the length of the sequence (measured in number of base pairs/residues). Now, given that q is fixed, limitations on Q (as given by the error threshold) also put constraints on the length of the sequence L.

For q close to 1, we can use the following identity:

$q^L \approx e^{(-L*(1-q))}$; where e is the natural exponent 2.71828....[1]

Then, substituting this into the expression we found for the error threshold yields:

$Q > 1/\sigma \rightarrow e^{(-L*(1-q))} > 1/\sigma$

[1] We simplify by using only the first order of a Taylor series, which is good enough for q close to 1

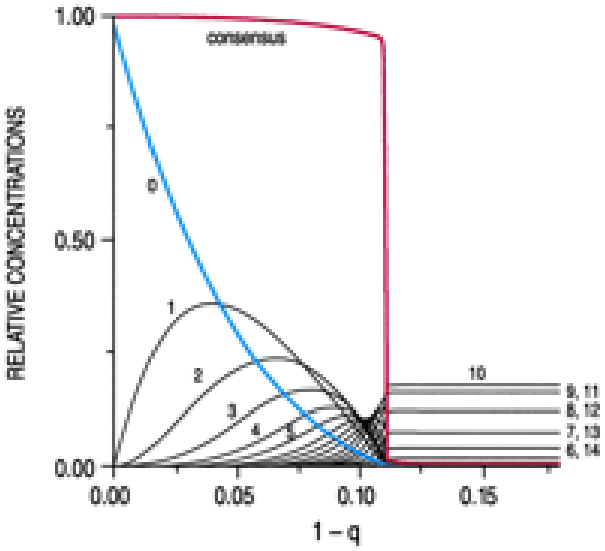


Figure 4.1: Graph visualising the Error Threshold On the X-axis is the error rate (1 minus the quality of replication). The Y-axis depicts the concentrations of the master sequence (blue) and its n-distant mutants. The abundance of consensus (the presence of a consensus sequence) it depicted in red. Not that after the threshold, all replicators are in equal abundance, and natural selection does not function.

If we now take the natural logarithm on both sides, we arrive at the following equation:
$-L * (1 - q) > -ln(\sigma)$

Which finally gives:

$L < ln(\sigma)/(1 - q)$

What have we learned by jumping through all these mathematical hoops? The interesting thing about this equation is that it predicts a limit in sequence length, above which information cannot be maintained because the master sequence can no longer be maintained in the population. In other words: for a given error rate per base pair, only a limited amount of information can be stored in a replicator. This is the concept of the **information threshold**, which of course directly relates to the **error threshold**. In simulations, it appears that RNA strings larger than +/- 50 bases are not under the error threshold anymore (Takeuchi and Hogeweg, 2007). Hence, for longer RNA strings another molecule (enzyme/ribozyme) would be needed to improve the copying fidelity (increase q). It has been calculated that the information threshold already occurs for a sequence length of 50, a 95% replication fidelity, and a very strong selection coefficient. If we imagine early, unoptimised copying of replicators, we can imagine that replication fidelity might be a lot *less* than 95%. This is thus not merely a theoretical problem: we should expect this **error threshold** or **information threshold** to be a major obstacle in evolution towards complexity that life has somehow overcome.

The dynamics before and after crossing the error threshold are quite different (Figure **??**). The main way in which we can recognise that the system has not yet crossed the error threshold is that the common ancestor of all individuals is the master sequence (in blue). Hence, all other sequences are only present in the population because they are mutants of the master sequence, and this is represented in the consensus sequence of all sequences in the population (which is still the master sequence; red). This is thus a clear example of a *quasispecies*: a master sequence and its close mutational neighbours make up the population. Note that although the master sequence is the common ancestor of all sequences present, it is not necessarily in the majority. As we get closer to the error threshold, the mutants greatly outnumber the master sequence.

Three important remarks at this stage:

1. Quasispecies theory is deterministic, and defined on infinite populations (in contrast to Muller's ratchet which describes a stochastic effect on finite populations).

2. The replicator equations have the nice feature of combining evolutionary and ecological time scales. Mutation happens while ecological dynamics also play out (often, these are separated to simplify matters).

3. By writing down simple differential equations we come to two fundamental concepts: error threshold and quasispecies. This turns Darwinian selection into a theory, away from a tautology: now it can be false (i.e. it is not always true that replicating sequences with differing fitness are within the grasp of Darwinian natural selection).

Importance of the information threshold

But is the information threshold really a problem for evolution? The most likely answer to this question is yes, since we have seen that the information threshold already becomes problematic for relatively short RNA sequences (see above). Furthermore, by looking at genome size and mutation rates we get a good indication of its consequences: research has shown that there is a negative correlation between genome size and base substitution mutation rate (Lynch, 2010). Larger genomes have less mutations, and in order to have less mutations, they are often equiped with repair mechanisms. The latter mechanism sounds like an "easy" solution to the information threshold, but it is in fact paradoxal: how to accumulate this repair information, if accumulating information was our problem to begin with?

It is important to note that we have used a *best-case scenario* in the *quasispecies model* (i.e. to prove there is a problem) to show there is a limitation of the power of Darwinian selection:

1. We have used an *infinite population*, which means that:

   (a) All possible replicators exist, and the best quasispecies is always selected. If you would start with only a subset, it might be more difficult. In other words: we start with all possible genomes of replicators, but in reality, not all possible replicators with a sequence length of n bases will have existed, so it might be more difficult for selection to lead to a master sequence.

   (b) There are no stochastic population dynamics, meaning that even if something is present in very low concentrations it doesn't go extinct. In effect, if there is 0.01 master replicator, we still take it into account, while it would actually be extinct in the real world.

2. We used *strong selection* and a *single-peaked fitness landscape*. This means that there is a very strong transition in fitness between the master replicator and the mutants. Said another way: we assume a large selection coefficient. All mutants are very unfit, while the master sequence is supremely fit. That is optimistic: not all mutants need to be much less fit than the master replicator. If the landscape is different, delocalisation of the quasispecies still occurs but the transition is less sharp (Takeuchi and Hogeweg, 2007). You will learn more about fitness landscapes in the next chapter.

3. We used a *fixed sequence length and no other constraints on length*. In actuality, there would be negative energetic selection on longer sequences: shorter sequences are replicated faster and take less energy to replicate.

Moreover, this analysis doesn't address how we get longer sequences in evolution, although it does allow us to focus on length alone. Even in this best-case scenario, sequence length slams into a brick wall if mutation rate is too high and sequences are too long. Life definitely has a major problem. This is called **Eigen's paradox**: to store more information we need better replication, but to get better replication we need to store more information(Peck and Waxman, 2010). In other words: longer sequences require better replication, but better replication can only be achieved by more proofreading or fidelity-enhancing processes, which would require more sequence length to code for them. Nonetheless, we exist, so the problem must have been solved. How, then, to cross the information threshold?

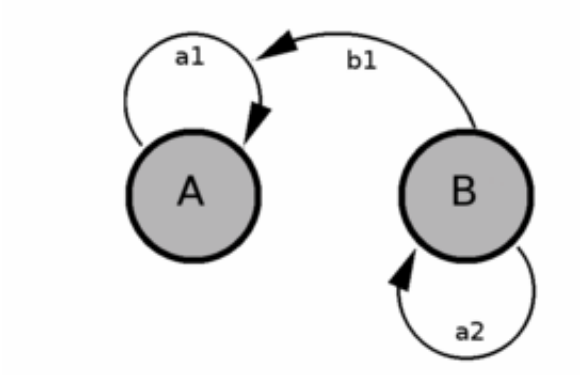


Figure 4.2: A simple two-species hypercycle

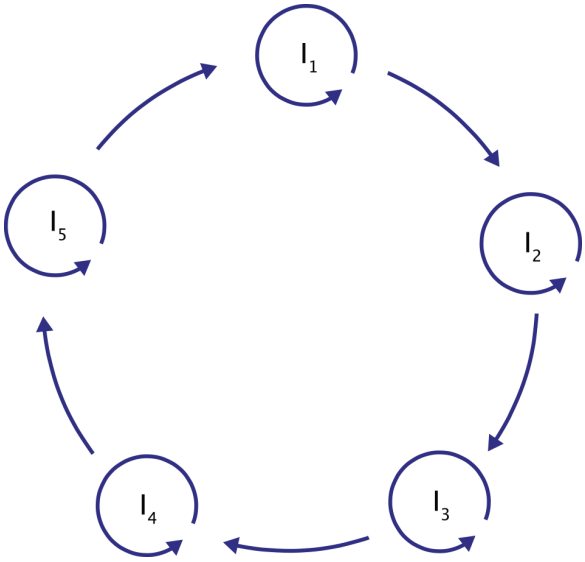


Figure 4.3: A five-species hypercycle

Hypercycles

A first attempt to explain how the information threshold could be crossed was also formulated by Eigen and Schuster (1977-1979) who considered an ecological solution: an ecology of interacting molecules that form stable networks and so together maintain more sequence length than each on their own(Eigen and Schuster, 1977, 1978). Their ecological modeling approach focussed on ecological stability (and thus, ironically, neglected mutations). A very simple form of this idea is depicted below (See Figure **??**).

The accompanying ODE equations are:

$\frac{dA}{dt} = a_1 * A + b_1 * A * B - \omega$ $\frac{dB}{dt} = a_2 * B - \omega$

where a_1 and a_2 are the catalysis-independent replication rates, b_1 is a catalysis-dependent replication rate and ω is a chemostat-assumption term (see the quasispecies equation). This leads to the system below:

If we generalise this two-species model to any number of species we get:

$\frac{dX_i}{dt} = a_i * X_i + b_j * X_i * X_j - \omega_i$

with the second term signifying that X_i receives catalysis from X_j. Thus, you can get hypercycles as below, with self-catalysis and help in catalysis from others in cycles of varying sizes (Szostak et al., 2016):

If we look at the simplified ODE, $a_1 + b_1 >> a_2$ would imply that A outcompetes B. The other way around (A catalysing B) will not work either, then B will outcompete A. We will always expect the replicators that are not catalysed to be outcompeted, and this holds for any number of species that catalyse each other. Hence we need feedback: a cycle.

The resulting dynamics depend on the cycle length. There is always one non-trivial equilibrium. When the cycle length is 3 or less than three, there is a fixed point attractor (see introduction to ODEs in model formalisms). At 4 replicators, there is a stable spiral. From 5 replicators onwards, there is a limit cycle with increasing amplitude. The latter notion illustrates one of the problems with ODEs: replicators in these systems can go to concentrations/numbers of 0.001 before receiving catalysis again. That is not realistic, as we discussed in Chapter 2 when we talked about the **attofox problem**.

In this model, the only stable topology is a circle, because non-catalysing branches will always outcompete other molecules. In other words, parasites that do not give catalysis but do receive it will quickly outcompete the mutualists. When mutations are added to the ecological system it is clear that even the stable topologies (cycles) are quickly destroyed by parasites which destroy its cyclical nature. If only one replicator evolves such that it gives less catalysis than it receives, the system is doomed.

Moreover, new hypercycles without parasites cannot invade the system since the X^2 term (the $X^i * X^j$) dominates the equations in the model. This is the only quadratic term, and thus has most influence on the fate of the system. Thus, concentrations of replicators matter more than their growth rate (while growth rate is what you could see as a measure of fitness). Hence, cycles with a better growth rate but low concentration could never invade an already established cycle. This in itself is an interesting consideration for Darwinian selection. Darwinian selection only works if replication rates are more of less linear, i.e. if in $a * X^n$:

1. $n > 1$, then survival of the first (very strong founder control)
2. $n < 1$, then survival of everyone (this will be covered further in a question during the practicals)

Peculiar properties of the hypercycle solution

We considered the stability of the system in terms of invasion of mutants. This consideration is illustrative of the importance of a famous essay by Dobzhanksky (Dobzhansky, 1973): Nothing in biology makes sense except in light of evolution. Mutations happen, and so one should always consider what a biological system does in response to these mutations: is it stable? In this case, we see that the hypercycles are resistant to invasion of hypercycles with *better replicators*. Invaders can't get a foot through the door if they are a tiny concentration to begin with, as you would expect when a mutant first arises (invades). There is thus **once-only selection**: once a hypercycle is established, it cannot be invaded by hypercycles with better replicators. Thus, this is no answer to the information threshold. In this way, *complexity could never arise*, because fitter hypercycles (higher growth rate) would not outcompete less fit hypercycles.

In fact, the solution as a whole is a bit peculiar. The information threshold is encountered because high mutation rates cause there to be a hard limit to the length of a sequence: increase it, and natural selection cannot work anymore. The conundrum is thus that longer sequences require lower mutation rates, while having a lower mutation rate is probably dependent on having a longer sequence that codes for some proofreading step or repair mechanism. This attempt at a solution, however, *does not incorporate mutations*. In fact, if any replicator would mutate (lower) its parameter governing how much catalysis it gives to another, that replicator comes to dominate and the system fails. Why? There is no pressure to coast a replicator into giving high catalysis to others. If it lowers its catalysis of others, it still gets help in catalysis itself (from an altruistic sucker), and might be able to catalyse itself more. Win-win! Except that if that goes on, the system eventually dies out. So, somehow, this "solution" to the error threshold only works when we allow for no errors at al!

A final weakness is that survival of a hypercycle with many replicators (>= 5) depends on limit cycles with high amplitudes: high fluctuations in replicator counts. If you correct for replicators where 0.0001 replicator still exists (as is possible in the ODE formalism), the cycle is actually unsustainable. Thus, the system is also instable for large cycle sizes.

What happens if we add space?

We will now consider removing the infinite and the well-mixed assumptions of the ODE hypercycle model. These are assumptions that are implicit in the ODE model formalism. They snuck in through the back door, but life isn't infinite and certainly isn't always well-mixed. We will explore how space and locality influence the hypercycle model.

Hypercycles in space

Spatial modelling systems do not assume well-mixed conditions, and when using a CA, it also assumes that there is limited reproduction space (i.o.w. population is finite). Here we discuss the study of hyper-cycles in a CA by Boerlijst and Hogeweg (Boerlijst and Hogeweg, 1991). In this model the only assumptions are: decay of replicators, replication (local in space) and catalysis of replication (local in space). Thus replicators can replicate locally, and catalyse others locally. With these assumptions, the CA rules are a translation of the ODE assumptions into the CA formalism.

The question then is: how does the system behave? A very general result of this model is that the system organizes itself into a particular form of mesoscale patterns, namely **spiral waves**. In these spirals, bands of individuals that receive catalysis from the band preceding their band are formed. So if A catalyzes B catalyzes C catalyzes D catalyzes A, we will see patterns in which we see bands of A - B - C - D - A, etc. In the figure below, we can see such spiral waves, where different replicators are given different colours. Waves arise from the core and spin outwards, showing the banded pattern described above.

Just like we saw in the hypercycle ODEs, the dynamics in the CA depend on the number of species (i.e. length of the cycle, see Table 4.1):

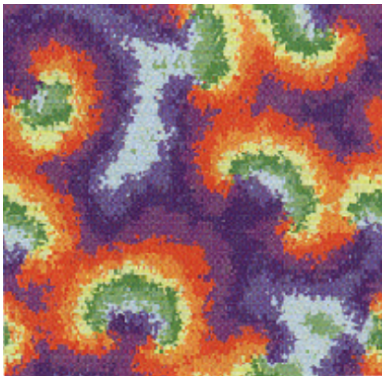


Figure 4.4: Hypercycles in space form into spiraling mesoscale patterns when the number of species is greater 4

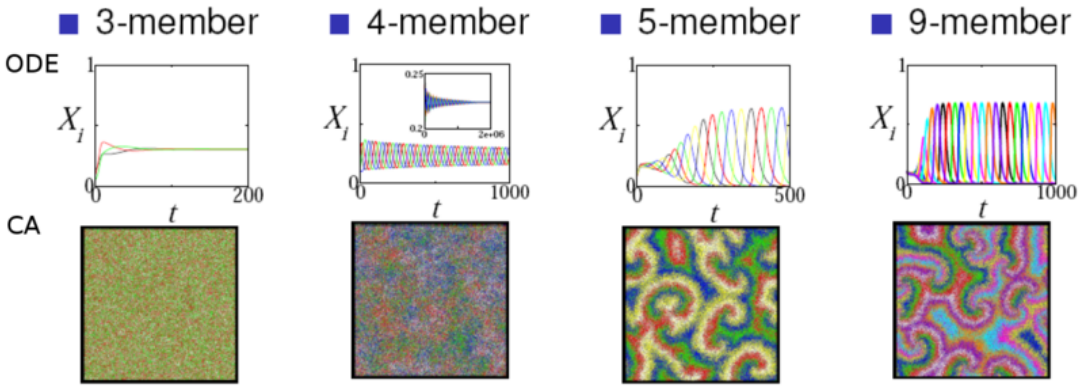


Figure 4.5: Where ODEs form "attofox" problems for more than 4 species, the CA hypercycles form into spiraling mesoscale patterns

Table 4.1: Modeling formalisms and their assumptions on space, time, and the modeled variable(s)

Cycle length (n)	ODE behaviour	CA behaviour
n <= 3	stable fixed point	no pattern / mixed
n = 4	stable spiral	chaotic wave pattern
n >= 5	limit cycle (note: attofox!)	spiral waves

So, we see that the stability of the dynamics is very different in the CA than in the ODE. In the ODE, we found limit cycles with high amplitude for systems with many species. Hence, we would expect extinction in the oscillations (when correcting for attofoxes). In the CA, this problem is solved: the stable spiral wave patterns ensure that a large number of species can be sustained.

We also see in the CA that the spirals grow from their core. All individuals that are part of a spiral are offspring of one of the few core individuals. Hence, long-term fitness is determined by location: only individuals in the spiral core are *at the right place at the right time* and will have non-zero long-term fitness!

The next question is *whether these spirals have any meaning for the biological system in question?* What do they represent? Unlike the rule 54 CA studied by Hanson and Crutchfield (Hanson and Crutchfield, 1997)), there is a model relation to something in biology, i.e. entities in the CA represent molecules. Hence, the mesoscale pattern has some biological meaning in terms of those entities. In other words: mesoscale patterns in elementary CA 54 consisted of particles, but they were just random effects of a binary rule set. Here, the rule set is based on biology, and hence the resultant mesoscale patterns could have an actual meaning in biological terms, and are likely very important to biological systems.

The next question we can ask is *whether such meaningful patterns are important, or just colourful scenery?* This question becomes clearer when we study the system's properties, in particular its resistance to parasites. Unlike the ODE model, we find that the CA system can recover even from introduction of very severe parasites (see Figure ??). We see that parasites can grow locally, but eventually die out as they are pushed out the the periphery of the spirals. Obviously, you could argue that if we were lucky not to introduce any mutant parasites into the spiral cores, we should expect that the parasites should be expelled from the system, because the spirals are generated and refreshed from their cores, washing out all parasites in the spiral. However, even if we specifically introduce parasites into the spiral core and these manage to take over a whole spiral, they are still purged from the system by other, non-infected

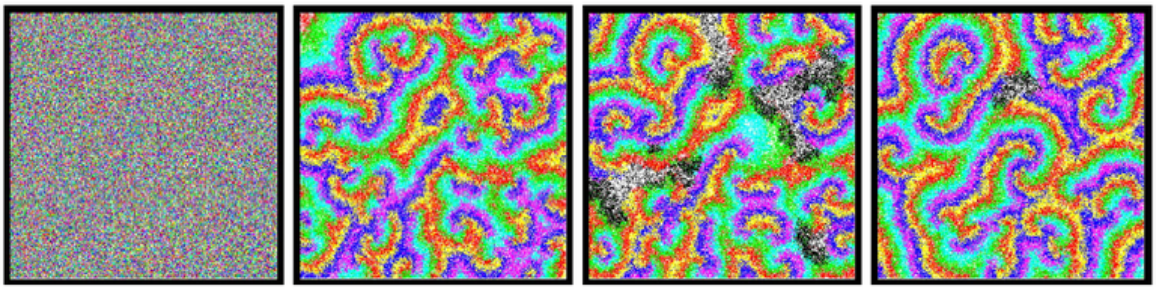


Fig. 1 – Pattern formation and expulsion of parasites in a spatial model of the hypercycle (cf. Boerlijst and Hogeweg, 1991a,b).

Figure 4.6: Hypercycles in space are very resistant to parasites due to the formation of spirals, where parasites are pushed out of the system

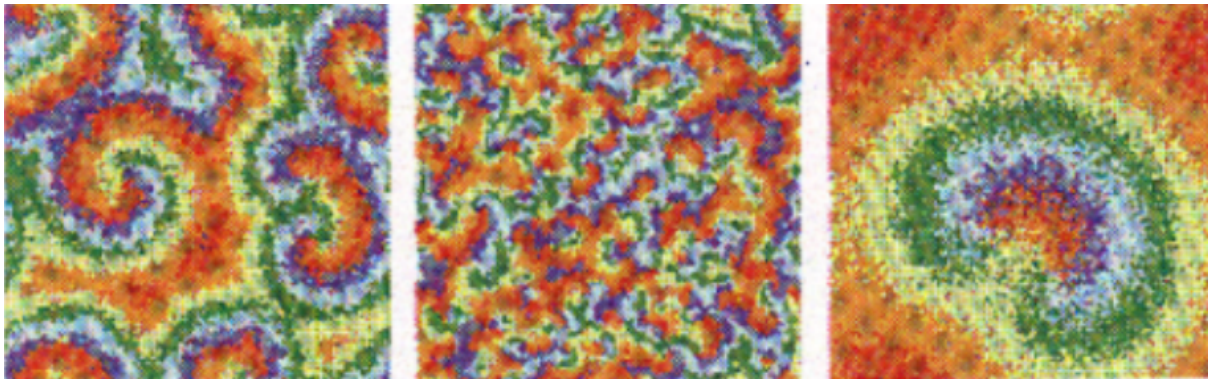


Figure 4.7: Hypercycles in space with different levels of diffusions (from left to right: low, 0, high)

spirals. In this instance, we take a worst-case scenario in terms of parasite invasion in order to show how resistant the system is!

Given that we find such a difference in results with the ODE model: *which is the best model?* This is actually a bad question! Instead, it is more fruitful to view both models as two extremes in terms of *local interactions* versus *well-mixed populations*. One can then ask what happens in intermediate situations: what if there is some level of diffusion? In the CA, we can add this diffusion in (see Margolus diffusion in timing regimes). Then, results show that with no diffusion, waves have quite a small scale and stochasticity plays a large role in the system (Figure ??). As diffusion increases, the spatial scale of waves tends to increase (larger waves) making them more resistant to parasite invasion. On the other hand, fewer waves fit in the field (there are only so many CA grid points) making the system more sensitive to invasion if a wave would be taken over by parasites. In a sense, these results indicate an irony, though it may take some time to understand it. In going towards the more well-mixed state (in an ODE, everything is consistently well-mixed), the power of mesoscale patterning is actually increased. We introduce diffusion, and the waves become larger! As long as you increase the space somewhat concomitantly with diffusion, mesoscale patterns keep occurring and shaping the dynamics of the system. This illustrates that the ODE result is the limit of extreme diffusion in an infinite domain. In other words: only in the extreme case that everything is well-mixed all the time and there is infinite space do the ODE results hold. In mixed space (local interactions with diffusion) on the other hand, spatial resolution (patterns) dominates, no matter how much we increase diffusion!

Multi-level evolution: emergent levels of selection

The previous section has shown that many properties of the ODE system can be reversed in a spatial system due to the emergence of mesoscale patterns (spiral waves). In the ODE model with 5 or more species, the hypercycle becomes a limit cycle and becomes unstable. In the CA, in contrast, it becomes a pattern of spiral waves, which is globally stable and can be resistant to strong parasites. Local

interactions and space thus make a huge difference. Next to these differences, we also observe a reversion in the direction of selection on death rate:

In the ODE model, increasing the death rate of a replicator is clearly a disadvantage. However, if we allow for mutations in the death rate of individuals in the CA, we see that a higher death rate can evolve! This might require some significant mental readjustment, so let us break this down:

1. We are in a CA, with local growth and interactions. If a replicator wishes to replicate, it needs an empty square next to it. Replicator A helps replicator B helps replicator C, etc. Thus, fast growth depends on empty space, and having the replicator that catalyses you close to you.

2. Spiral waves form. It is known from theory surrounding excitable media that spirals that rotate faster invade on the space of spirals that rotate slower(Krinsky and Agladaze, 1983; Boerlijst et al., 1993).

3. It is also known that all offspring comes from the core.

4. Given this knowledge, we can understand what happens:

   (a) For an individual replicator, a higher death rate seems unfavourable: it has less time to replicate.

   (b) However, spirals that rotate faster can invade spirals that rotate slower.

   (c) Faster rotation, in this case, entails a quicker succession of replicator A, by B, by C, etc.

   (d) A limiting step is grid point occupancy: if replicator B wants to replicate into a spot where A still is, it cannot. This limits the speed of the spiral.

   (e) Thus, given that faster-rotating spirals are *fitter spirals*, a higher death rate evolves, so that replicators more quickly vacate space, resulting in a faster-rotating spiral.

This process does not include a trade-off where higher death rates lead to a change in some other replicator parameter like birth rate. Instead, this is a *new level of selection*: faster-rotating spirals are fitter, and this reverses the selection pressure on the lower-level hypercycle replicators).

This illustrates how local interactions in space lead to non-local selection criteria in the form of spirals. Therefore, in contrast to the ODE, we do not get the phenomenon of *once-only selection* because locally stronger molecules can win and globally faster spirals win. Spiral waves can therefore be said to enslave the molecules they are made up of because the fate of these molecules is completely determined by how good the spiral that they live in is. Hence, we get positive selection for higher death rates and for giving catalysis which is a reversal of selection relative to the ODE.

This argument only holds for the case that there are stable spirals. However, these only occur for a certain parameter range of death rate, birth rate and catalysis rates. If, for example, the death rate of the mutants is too high, the spirals will disappear. Once there are no more spirals, the selection for early death will disappear as well and the direction of selection will again be reversed. Thus, this does not weaken but strengthens our conclusion: it is the selection pressures exerted by the mesoscale organisation that leads to this result. To summarise, the events proceed as follows:

spirals → selection for higher death rate → faster spirals → selection for even higher death rate → spirals disappear (start of chaotic behaviour) → no higher-level selection pressures → selection of lower death rate

These reversals of the direction of selection will cause the system to evolve to those regions on the border between spirals and spiral break-up, and hence lead to evolution to the **edge of chaos**, or **border of order**: the parameter value just below the value where spirals break down.

If we think back to the classification of all 1D CA models with the λ parameter, you might remember that there was a class IV with universal computation and unpredictable behaviour. This was wedged in

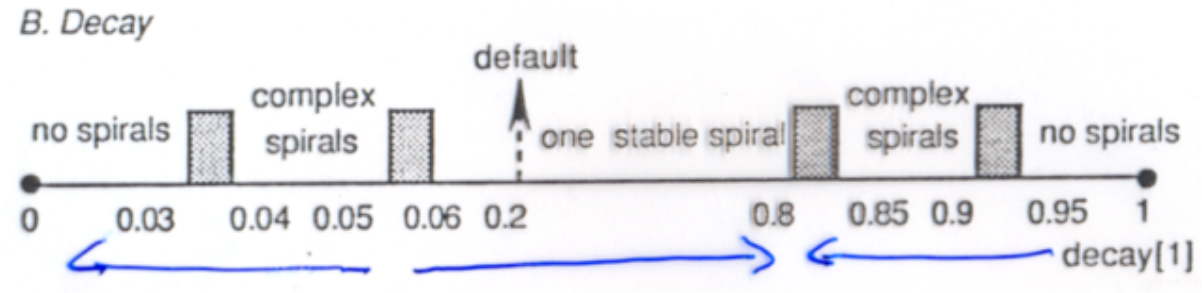


Figure 4.8: The decay rate of individual replicators is given on the axis, and it is shown which behaviour is observed at different values. The blue arrow displays the direction of evolution

between class II CA, which had a limit cycle, and class III CA, which had high dimensional chaos. Thus, there was a small sweet spot just beyond class II CA models but just before class III CA models where there was an unusual and rare behaviour. Here, the systems evolves to something between class II and class III. This, too, can be seen as a *border of order* of sorts: a small region just before chaos sets in.

Mapping higher-level selection in hypercycles back to the information threshold

It is important to consider that this spatial hypercycle system does not directly solve the problem of the information threshold. We merely look at whether shorter or longer hypercyles can invade: if you introduce a mutant that skips one replicator and gives catalysis to the next (i.e. normally A -> B -> C, now A-> C), what happens? We have, however, gained a very important insight from this study, namely the concept of **multi-level evolution**. We have seen that individual interactions lead to spatial patterns (the mesoscale) which through mutation and selection (Darwinian evolution) affect individual interactions and therewith spatial patterns. Thus, the replicators cause the mesoscale patterns, which cause the replicators to change, which changes the mesoscale patterns. This has implications for how we view the evolution of complex systems and how we model them. The traditional view was that we could make a model, allow a certain variable to evolve, and thus see what the optimal evolved value for that variable is. The CA view is that there are micro rules which are set in stone, which cause emergent mesoscale patterns. Now, we arrive at a third vision: the evolutionary system generates mesoscale patterns, which in turn influences the selection pressures on the micro scale, which then again influence the mesoscale patterns. The rules of "what is good to do", is thus *not* set in stone.

This is extremely important: we must let go of the idea of a static set of rules which are set in stone. We have seen now that the lowest level does not make sense except in light of higher level processes. It does not make sense that replicators should evolve higher death rates, except in the light of higher-level spirals. Thus, evolution works on multiple levels, and we need an understanding of all for it to make sense.

One more point to consider with respect to spirals is that they are very generic and occur in many different systems. This means that conclusions that are drawn from them are very fundamental! For instance, in PDEs spirals do not arise spontaneously as in CA due to minimum scale, but they need to be initialised (Boerlijst and Hogeweg, 1995) However once initialised we see that they cannot be resistant to parasites since parasites will be in all cores and the system dies out. However this is due to the everything being everywhere problem (think of the *attofox problem*). When a threshold is used for a probability of a molecules being present (i.e. to conserve mass) the PDE shows the same properties as the CA (Boerlijst and Hogeweg, 1995). Spirals are therefore generic, as are the conclusions drawn from them.

Subconclusion hypercycles in space

We have now seen that the hypercycle model in space behaves completely differently. Where there are limit cycles in the ODE, there are spatial spiral waves in the CA model. This allows a *higher-level*

selection pressure to shape the fate of the replicators. There is evolution to *the border of order*: replicators increase their death rate to be as high as possible without going so high that the spirals in which they are present go extinct. In the ODE, parasites kill the system. In the CA, that is not true: even if they are introduced into the core of a spiral (where all progeny comes from), they kill that spiral and other spirals then fill up the empty space. The system survives. By adding diffusion to the CA, the spatial situation more approximates the ODE. The extreme locality of the CA system is more relaxed. Despite that, mesoscale patterns continue to exist and shape the behaviour of the system. This shows that the ODE findings are only true for the limit case: only when population size is infinite and everything is perfectly mixed do they hold. While this spatial system thus shows a very different result than the ODEs, and shows us the importance of multi-level selection, it is not the solution to the information threshold: if mutations are added, there is only limited stability.

Further than Dobzhansky

Dobzhansky wrote that nothing in biology makes sense except in the light of evolution. At this stage, we have added to Dobzhansky's quote. We have seen a glimpse of the importance of CA models to understand what happens in biology. Local interactions, micro-macro transitions (local rules give an emergent macro behaviour), non-linear dynamics, mesoscales, and simple rules that lead to complex behaviour: they are all invaluable to understanding biological systems. We have glimpsed this specifically through modelling hypercycles in a CA system and observing the resultant behaviours of the system. Nothing in biology makes sense except in the light of (lessons drawn from) CA models. The most condensed bottom line of this course is that:

"Nothing in biology makes sense except in light of ***multilevel*** *evolution"*
Of course, giving a condensed bottom line at only about 60 pages in might strike you as premature. Do not fret, for many details will still be added in the coming chapters.

Minimal eco-evolutionary model of emerging higher levels of selection

The hypercycles were a proposed solution for the information threshold problem. Once we had implemented the hypercycles in a CA model, we observed that we could use this model to study the emergence of mesoscale patterns (spiral waves), their stability against parasites, and higher levels of selection. However, we can also go back one step and ask: what kind of dynamics do we get if we make a minimal model of a replicator (i.e. an "RNA") and a potential parasite? Such a model was made by Takeuchi and Hogeweg (Takeuchi and Hogeweg, 2009a).

Consider a system with a replicator R and a parasite L. The parasite can be in a folded (functional) and unfolded state, and only the unfolded state can be replicated (i.o.w. it is only truly parasitic when *unfolded*). The parameter l of a parasite describes which proportion of the time it is folded (and hence cannot be replicated). Both replicators and unfolded parasites can be replicated by a replicator R. The parameter kL describes how strongly parasites bind to the replicators. The system can be summarised by these reactions:

Not that in this system, parasites can evolve two things: how parasitic they are (kL), and how *often* they are parasitic (l). This system was implemented in space (CA). What happens if we let the system run, and let the parameters l and kL evolve? The system forms waves in space. There is a difference between young waves (that have just arisen) and older waves. In young waves, the average l value of parasites is high (weak parasites), while in older waves the average value of l is lower (see figure; blue = high l, yellow = low l). Hence, we see that over the lifetime of the wave the parasites in the wave become stronger, because they spend less time in their folded state and more time being replicated.

However, if we look at a longer timescale we see that the average value of l in the system goes up: the system seems to evolve weaker parasites! How can this be? Remember that the parasites can evolve two parameters:

1. l : how often they are in their folded (inactive) state

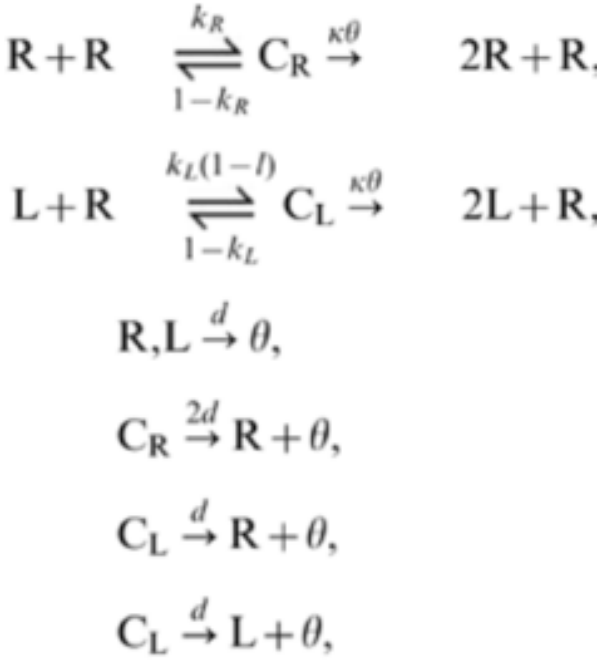


Figure 4.9: All reactions in the simple replicase-parasite (RP) system of Takeuchi and Hogeweg (2009a)

2. kL : how strongly they bind to the replicator

If we look at the evolution of these two parameters, we observe an emergent trade-off between kL and l (see below). This trade-off is not inherent to the individuals, but rather emerges because of selection on the higher level entities (the waves). Evolutionary trajectories first converge towards this trade-off and then move upward along the trade-off line, increasing both kL and l, leading to parasites that bind more strongly to the replicator (high kL) but do so less often (high l).

Why do we find this trade-off? As said, it emerges from selection on the wave level. Intuitively, you might expect that parasites would become stronger and stronger, increasing their kL and decreasing their l. This is also what we see over the lifetime of a wave. However, waves with very strong parasites have a very high probability of dying out (being consumed by the parasites) quickly, and will hence not be able to produce many new waves. In this way, selection at the wave level poses a maximum on the strength of the parasites. Why then do we move up on the trade-off (instead of down, towards lower l and kL)? Again, we need the level of waves to understand this: If l is high the probability of birth of a new wave is higher, because these starting waves experience little trouble from the parasite. To explain that a bit more in-depth: waves of replicators are followed by waves of parasites. These parasites have a high chance of being folded, and a high chance of binding to a replicator. This is a game of chance, and by chance a replicator in the wave might find itself with free space at the *back of the wave*. If the parasite would have a low l, that would immediately be bound to and this nascent wave killed off.

To look at this more closely, we can use the ODE framework: if we make an ODE system of a replicator and a parasite, and make the parasite parameters evolvable, we will find that the parasite wins out and the system goes towards extinction. In ODEs, space is infinite and everything is well-mixed, so no mesoscale patterns arise. However, we can use this ODE framework to check what happens after evolution of the CA system. If we take the evolved parameters from the CA system and run the ODE system of replicators and parasites with them, we might find out more about the workings of the CA system. In Figure ??, we see that values of kL and l both have a tendency to rise in the simulations. If we take the initial parameters and the final parameters of the system, we can see a marked difference in the behaviour of the replicators. In the initial parameter regime, replicators only increase slightly, and do so slowly. In the final regime, replicators increase much more rapidly, and reach much higher numbers before the parasite catches up. This is exactly the maximisation of wave birth that we mentioned above:

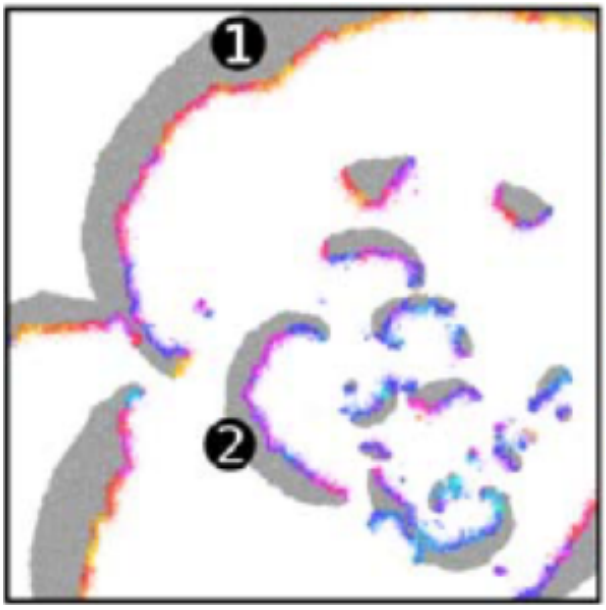


Figure 4.10: Wave formation in RP-system by Takeuchi and Hogeweg (2009b), where waves have a lifetime of their own in which parasites become increasingly aggressive.

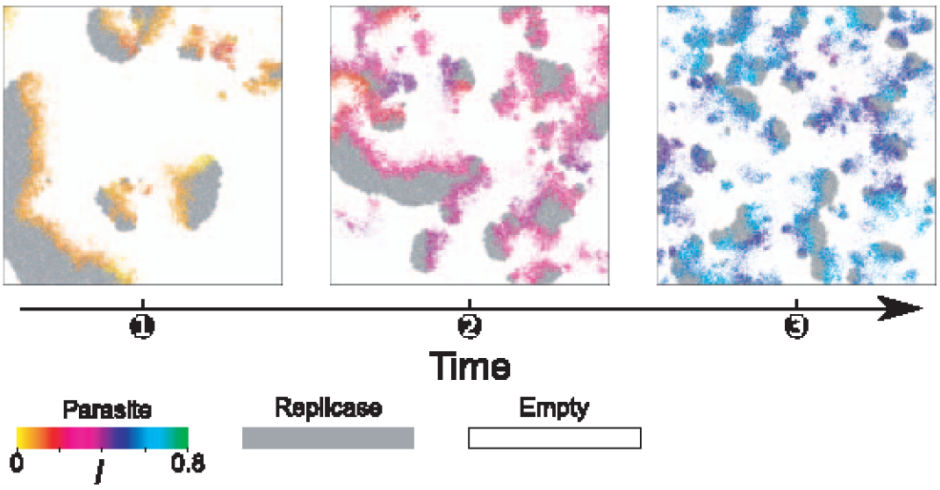


Figure 4.11: Evolution towards increasing l in RP-system by Takeuchi and Hogeweg (2009b)

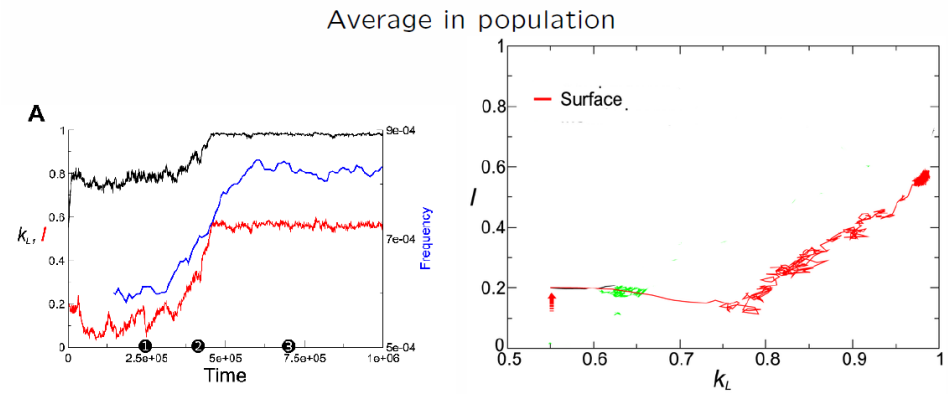


Figure 4.12: Evolution towards, and on the trade-off between l and K_L in RP-system by Takeuchi and Hogeweg (2009b)

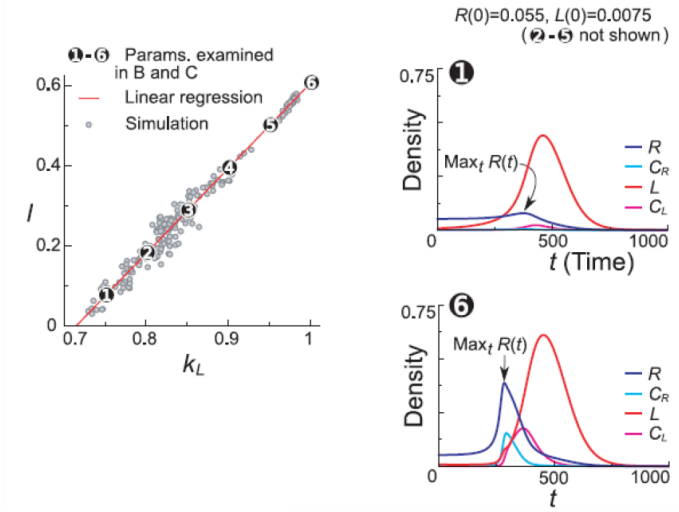


Figure 4.13: ODE versions of the RP system by Takeuchi and Hogeweg (2009b) to study what the evolved parameters entail.

if a replicator breaks through at the back of a wave that is being 'eaten' by parasites, it can *very quickly* replicate away and start a new wave.

Note that the waves we see in this model have the following qualities:

1. They are born, live for some time, and die.
2. They mature: the strength of the parasite changes over the lifetime of a wave
3. They mutate and can be selected
4. They compete with each other (for space)

Hence, these waves should be seen as Darwinian entities, upon which selection can act. This model shows an example of *long-term evolution on the level of these mesoscale patterns*. Interestingly, the system once again evolves to the *edge of chaos (border of order)*: the fastest possible waves that do not die out are retained in the system. Even though one would naively assume that parasites would maximize their availability for replication, we have shown here why this needs not be the case.

A final note: this system has now evolved parasites that are in a folded state most of the time. In this folded state, these parasites might have some different (ribozymatic?) function. It shows that in a system where a parasite preys upon a replicator, at least the *ability* to have some function besides replication is opened up. Hence, this system shows a potential for ecosystem-based information accumulation through its parasites.

Separation of time scales

The fallacy of separating evolutionary and ecological time scales

We will now discuss the problem with separating evolutionary and ecological time scales. If you recall, we mentioned the importance of time scales in our discussion of the CA formalism. There, we mentioned that how you should encode the next state function (NSF) depends on the time scale of the interactions that you are modelling. If a modelling step entails the passing of a whole year, and therefore the competition between seedlings of different trees to grow in a certain spot in a forest, you should somehow encode that. You could do this by implementing a claim-mechanic: have all trees bordering a square engage in a sort of fitness-battle (with some chance involved) and the victor's seedling then grows into a tree there. You thus influenced the time scale of competition: if you allow squares in the CA to pick a random neighbour and copy its state, you basically say that competition takes a lot longer to have an effect

than the reproductive act: others dont influence colonisation directly. Rather, competition is then on the level of empty space and which neighbours there are. In the tree example with the claim-mechanic, you know that you are modelling a longer stretch of time, and thus the times cales of competition and reproduction do intertwine.

Here, we look at another time scale issue. Oftentimes, models assume that evolutionary and ecological time scales are separated. What do we mean by this? The simplest example are ecological models (population dynamics) that assume that, since mutations are rare events, they do not happen often and can safely be ignored. Another approach is that mutations happen, but that it is assumed they are either competed out of the population instantaneously if they decrease fitness, or fixate instantaneously if they increase fitness, and the ecological dynamics can then continue. An example of such a study is studied in the exercises. In reality, of course, there is a lot of standing variation in a population due to mutations, and population dynamics could very well be affected by this. We will showcase two examples that clearly indicate that separation of time scales, and deciding what is fit a priori, don't work.

Time-dependency of fitness in a spatial host-parasitoid system

We often think that fitness is a clear concept: the more of your offspring survive, the fitter you are. However, the following example shows that it is not so easy. We look at a host-parasite system, as classically defined by Nicholson and Bailey (Nicholson and Bailey, 1935). The conclusions here were reached by work by many people, including Boerlijst and Hogeweg (Boerlijst et al., 1993; Savill et al., 1997). In this system, there are hosts (similar to prey) that are parasitised by parasitoids (parasites that live alongside or inside the host before killing them) that kill them. These parasitoids can move towards the hosts with a certain inclination B. If B is 0, parasitoids randomly diffuses. If it is > 0, they preferentially moves towards higher host density. When $B = 1$, the parasitoids sort perfectly according to host density: two hosts will attract twiche as much parasitoids as one host. If $B > 1$, parasitoids move towards areas with the highest host density. In that case, if there are 5 hosts in location A, and 6 hosts in location B, parasitoids would all move towards location B. Initially, you would assume that for optimal parasite replication, it is best if $B = 1$.

The system is simulated as a MAP lattice: as you may remember, MAPs are a discrete form of ODE. Each cell in a grid thus contains such sets of equations, and hosts and parasitoids can move between lattice points (grid points). In the figures, hosts are not shown, and patches are coloured according to the parasite with the highest density in that patch (Figure ??). We attempt to understand how and why the migration parameters of the parasites evolve.

In this simulation, we can see three levels of selection:

1. Individual hosts and parasitoids
2. Chaotic waves versus spirals of hosts and parasitoids
3. Regions of chaotic waves or spirals

By studying the life cycles of patterns in the model, it becomes clear that a life-history of spirals emerges. High B leads to chaotic waves, whereas low B leads to spirals. Thus, the evolution of B determines what spatial patterns emerge, and there is a distinct order of events:

1. New spirals are born at the interface between spiral and chaotic wave regions with high B as descended from chaotic waves
2. These new spirals rotate fast, as they have active migration
3. Over time, B within spirals evolves towards lower values, causing these spirals to lose their domain (faster spirals outcompete slower spirals, as we have seen)

Why does this happen? Well, we know that all individuals in a spiral come from the core. Therefore, being in the core is a good thing: you have high fitness. Individuals with low B have more of a chance to stay in the core, because individuals with high B migrate out towards higher host densities! In chaotic

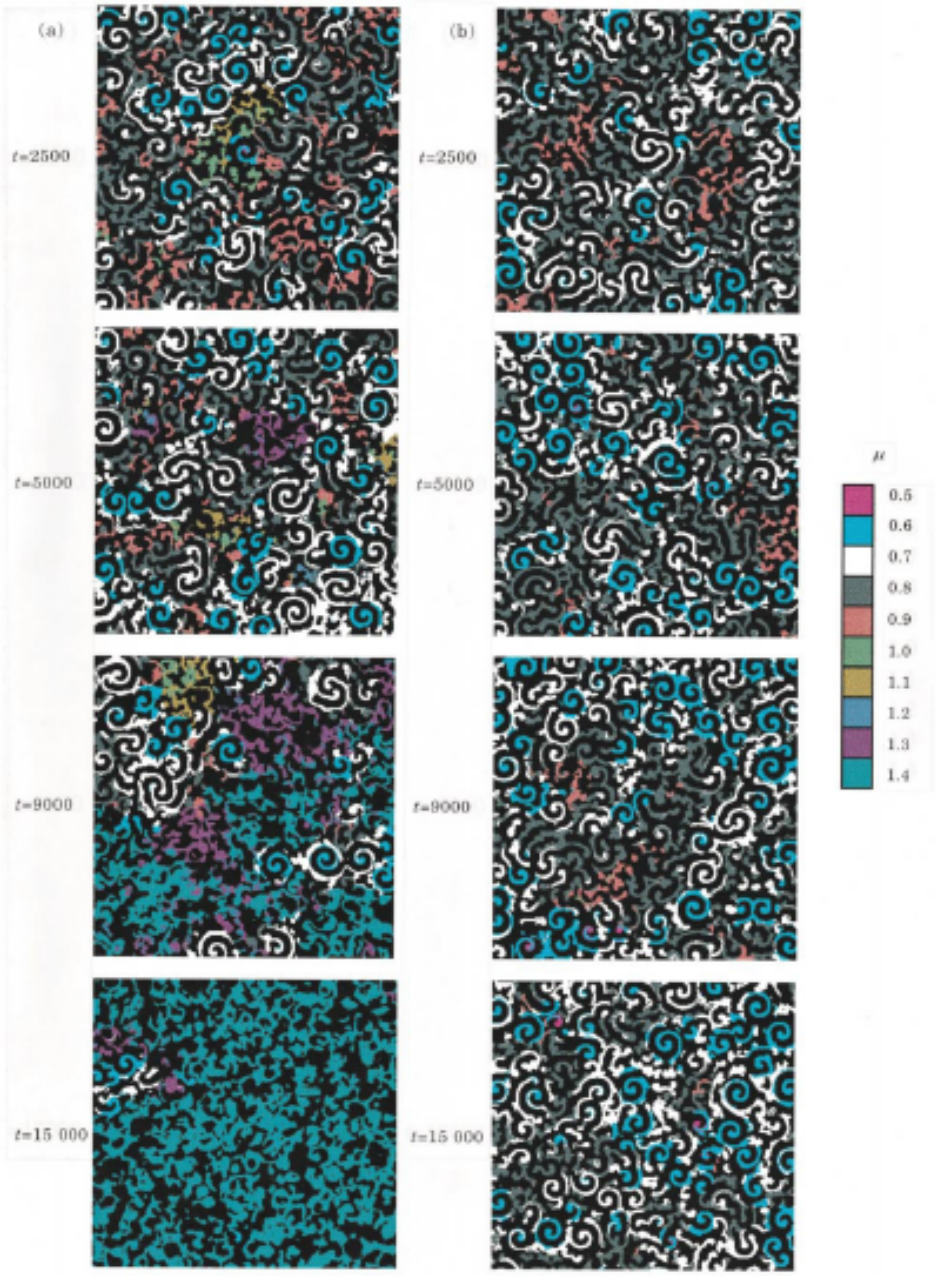


FIG. 1. Snapshots of typical simulations at several generations. The lattice size is 300 × 300. The colour of a patch represents the parasitoid type with the highest density in that patch (hosts not shown). If the density of the parasitoid type with the highest density is less than 1 the patch is coloured black to better observe the spatial patterns. Only parasitoids with $0.5 \leq \mu \leq 1.4$ are observed. Parasitoids with other aggregation strengths exist but hardly ever with the highest density in a patch. Initial and boundary conditions are identical in both simulations. (a) $\alpha_P = 0.2$. There is a long transient where spirals and high μ turbulence coexist. However, turbulence, composed of a range of parasitoid types from 1 to 1.8 with $\mu_c = 1.4$, finally outcompete the spirals. (b) $\alpha_P = 0.25$. Only spirals and low μ turbulence exist. High aggregation strengths cannot invade and out-compete spirals. Other parameters are $\alpha_P = 0.9$, $\eta = 0.001$, $\lambda = 2$, $b = 1$ and $\alpha = 0.05$.

(facing p. 12)

Figure 4.14: Spirals in lattice map model by Nicholson and Bailey (1935)

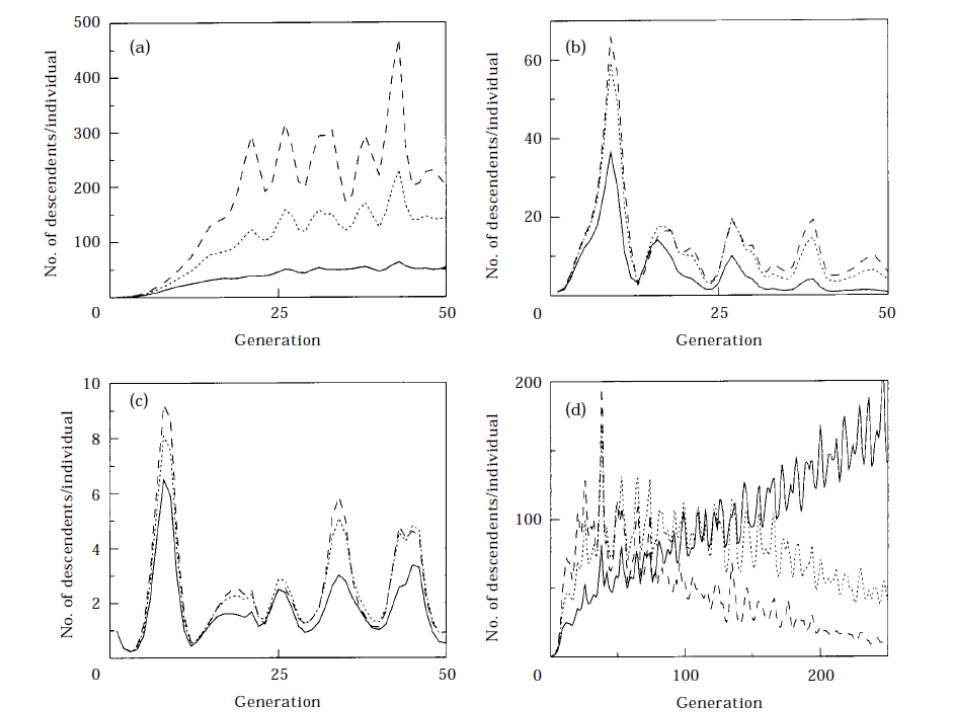


FIG. 9. Inclusive fitness of parasitoid types within spatial patterns. (a) spiral core, (b) spiral arm, (c) turbulence. At generation zero all parasitoids within an area of 5×5 patches are tagged in each pattern. Over time the number of descendents per tagged individual at generation zero is calculated to give an inclusive fitness for each aggregation strength. In all patterns the higher the aggregation strength the higher the inclusive fitness ($\mu = 0.5$ solid, $\mu = 1$ dotted, $\mu = 1.5$ dashed). Parasitoids within a spiral core have the highest inclusive fitness of all parasitoids (c.f. y-axis scales). (d) shows the number of descendents from a core but plotted over a longer time period. Initially the higher aggregating parasitoids have the highest inclusive fitness but, because of the longer time, the spiral has the opportunity to interact with its environment (in this case turbulence) and the lower aggregating parasitoids have the highest inclusive fitness.

Figure 4.15: Short vs long-term fitness in lattice map model by Nicholson and Bailey (1935)

regions, however, it makes sense to have high B values: you need to migrate towards hosts to reproduce, while in a spiral, a new wave of hosts will come along.

It becomes clear that there are various selection pressures especially when inclusive fitness is analysed over different time scales (see Figure ??), i.e. how much fitness individuals have over the generations. When viewed over 50 generations, low B parasitoids have the least offspring, no matter what spatial pattern individuals are in. Keep in mind that this is inclusive fitness measured over a much longer period than is done in any field experiment. If we look over 300 (!) generations, individuals with the lowest migration towards hosts have the greatest fitness within spiral cores and also dominate over other locations (spiral arms and chaotic waves). Thus, in the long term, low values of B are fittest. So does everyone evolve to low B values? Clearly this does not happen, and the explanation is that over the 300 generations descendants from low B ancestors will always evolve to higher B values: once away from the spiral core, you are fitter locally with higher B.

In this spatial system where mutations in migration towards hosts can occur, we see many counterintuitive or novel phenomena:

1. There is no fixed notion of fitness, which always remains the same.

2. At one time point, lower fitness individuals might be the major source of offspring in the system (initially low fitness of low B individuals)

3. In any biological system one should expect multiple evolutionary time scales

Thus, the most important conclusion here is that fitness is not static, but instead that **fitness is a time-dependent function**. Indeed, immediate fitness benefits can be overruled by long-term fitness

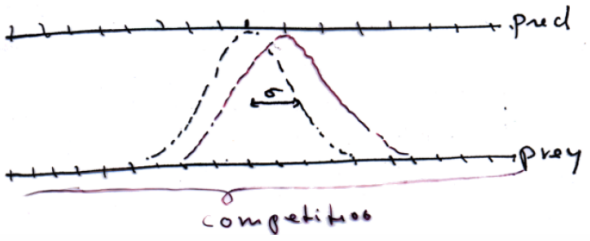


Figure 4.16: Phenotypic space of predator prey dynamics implemented by Van der Laan and Hogeweg (1995)

effects: individuals with low B evolve, even though they initially have low fitness. Only after many generations and within the core of a spiral are they incredibly fit.

Eco-evo predator-prey model

Van der Laan and Hogeweg developed a predator-prey model to study the interaction between ecology and evolution (Van der Laan and Hogeweg, 1995). The model was set up with many phenotypic variants with a probability of consuming each other based on Gaussian distribution around predator phenotypes (Figure ??). Phenotypes are on a wrapped scale (think of the time of day being active; 24:00 is right next to 1:00. In the same way, maximum and minimum values are right next to each other mutationally on this scale) and there is no space so that all prey and predators compete globally. There is therefore only *shape-space*: the space where variants are ordered (nearness). Offspring can be mutants on this shape-space. This shape-space is depicted in the figure below .

You thus see that there is 1D space for both predators and prey to sit on. The Gaussian curve from a predator to the prey below shows how much of the prey with that phenotype the predators can eat. σ is the parameter that determines how broad the area that each predator phenotype can eat is, higher σ thus gives a more generalist predator. Mutations in the model can be turned off and on, and mutation speed for predators and prey is the same (not because we assume that is so in real life, but because it is not what we are looking at here). We will now look at what simultaneously simulating ecological and evolutionary dynamics does to the system.

Simulations are initialised with monomorphic (only one phenotype) and symmetrical predator and prey populations (i.e. they start at the same position along the phenotype axis). Quickly, however, there is speciation into two predator-prey pairs. Interestingly the predators are not on top of the prey, but in between them. Moreover, there seem to be evolutionary oscillations in the parameters.

So how to characterise this role of evolution? The most straightforward way is to stop mutations and see what happens. When this is done the system mostly dies out, though it does depend on when mutations are stopped (not shown in the figure). This indicates that mutations affect ecological population dynamics. Moreover, diversity is lost when mutation rates are set to zero. Note that this is not just a quasi-species, but four "true" species: they exist separately on the phenotypic scales. Given that they exist *with* but (often) die out *without* mutations, we can see that *mutations stabilise this fairly diverse ecosystem*, whereas their lack dooms it. What the wiggles show is that, at all times, populations are being pushed phenotypically. Although this is a very simple model, this could be an important mechanism for maintaining ecosystem diversity, i.e. continuous adaptation through mutations maintains diversity.

To gain more insight, the system is described using ODEs and parameters are fitted (Figure ??). Note how that apart from the aforementioned evolutionary oscillations, we here observe strong ecological oscillations. The dotted lines are the system when mutations are allowed. The oscillations are less extreme. This thus leads us to the conclusion which is clear as day: mutations stabilise ecological dynamics.

Indeed, population dynamics often uses the argument that mutations are such rare events that they don't really impact the ecology. The lower figures show that the *lower* the mutation rate, the *more*

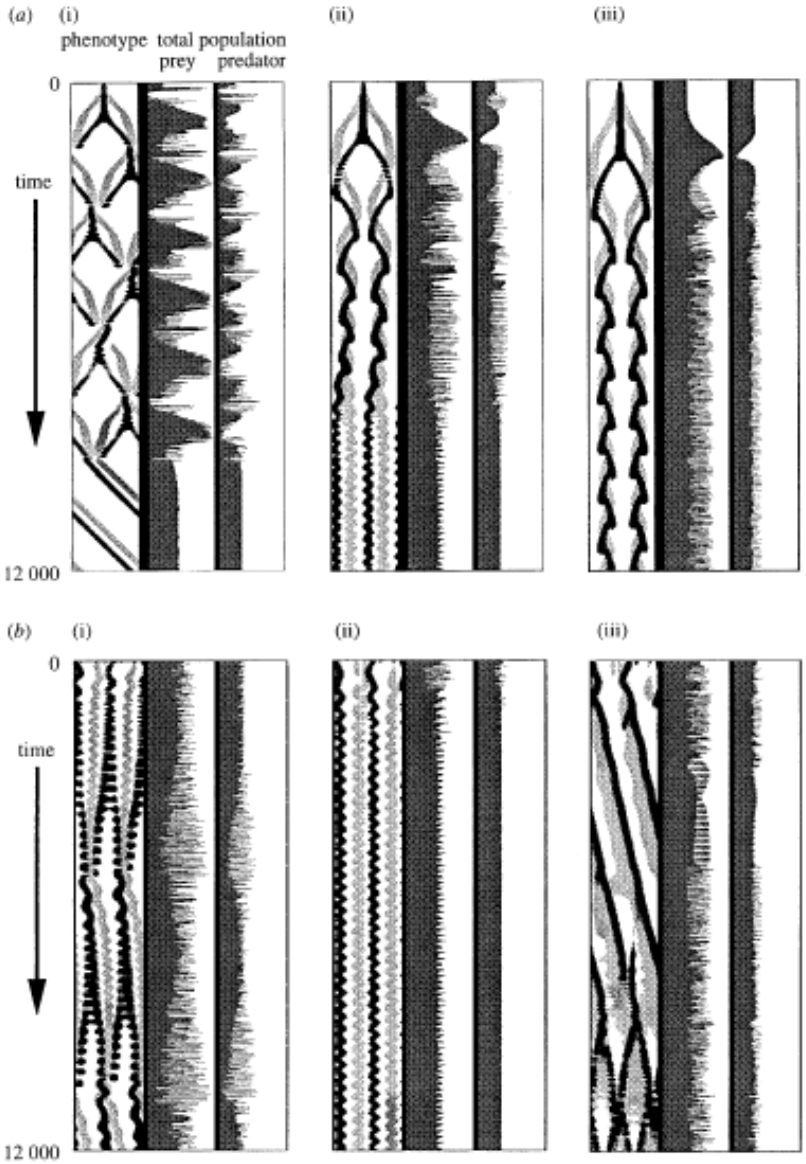


Figure 1. Time series of evolutionary and population dynamics, with ⟨a⟩ symmetric and ⟨b⟩ random initial configurations and with (i) $\sigma = 0.4$, (ii) $\sigma = 0.5$ and (iii) $\sigma = 0.6$. In the three columns of each time series, time ranges from $t = 0$ to 12000 with a display-interval of 15 time steps. The first column of each time series shows the phenotype axis of both prey (grey) and predator (black). Only the phenotypes with a local population density higher than 0.01 are indicated. The other two columns show the total population density of prey and predator on a horizontal axis which ranges from 0 to 200.

Figure 4.17: Space-time plots of the phenotypic axis (left panel) and population densities of predators and prey (middle and right panel) Van der Laan and Hogeweg (1995)

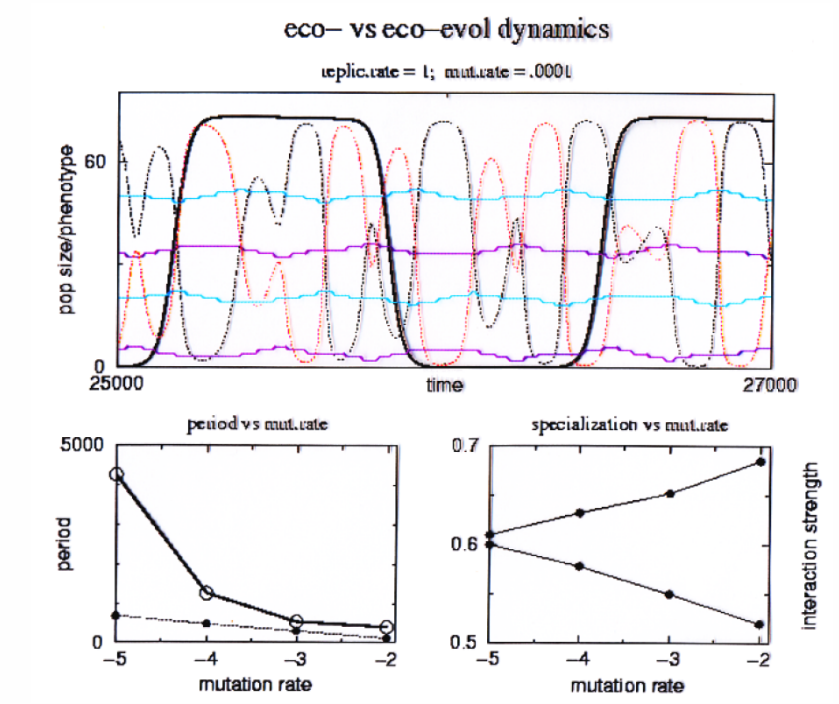


Figure 4.18: Ecological vs evolutionary dynamics in predator prey model by Van der Laan and Hogeweg (1995)

the purely ecological and ecological + mutation system are out of sync with each other in the ODE formalism. This counterintuitive finding happens because mutations affect how far predator and prey bands are away from each other: small mutations result in some separation between predator and prey, while high mutation allows close matching and more specialised predators. Taken together, this leads us to the following considerations:

1. There is a narrow parameter range to make ODEs biologically viable at an ecological time-scale (i.e. when time scales are separated and mutations don't occur).

2. The population dynamics of each species (the period of the system) are much faster in the evo-eco model, which is counterintuitive given that evolution is always considered to be a slow process. Instead evolution speeds up ecological dynamics, showing that these time scales interlock, and interplay.

Changing the width of the interaction of phenotypes (i.e. the width of the Gaussian, σ) also leads to interesting differences. As σ increases the evolutionary oscillations take longer. Moreover as σ decreases the system changes from a static system to runaway red queen dynamics (figure above).

The bottom line of this all is thus that we now have an **existence proof** of a counter example to fast ecological processes and slow mutational processes. We have shown conclusively that they can interact and that, in this case, larger disparities in speed means it is *more important* to include mutations, because the behaviour is then most different. Evolution could play an important and continuous role in *stabilising ecosystems*.

Adaptive dynamics (the study of invasion dynamics by completely separating ecological and mutational time scales) is thus not a realistic choice for modeling such systems. We cannot say beforehand that ecology and evolution do not interact or influence each other. In fact, it is more likely that they do.

Conclusion separation of time scales

It is abundantly clear by now that separation of time scales can lead to wrong ideas about evolution. We have seen from the first model that fitness is a time-dependent function. If you look at inclusive fitness over fifty generations, parasitoids with high B parameters did very well for themselves. However, over 300 (!) generations, it is clear that those with low B parameters are the long-term victors due to spatial pattern formation. Immediate fitness comes from chasing hosts optimally (high B), but long-term fitness comes from doing the exact opposite. The latter does evolve, even though it is detrimental in the short term! If we would not have looked on a longer time scale, we would never have understood why this happened.

In the second example, we saw clearly that adaptive dynamics (where one separates ecological and evolutionary time scales) is a poor reflection of reality. The model provides an existence proof of a different situation: mutations actually stabilise a fairly diverse ecosystem, that breaks down when mutations are stopped. Moreover, the less often mutations happened, the more important it was to include them (because the discrepancy between the system with and without mutations grew larger). This counterintuitive result clearly shows that we cannot intuit beforehand what we should do, and that separation of time scales because we presume they will not interact too much is a very bad idea.

We now turn to another subject entirely, though the interaction of time scales there is also important. We have looked at hypercycles, but will now look at another basic entity of life: cells or vesicles.

Vesicles, group selection, and the information threshold

In this section, we will take a look at models with a predefined higher level: cells. We have seen how higher levels can come into existence and protect systems from parasites. We will now see how this predefined higher level impacts on the information threshold and how selection pressures from different levels interact. We thus aim to answer the question: *Can group selection lead to stable coexistence of several species and so generate stability against parasites?*

Wilson's group selection model: an example that it can work in theory

Group selection has long been seen as a very controversial term because it was often formulated in terms of the good of the species (Wynne-Edwards, 1965). For the last couple of decades, there has been an explicit focus on the individual as "the" unit of selection, be it the individual genes or organisms. This individual selection is furthermore thought to always undermine group selection (Leigh, 1977; Alexander and Borgia, 1978). However, we have already seen some examples in which looking at selection at a single level was insufficient to explain the evolutionary dynamics: the outcome was determined by selection on characteristics of higher level entities (e.g. faster moving spirals, or waves that survive long enough to produce new waves). Recently, the debate around group selection flared up again when 3 scientists (evolutionary mathematical biologist M. Nowak, biomathematician C. Tarnita and ant-specialist and famous evolutionary biologist E.O. Wilson) published a paper in Nature (Nowak, Martin; Corina, Tarnita; Wilson, 2010), in which they claimed that the evolution of eusociality could better be explained by group selection than by **kin selection**: the famous explanation for social behaviour introduced by Hamilton and described in his Hamilton's rule (Hamilton, 1963, 1964). Their paper was met with a large amount of heated replies and a defence of their paper by the authors(Nowak et al., 2011; Ferriere and Michod, 2011; Abbot et al., 2011; Herre and Wcislo, 2011; Strassmann et al., 2011). The debate has still not been settled.

However, in work in the 1970s, D.S. Wilson formulated a group selection model in which he studied the conditions in which group selection could occur given the individual as the unit of selection(Wilson, 1975; Slatkin and Wilson, 1979). The model was formulated with explicit higher-level trait groups and selection within and between trait groups. The main idea was to look at the evolution of altruism and to separate kin and group selection: he needed a model that did not use relatedness. The model works by defining patches of space (trait groups). Within these patches, there are two types of individuals, who work according to the following equations:

1. $\frac{dX}{dt} = aXX – cX$

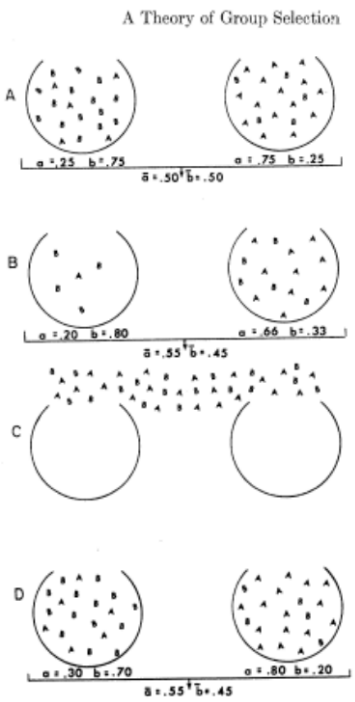


Figure 4.19: Wilson's model of group selection Wilson (1975)

2. $\frac{dY}{dt} = aXY$

We have X, which is an indiscriminate altruist: it helps both X and Y to replicate. One could think of this altruist as one who makes alarm calls: it hinders the caller, but aids all others of his kind nearby. Thus, X incurs a cost on replication because of its behaviour: c. If we were to simulate this as a full ODE, X would be removed from the population. It incurs a cost that Y does not, so it will never survive. However, we now have distinct patches, and the patches can only support so many individuals. Then, if $c < a$ it can persist.

After a certain growth period, individuals disperse to a random other patch, there is a binomial distribution of X and Y over the patches. Here comes the kicker: within any one patch, X loses out over Y. But patches with X and Y together will have more progeny in total! The help that X gives ensures this. Remember also that $c < a$, so helping also helps yourself. In this way, X will stay in the population, because patches with X will have more progeny.

There were many arguments leveled against this model. For example, dispersal from patches needs to be quick enough that X is not already out of the population. Another argument is that no organisms work like this. However, we might say that is not the point. Even if it is unlikely to find this situation in nature, the idea was the individual-level selection would always win out over group-level selection. Here, without invoking kin selection, Wilson showed that that need not be the case: an **proof of principle**.

What are the important insights from this model? The following:

1. Group selection works according to compartmentalisation of the population.
2. Kin selection is a parameter of group selection: it is not needed in this model.
3. Group selection causes a reversal of selection relative to individual-level selection: altruism evolves instead of non-altruism!

Next, we will consider how adding an explicit layer of groups affects a model of prebiotic evolution.

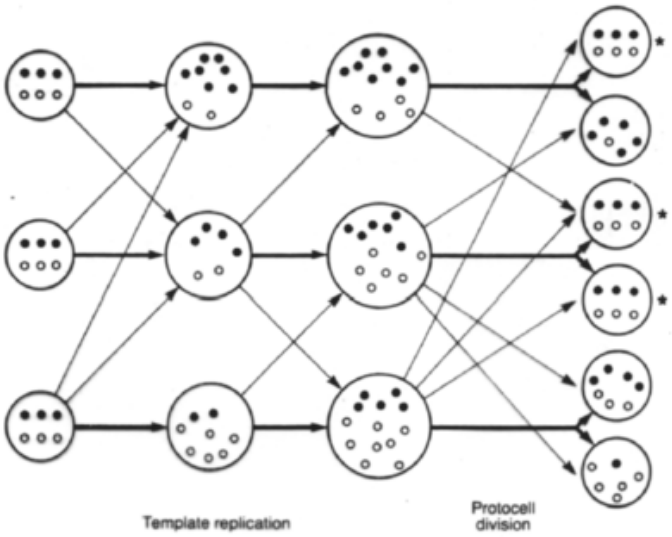


Figure 4.20: Simple illustration of the stochastic corrector model by Szathmary and Demeter (1987)

Stochastic corrector model: a vesicle-based solution to the information threshold

The potential role of group selection in the origin of life was studied by Szathmary and Demeter (Szathmary and Demeter, 1987). They asked whether group selection could help overcome the information threshold. They developed a model based on ideas similar to the model of D.S. Wilson, but then in a prebiotic setting. This model is called the **stochastic corrector model**. The model considers two explicit levels: molecules and vesicles (which contain the molecules). Note that this is again a hypercycle-like model, in that we look at ecological dynamics *without mutation*. The model considers two types of molecules, X and Y, which together form a replicase (heterodimer) that can replicate both X and Y, causing growth of the vesicles. Hence, vesicles need both X and Y to grow and grow fastest if [X] = [Y]. However, parameters are chosen such that X replicates faster than Y, making the system unstable. Vesicles grow and split without any interchange of molecules (this is thus an extreme grouping case, similar to completely isolated patches). In this process, however, vesicle composition could change due to stochastic processes during vesicle division and because of stochastic replicator dynamics (see below). Moreover, multi-level selection is explicitly implemented in that vesicles need both X and Y molecules to replicate (i.e. cooperation), but within a single vesicle, X outcompetes Y. The question here is then: *can group selection stabilise the system?*

The system of equations used is as follows:

1. $\frac{dX}{dt} = aX(XY)^{\frac{1}{4}} - dX - X((X+Y)K)$
2. $\frac{dY}{dt} = bY(XY)^{\frac{1}{4}} - dY - Y((X+Y)K)$
3. Whereby $a > b$

If one runs this as an ODE (without vesicles), Y is competed out of the population. X replicates faster, so X comes to predominate, and if Y dies out the system dies out (because the replicase is the heterodimer). In the real model, everything is discretised. This means that there are a finite number of replicators, and there is a random chance that the replicase makes an X or a Y. Thus, a cell that starts out with equal amounts could end up with unequal amounts upon replication. Additionally, there is randomness (stochasticity) upon division: if you have 5 X and 5 Y, you could, by chance, give 2 X and 4 Y to one daughter cell, and 3 X and 1 Y to another. Hence, there is the potential for *stochastic correction* of molecule amounts in daughter cells. The principle is shown in the figure below.

This principle mirrors the master equation that we encountered in studying the information threshold.

Here, the master vesicle is that with [X] = [Y]. All other types are 'mutants'. In this case, the fidelity of replication would be how randomly a vesicle, upon division, divides the replicators: the less random, the higher the copying fidelity of the vesicle. Note that there are no actual mutations here, and that there is a difference in time scales between vesicle dynamics and replicator dynamics that wasn't there in the master equation. Nevertheless, there is some similarity.

So what happens in this system? Although the two dynamics (intra- and inter-vesicle dynamics) are separated, they are statistically related in that the replication dynamics are used as a parameter in the vesicle dynamics. However, the population of vesicles is considered to be constant (no extinction). The model results show that both molecules can be maintained in the system due to group selection, and that even the master cell (which contains equal amounts of X and Y) can persist. However, this happens only if there are few replicators per vesicle. In other words, this system works when the number of molecules is small. Why? Because this makes it easier to correct excesses of either molecule by stochastic correction. If you have 6000 molecules of one and 4000 of the other, a little stochasticity on the redistribution to daughter vesicles will not matter much. But if you have only 6 of one and 4 of the other, the luck of the draw reigns supreme, and stochasticity is powerful! Vesicle-level selection is also less powerful when there are many molecules within vesicles: if you have two vesicles, one with [X] = [Y] = 10, and the other with [X] = 5 and Y = [10], the first is obviously much better. With such small numbers of molecules, a difference of 5 molecules is a huge relative difference. However, if you have two vesicles, one with [X] = [Y] = 1000, and the other with [X] = 1000 and [Y] = 1200 the relative difference between the two vesicles is smaller, even though the absolute difference is larger. Thus, selection on the vesicle level becomes weaker as the number of within-vesicle replicators grows, because small differences in small numbers of molecules have a relatively much stronger effect than differences in large numbers of molecules.

How does this influence the information threshold? The stochasticity/discreteness in the system means that the information threshold is crossed more easily. Why? In the ODE system, we allowed for populations of 0.0001 individuals. Now, that doesn't happen, and it is possible to lose the master vesicle completely. Similarly, limited diffusibility (when the vesicles don't move much from their initial location) lowers the information threshold, making it easier to cross. Why? Well, the stochastic corrector works because selection between vesicles keeps the master cell (with equal [X] and [Y]) in the population. However, if vesicles don't diffuse, who are master vesicles likely to be next to? Because they are the best at replicating, they will be positioned next to each other. Therefore, the master vesicle will not compete with an average population, but mostly with copies of itself. These latter points were not realised in the model since there was no mutation! Therefore, although the model shows that group selection (the higher level selection on vesicles) can help to maintain information given certain molecules, group selection only works under those conditions that worsen the case for information relative to the information threshold if mutation is included.

Note that vesicle death rate needs to be tuned, vesicles need to be small enough (for enough difference between them for vesicle-level selection to act upon), and there need to be enough vesicles in the system (for selection to have something to choose from).

Vesicles and the information threshold

At this stage the role of vesicles in the information threshold is still questionable. It can work, but stochasticity makes it easier to go over the information threshold. Also, time scales and the size of vesicles need to be tuned. Mutations were absent, but the information threshold exists due to them. Therefore, we wish to answer the following questions:

1. Can vesicles alleviate the information threshold when mutations happen?

2. Can group selection play a role with respect to the information threshold and overrule the negative effects of stochasticity due to vesicles?

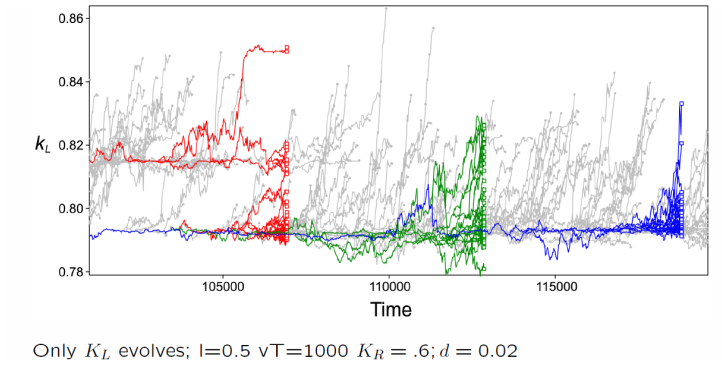


Figure 4.21: Ancestor trace of kL values of individual replicators within vesicles (different coloured lines). Within vesicles, kL increases. However, succesful vesicles are those with low kL at birth.

Micro- and Macrolevel dynamics: intricate implicit mutual interactions

In the previous sections, we have first seen examples of *emerging* higher level patterns on which selection could act (e.g. spiral waves in the hypercycles, and waves in the minimal eco-evolutionary model) and later examples of *predefined* higher levels (e.g. Wilson's trait groups, cells in the stochastic corrector, and vesicles). Next, we will try to directly compare the dynamics of emerging and predefined higher levels. We do so by looking at a modeling study by Takeuchi and Hogeweg (Takeuchi and Hogeweg, 2009b). This is in many ways the same model discussed in previous section.

On the micro-level:

1. Two types of molecules: replicators (R) and parasites (P), i.e. an RP-system.
2. Parasites are either in a folded or unfolded state. They can only be replicated if they are unfolded, but when they are folded they can perform a certain task, which in this model is assumed to be lipid production. Hence, parasites in the folded state enhance the growth of the vesicle they are in. The parameter l describes the fraction of the time the parasite is folded, the parameter kL how well it binds to the replicators (as before).
3. Ongoing mutations (in contrast to the stochastic error corrector, which had no mutations).

On the macro-level:

1. Explicitly defined vesicles, modeled in a CPM (see Cellular-Potts model earlier).
2. Growth rate of vesicle depends on the number of folded parasites inside.
3. Vesicles can divide once they contain a certain minimum number of molecules.

Results of this model first of all show a nice example of (stochastic) correction: an ancestor trace on the values of kL shows that although kL values might be quite high at any given time, the long ancestor has a relatively low value of kL (and hence is a relatively weak parasite) (see above). To understand the evolutionary dynamic better we apply a technique called ancestor tracing. An **ancestor trace** works as follows: at a certain time point, you sample the whole population. You then follow the evolution of certain parameters in the sampled individuals with that ancestor. In this way, you can see what happens to descendants of certain ancestors in the system. The explanation for the behaviour is that in the short term, stronger parasites are selected within the vesicles. However, in the long term, between-vesicle selection selects for vesicles that contain relatively weak parasites because these vesicles contain more replicators and hence grow faster. Stochastic variation between vesicles allows for this higher-level selection.

In the long run, we again find that there is a trade-off between kL and l (as we saw in the non-vesicle model) and that over time, both the value of kL and l increases. This makes sense, because folded

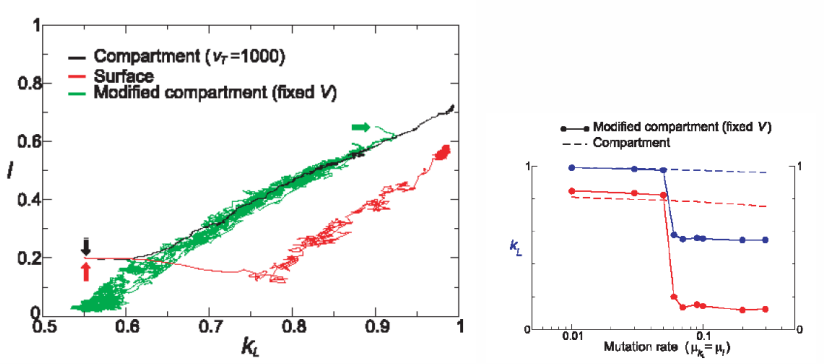


Figure 4.22: Vesicles show a similar trade-of as the surface model in the RP-system, but this trade-of can be reversed when the parasite gains a function for the cell growth.

parasites increase the growth rate of vesicles (and high $l \rightarrow$ many folded parasites). However, if this dependency of vesicle growth rate on folded parasites is removed, the direction of movement on the trade-off changes and instead vesicles with both low l and low kL are selected. This means that in this case, the selection for fast replication of the parasite dominates. This result is only found for high mutation rates.

Why does this happen? And why can this system still sustain itself (i.e. why aren't the vesicles killed by fast-growing parasites)?

To investigate this, we measure the death rates of vesicles. Two things are changed: the vesicle size is now constant (no growth), and vesicles perish when no replicator or parasite is inside. We consider these as a function of Δl, the distance from the Hopf bifurcation value of l, the value at which the parasites become too strong and the system becomes unsustainable. You can see this from the Figure **??**: though for different kL values it happens at different times, at certain l values the system breaks down. If you take the distance from this l value, Δl, then you can compare scenarios with different kL values.

If we do this, we see a remarkable thing occur. Note that in this modified model, vesicles do not grow (vesicle size is constant), so selection cannot act on growth rate. However, it can act to minimise death rate of vesicles. That is what happens. Compare low kL parasites (green, fast replication) and high kL parasites (black triangles, strong binders but folded most of the time). For high kL values, the death rate of the vesicle is lower than for low kL as long as you are before the bifurcation point ($\Delta l > 0$; see inset in figure), but *beyond* the bifurcation point ($\Delta l < 0$) the death rate is lower for low kL. Within vesicles, there is selection for stronger parasites and hence for lower values of l, i.e. vesicles will move to the left on the x-axis. If mutation rate is high, evolution is fast and hence the system will be close to the bifurcation point. Selection for lower death rate of the vesicles will then lead to selection of lower kL.

This is some tough material, so let us reprise: we now have pre-defined higher-level vesicles with a set volume. We know that a parasite would normally want to be replicated as quickly and often as possible. Thus, it would maximise kL and minimise l. However, we saw that in the scenario where you make vesicle growth dependant on the l value of the parasite (for example, you assume that the parasite in its folded state aids lipid production), l, the foldedness of the parasite, is increased over time. The concomitant increase in kL makes sure that if the parasite is unfolded it is replicated very efficiently. If you make vesicle growth independent of parasites (but make vesicle death contingent on having at least one replicase or parasite inside), something different happens. Instinctively, you would think the parasite would be free to do as it pleases: decrease l while leaving kL high, thus getting maximal replication, with the stochastic correction of vesicle division saving the whole system frome extinction.

That is not what happens. Instead, kL is minimised (if mutation is high!). Why? If kL is smaller, the increase of death rate when going over the Hopf bifurcation (left of Δl) is lower. Just before the Hopf bifurcation (when $\Delta l > 0$), higher kL values have lower death rates (see inset in figure; note that the scale of the y-axis is much smaller). In other words: if l is mutating all the time, and mutates fast, the

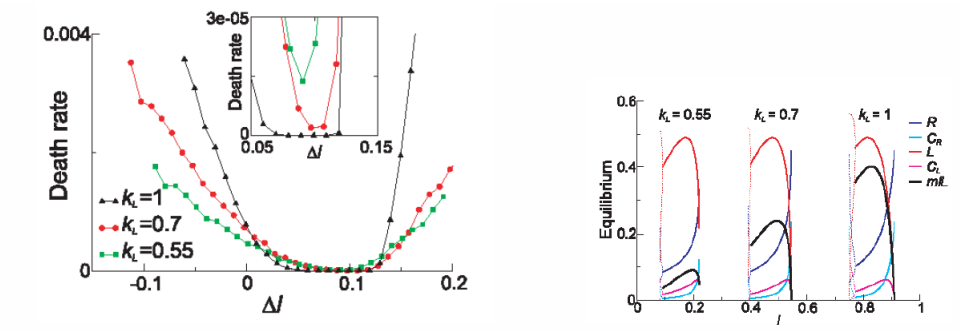


Figure 4.23: Death rates of vesicles close to the hopf bifurcation (=information threshold) reveal that, even though low kL results in higher death rates before the hopf bifurcation, is lowers the death rates when the system passes the hopf bifurcation. Evolution has thus become concerned with not dying when surpassing the information threshold.

chance that it mutates to a value that can kill the vesicle is high. The stochastic error corrector can save the day, and if kL is lower (the parasite associates less vigorously with the replicases), the vesicle survives for longer, and therefore has more chance of having progeny.

Thus, we see that kL is minimised because evolution proceeds to the **border of order**. If mutation rate is high, then many parasites will mutate over the border of order, into a regime where the vesicle could die. By minimising kL, death rate of the vesicles is minimised. Stochasticity in the system is maximised: by being as close as possible to the Hopf bifurcation, random mutations have the largest effect. We see here also that there is **survival of the flattest** at high mutation rates: in this context it means that vesicles with minimal kL have a slower rise in death rate over the **information threshold**/Hopf bifurcation, so the flatter this increase of death rate, the better they can survive. So, in a regime with low mutation rates, there is a delicate balance between internal and external dynamics. In a regime with high mutation rates (and set vesicle size), the internal dynamics change, and the system becomes concerned with not dying so fast once it eventually goes over the **information threshold** (Hopf bifurcation).

What does this model tell us about explicitly defined versus emerging higher levels?

Comparing this model to the non-vesicle case:

1. Explicit higher level entities (vesicles) are less stable than emerging higher level entities (waves), especially at high mutation rates. That is logical: waves form by themselves, so they have to be stable or they wouldn't form. Vesicles are imposed, and therefore only stable given certain conditions.

2. Stochasticity is maximised and used by evolution, both for correction and tuning. In other words: it is used to have the best selection between cells, in this case by bringing cells close to the Hopf bifurcation, so that those going over have the most dire consequences.

3. Implicit interactions (in explicit multilevel models) can automatically mutually tune the parameters (in this case, death rate).

One step further: evolutionarily stable disequilibrium in a stationary population

We now look at more recent work of Takeuchi, Kaneko, and Hogeweg (Takeuchi et al., 2016). Here, the previous model has been simplified. Now, there is no explicit parasite, simply two replicators, R and R', that need each other to replicate. Additionally, vesicles are not CPM vesicles, but just compartments in which there are replicators. The internal dynamics are once again ODEs which, if left to their own devices without other selection pressures, would lead to extinction. This is because replication requires two replicators to form a complex, so that one of them may be replicated (template-mediated replication). The replicated molecule is the one that served as template, but its parameter for serving as a catalyst,

k, which initiates complex formation, can mutate a little. Being the molecule doing the replicating is altruistic: while you are replicating, you cannot be replicated. You would thus expect that, for both replicators, catalyst propensity decreases all the time. This would lead to ever-decreasing replication. As the vesicles need them to grow, they would then die out (Figure 4.24).

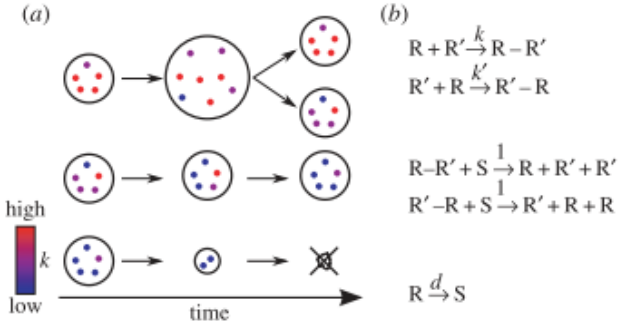


Figure 1. Schematic of the model. (*a*) Protocells containing replicating molecules (substrates are not shown). Colours indicate the catalytic activity of molecules k. A protocell with high-k molecules grows and divides (top) and that with low-k molecules shrinks and dies (bottom). Molecules within a protocell evolve toward decreasing k (middle). (*b*) Reaction scheme. R (R′): replicating molecules, R−R′ (R′−R): complex, S: substrate. Each replicating molecule is assigned a unique complex formation rate $k \in [0, 1]$. Any pair of molecules can form a complex. Each pair can form two distinct complexes depending on which molecule serves as a catalyst or template, as indicated by prime symbols (top). The complex formation rate is given by the k value of the catalyst. Replication produces a new molecule, whose k value is copied from the template (middle). This k value is slightly modified by a mutation with a probability m per replication (see Model). The k values of all molecules were initially set to unity at the beginning of each simulation. All molecules decay at the rate d (bottom). $m = 0.01$ and $d = 0.02$ unless otherwise stated. (Online version in colour.)

Figure 4.24: Schematic that describes the workings of Takeuchi's model. Taken from (Takeuchi et al., 2016).

Similar to the previous models, there is thus a minimisation of catalysis within the vesicle (complex formation is selected against), but a maximisation of catalysis between cells (the best vesicles are those with replicators with high rates of catalysis). This provides two opposing selection pressures. Of course, the rate at which an extinction can happen depends on mutation rate, which governs how quickly replicators can change their catalysis rate. A new feature is that there is competition over an explicit resource, which diffuses over the system, and is used up at replication (think of nucleotides). The vesicle, or, protocell volume is equal to the resource + the molecules within it.

The model works in steps: first, many reactions within protocells are allowed to occur. Then, substrate diffuses over protocells, with a probability proportional to the number of replicators (so cells with more replicators have a higher chance of getting more substrate). Replicators do not diffuse. Then, there is protocell or vesicle division: if a protocell has more than V particles (substrates + replicators) it divides, and its particles are randomly split between daughter cells. This serves as a version of the **stochastic corrector**.

Evolution hidden behind a seemingly static population

The first thing to check in the model is how the different selection pressures interact. This can be done most easily by varying V. As we mentioned previously, vesicles with larger within-vesicle populations of molecules cause the within-vesicle selection pressure to be dominant. Firstly, this is because stochastic

differences in large populations of molecules are small, and secondly because larger volumes divide less often. In small vesicles (when division already occurs at low V), between-vesicle selection is much stronger: stochastic processes at division and low molecule numbers mean that, by chance, some vesicles will be much better than others. The dependency of average k (catalysis propensity) on V clearly shows this relation (Figure 4.25).

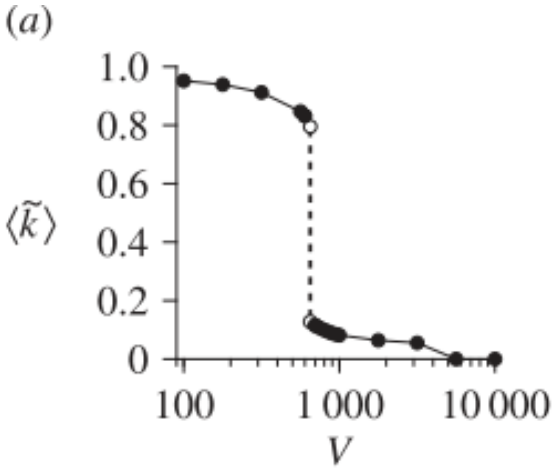


Figure 4.25: Dependance of the average k parameter on the division volume V of vesicles. Note the region in the middle where the higher-level and lower-level selection pressures are matched. Adapted from (Takeuchi et al., 2016).

At intermediate values of V, the two selection pressures are about equally strong, and the average value of giving catalysis k is more or less constant. However, this statistical rigidity of k belies a tumultuous underlying evolutionary process. In Figure 4.26, we see an ancestor trace of a certain cell for $V = 1000$. In black you see the normalised protocell size (cell size/V, right axis) for a protocell along the line of descent (i.e. over the generations). Circles are cell division events. The average value for k within the protocell is the red line, and the orange area surrounding it shows the range of values that k can take for all replicators in the cell.

Let us first focus on k. Over the generations, the average value of k (red) in a protocell declines, until it suddenly shoots back up again. This process is repeated continuously. At the same time, you see that as k becomes lower, the normalised cell size of the protocells only just manages to skirt past 0 (the black line almost hits the x-axis). It goes back up again just as k shoots up as well.

So how does the system survive? There is continuous downward evolution of k. As k decreases, cells become smaller. Small cells have a lot of stochasticity upon division. Purely stochastically, you can thus have one with only good replicators (high k values), which will be selected. Since this all happens by chance, note that most cells perish, and only the odd cell that manages to score the jackpot survives, immediately increasing the average k value and beginning the process anew. This process happen not at the cellular population level, but at the level of replicators within individual cells. Here, we look at an ancestor trace of one such lucky protocell, whose ancestors continuously got replicators with high k values upon division.

If division volume is smaller ($V = 316$), cell size reaches less catastrophic lows, and k is much more constant. This is consistent with the fact that the stochastic correction effect is stronger at smaller cell sizes, and that between-vesicle selection is stronger if there are less within-vesicle molecules (Figure 4.27.

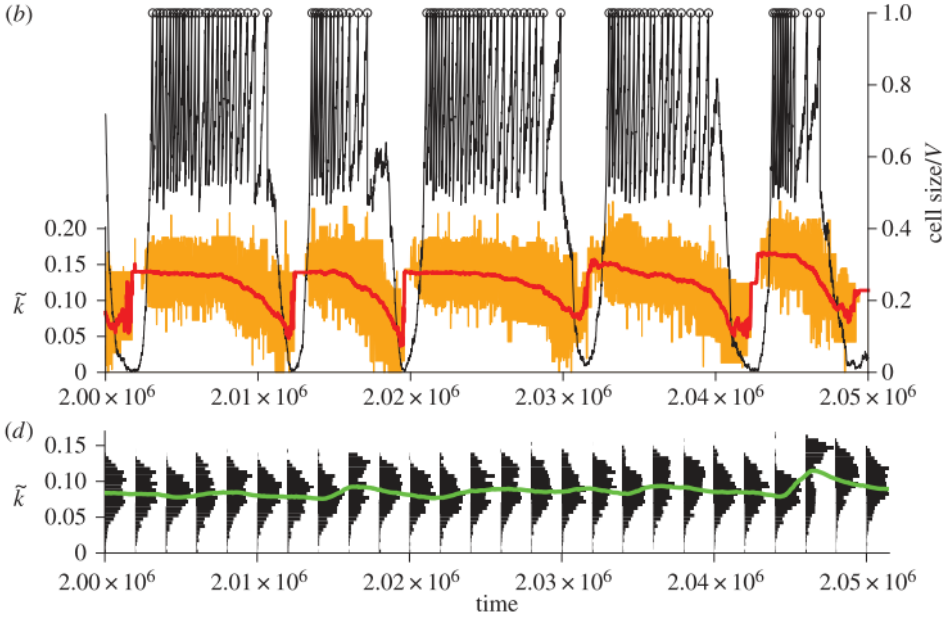


Figure 4.26: Top: Changes in average k values per cell (red), range of k values in the replicators within a cell (orange), and normalised cell volume (cell size/V) (black) over time. Bottom: population-level k averages over time. Target volume V for division is 1000. Taken from: (Takeuchi et al., 2016).

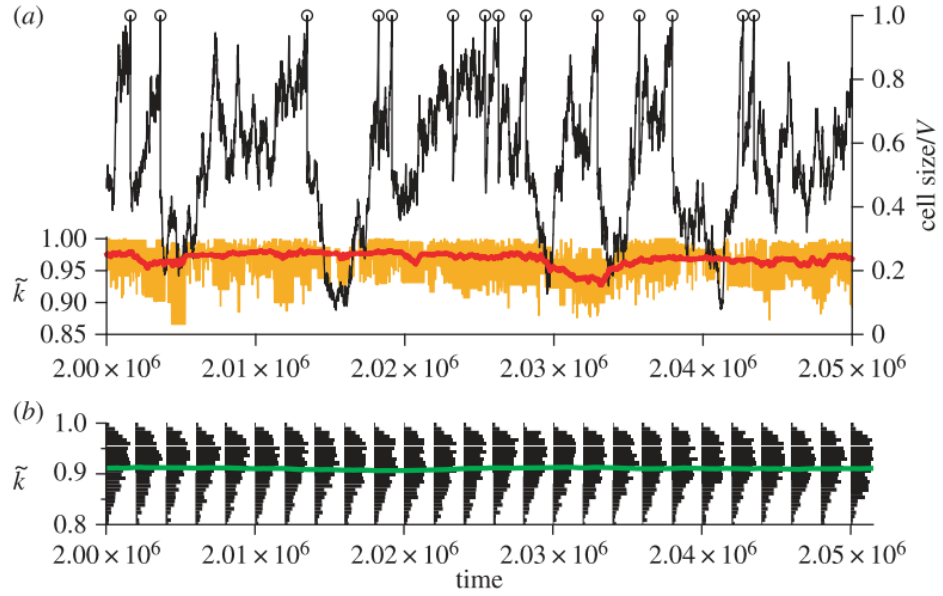


Figure 4.27: Top: Changes in average k values per cell (red), range of k values in the replicators within a cell (orange), and normalised cell volume (cell size/V) (black) over time. Bottom: population-level k averages over time. Target volume V for division is 316. Continual movements to near extinction are prevented by the higher strength of vesicle-level selection. Taken from: (Takeuchi et al., 2016).

You can wonder whether it is a good thing that cells almost die all the time. It does not sound like a particularly great idea that there are continuous bottlenecks where most cells die. On the other hand, it might help survival of the system as a whole at larger division volumes. If one kills protocells that are too small, the protocells with replicators that are driving themselves to extinction are filtered out. You

might think that removing bad protocells could help the system. However, it also prevents stochastic correction! As it turns out, if protocells that become too small are removed, the system collapses at much earlier values of V than if the evolutionary oscillations (low k, bottleneck, stochastic correction, higher k) are allowed (Figure 4.28). Additionally, the distribution of cell sizes in the population is one with many small cells (that can rescue the system) and some large cells (whose k values fall over time). The arrows denote the two V-values discussed above (1000 and 316).

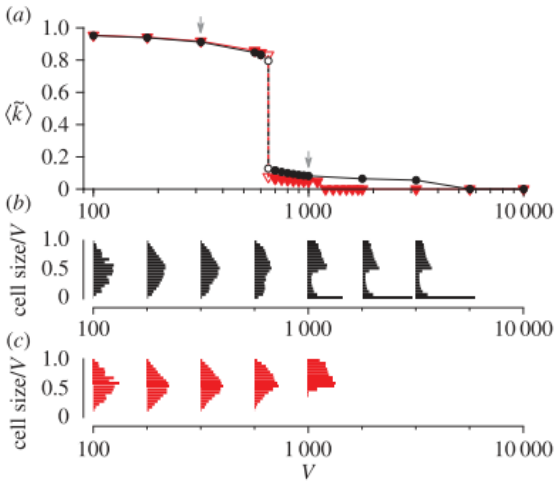


Figure 4.28: Top: Population Average k values for a range of division volume (V) values. Arrows denote the values 316 and 1000 (discussed above). Bottom: distribution of normalised protocell sizes in the system at different values of V. Red: protocells with volumes below a certain threshold are removed. Black: all protocells, no matter how small, are kept in the system. Taken from: (Takeuchi et al., 2016).

The specific conclusion of this story is thus that the oscillatory evolutionary dynamics contribute to the survivability of the system, and that the system can survive only when the two selection pressures can somehow remain in balance (Figure 4.29). A more broad interpretation of this result is that when selection pressures are matched, evolution is uniquely poised to find creative solutions.

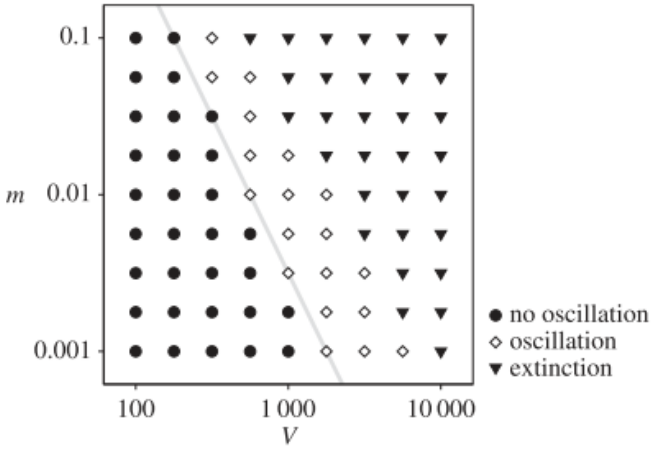


Figure 4. Phase diagram with respect to V and m. The boundaries between the parameter regions where the evolutionary oscillation occurs (diamond) and where it does not occur (circle) has approximately the same slope as that of $mV^2 = \text{const.}$ (grey line); extinction (triangle).

Figure 4.29: Parameter ranges (division volume and mutation rate) and the resultant dynamics of the system. When within-vesicle selection and between-vesicle selection are matched, oscillatory dynamics occur. Note that mutation rate affects within-vesicle dynamics (governs how fast the catalysis parameter k can mutate), whereas V affects both between- and within-vesicle selection (regulates strength of the **stochastic error correction**, see Figure 4.25). Taken from: (Takeuchi et al., 2016).

Evolutionary advantages and properties of more RNA-like replicators

Until now, we have looked at replicators. We have talked about the RNA world and called our replicators RNA-like. However, they are not truly RNA-like, in more ways than one. We have just looked at a system with two replicators, where either could be the template in a complex. In reality, however, what is replicated upon replication is the complementary strand of the template, not the template itself. The plus strand is a template for the minus strand and the other way around. What would happen if we introduce plus and minus strands into our models? Does this matter for the dynamics, and what situation evolves? Something else we have not yet considered is the shift from a RNA world to a world that contains DNA. Besides chemical constraints, we must marvel at the fact that somehow a system that has catalysis and replication in one molecule (RNA can do both) shifts to a system where there is a molecule that only carries information (DNA) and a molecule that only performs catalysis (or codes for proteins, as is the case nowadays), RNA. If we think that DNA somehow evolved from RNA, then why would an RNA ever devote itself solely to catalysis? As we have seen, being replicated is where it's at, and giving catalysis to others is a loser's game that can only be enforced by higher levels (whether emergent, like waves, or predefined, like protocells). We are going to take a look at these questions, and see some surprising results. First, let us look at what happens when plus and minus strands are added to the model.

Symmetry breaking

We take the last model we discussed, and add in the reactions and parameters needed for complementary replication (Takeuchi et al., 2017). What does this mean? The figure below explains it best. There are now 4 replication (k) parameters: plus to plus, minus to minus, minus to plus, and plus to minus. All these can evolve, and different complexes can form, where different strands can be either template or catalyst.

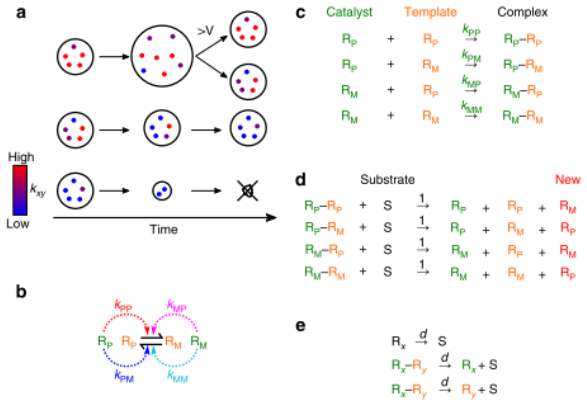
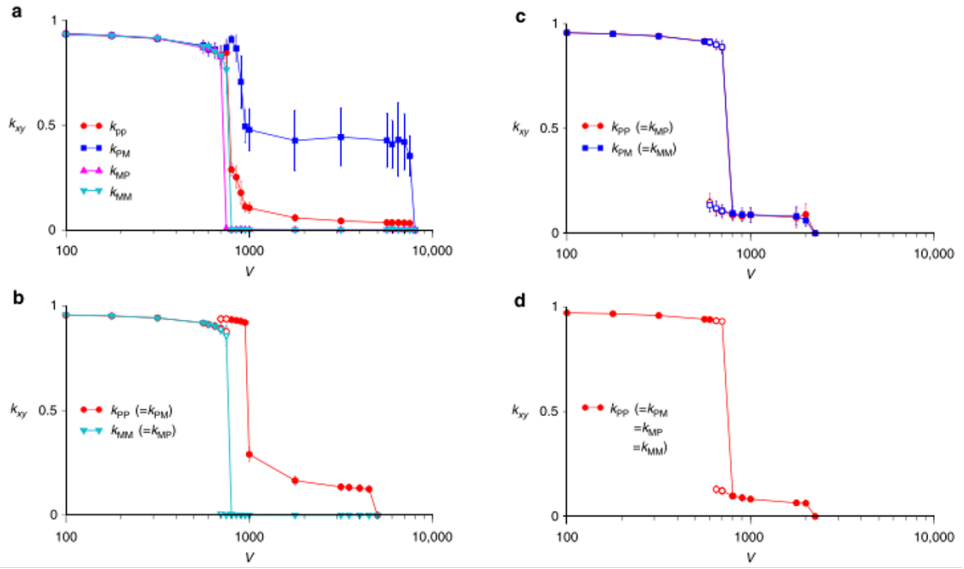


Fig. 1 Schematic of the model. **a**. Protocells containing replicators (substrates not shown). Colours indicate the catalytic activity of replicators k_{xy}. A protocell with high-k_{xy} replicators grows and divides (*top*); that with low-k_{xy} replicators shrinks and dies (*bottom*). Replicators within a protocell evolve towards minimising k_{xy} (*middle*). **b**. Schematic of replication and catalysis (see also **c** and **d**). *Solid arrows* indicate replication (template→product); *dotted arrows* indicate catalysis (catalyst⇢reaction). **c**. Complex formation. R_x denotes a replicator ($x \in \{P, M\}$); $R_x - R_y$, complex between R_x serving as a catalyst (*green*) and R_y serving as a template (*orange*) ($y \in \{P, M\}$). The complex formation rate is given by the k_{xy} value of the catalyst. **d**. Replication. S denotes a substrate. A newly-synthesised replicator (*red*) is complementary to the template (*orange*), with its k_{xy} values copied from the template with possible mutations. **e**. Decay. $d = 0.02$, unless otherwise stated

Figure 4.30: The model with **symmetry breaking**. Now, different strands have different catalysis parameters for the different reactions: plus to plus, minus to minus, plus to minus, and minus to plus. Substrates (S) are needed for replication. Taken from: (Takeuchi et al., 2017).

The question is: what do these extra degrees of freedom add to the system? As it turns out, they add a lot. If the parameters are allowed to mutate freely, one becomes very high, one has some positive value, and the others are extremely close to 0 (though never 0 outright) (Figure 4.31). What we observe here is a phenomenon called **symmetry breaking**: one of the strands gets high catalytic rates (in this case, we designate that strand as the plus strand, you can see that the catalysis of plus to plus k_{PP} and the catalysis of plus to minus k_{PM} are relatively high) whereas the other strand gets values that are extremely close to 0 (k_{MM} and k_{MP}), and the amount of the strands also drastically diverges: the plus strand is abundant, whereas the minus strand is very rare (not shown). You can see that survival in this system is possible for higher values of V than if symmetry breaking is not allowed at all (Figure 4.31d), or if it is limited in some way (Figure 4.31b and c).



Equilibrium average catalytic activities (k_{xy}) as functions of cell size (V). The values of k_{xy} were first averaged over all replicators at each time point. Then, the average (symbols) and s.d. (error bars) of k_{xy} over time were calculated after equilibration. The *open symbols* indicate metastable states (i.e. states reachable from different initial conditions with no state transition observed within 10^7 time steps; a replicator decays approximately with a probability d per time step). For visibility, the non-catalytic strand (if it evolves) was always taken to be the minus strand. Parameters: mutation rate (m) = 0.01, d = 0.02, N = 50V (total number of particles in the system is 50 * division volume). **a.** The full model (no symmetry imposed). **b.** The model in which kinetic symmetry is imposed (i.e. $k_{PP} = k_{PM}$ and $k_{MP} = k_{MM}$). **c.** The model in which functional symmetry is imposed (i.e. $k_{PP} = k_{MP}$ and $k_{PM} = k_{MM}$). **d.** The model in which both kinetic and functional symmetry are imposed (i.e. $k_{PP} = k_{PM} = k_{MP} = k_{MM}$). Adapted from Takeuchi *et al.* (2017).

Figure 4.31: Taken from: (Takeuchi et al., 2017). (note that b and c are not discussed in this reader)

This is basically the same trick as before, but now, it is much more efficient because of specialisation (you see that the range of volumes that the protocells can survive is much larger). Also, this arrangement increases the power of mutations: a very small mutation in k_{PP} has a huge effect on intracellular replicator concentrations, because there is so much plus strand that a small change in its catalytic rate translates into many more reactions.

It is very interesting that this happens, because there is no explicit selection pressure for **symmetry breaking**: at low cell volumes (V), between-cell selection prevails and all catalytic rates are maximised. At very high cell volumes, all catalytic rates are minimised (within-cell selection prevails) and the system dies out. What we see here is again a creative evolutionary solution when two levels of selection are matched in strength. This creative solution is somewhat similar to the emergence of DNA as a separate information carrier: one strand loses (almost) all catalytic activity (more like DNA), while the other gets very high catalytic rates (more like a catalytic RNA or proteins).

Symmetry breaking in a spatial model

We have seen **symmetry breaking** in a model with imposed multiple levels (protocells versus replicators). However, we first encountered higher-level selection pressures in the context of emergent waves. Can symmetry breaking also occur spontaneously? This is a question that von der Dunk *et al.* investigated (von der Dunk et al., 2017). The previous model happened in protocells that occupied no real space. Now, replicators are in a CA grid, where complex formation can happen when a binding partner is next to you, and replication can happen when there is an empty space next to you.

In this model, the diffusion parameter (D) is analogous to the volume parameter in the protocell model. Why? We have seen (all the way in the beginning of waves in CA models) that adding diffusion actually increases wave size and thereby decreases the amount of waves and increases the amount of replicators per

wave. Thus, more diffusion means more replicators per wave and less waves to select between, thereby strengthening replicator-level selection relative to higher-level selection, just as increasing protocell volume does this. So, what happens to the catalytic parameters in this spatial model for different levels of diffusion?

In the model with all degrees of freedom (all 4 parameters can evolve freely), **symmetry breaking** also occurs in space (Figure 4.32). A difference here is that in space, it is not so that at low diffusion values, all catalytic parameters are high, as was the case in the protocell model. That is because those values were enforced by the higher level, but here, that higher level first needs to emerge. It only emerges once higher diffusion rates cause die-off of replicators and empty space results. Again, the model without different catalytic rates (replicators without strands) goes extinct much earlier.

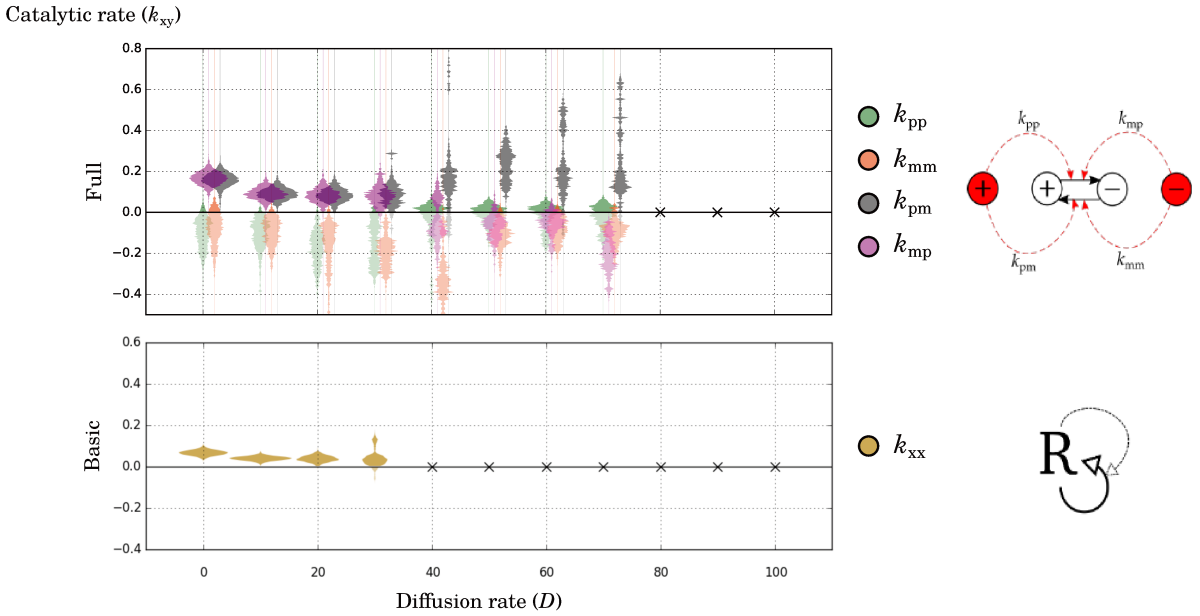


Figure 4.32: Evolved steady-state values of catalytic parameters for different diffusion (D) values in space, for different models (ranging from completely independent catalytic parameters to completely constrained catalytic parameters). Increasing diffusion rates are analogous to increasing division rates in a protocell model. The system is most resilient if all catalytic parameters are allowed to evolve freely, obtaining optimal benefit from **symmetry breaking**. Taken from: (von der Dunk et al., 2017).

Of course, these are steady-state catalytic rates. It is interesting to see what happens in the transient, and how exactly evolution gets these results. This is displayed in Figure 4.33. Since there is a lot to see in this figure, let's go through it step-by-step.

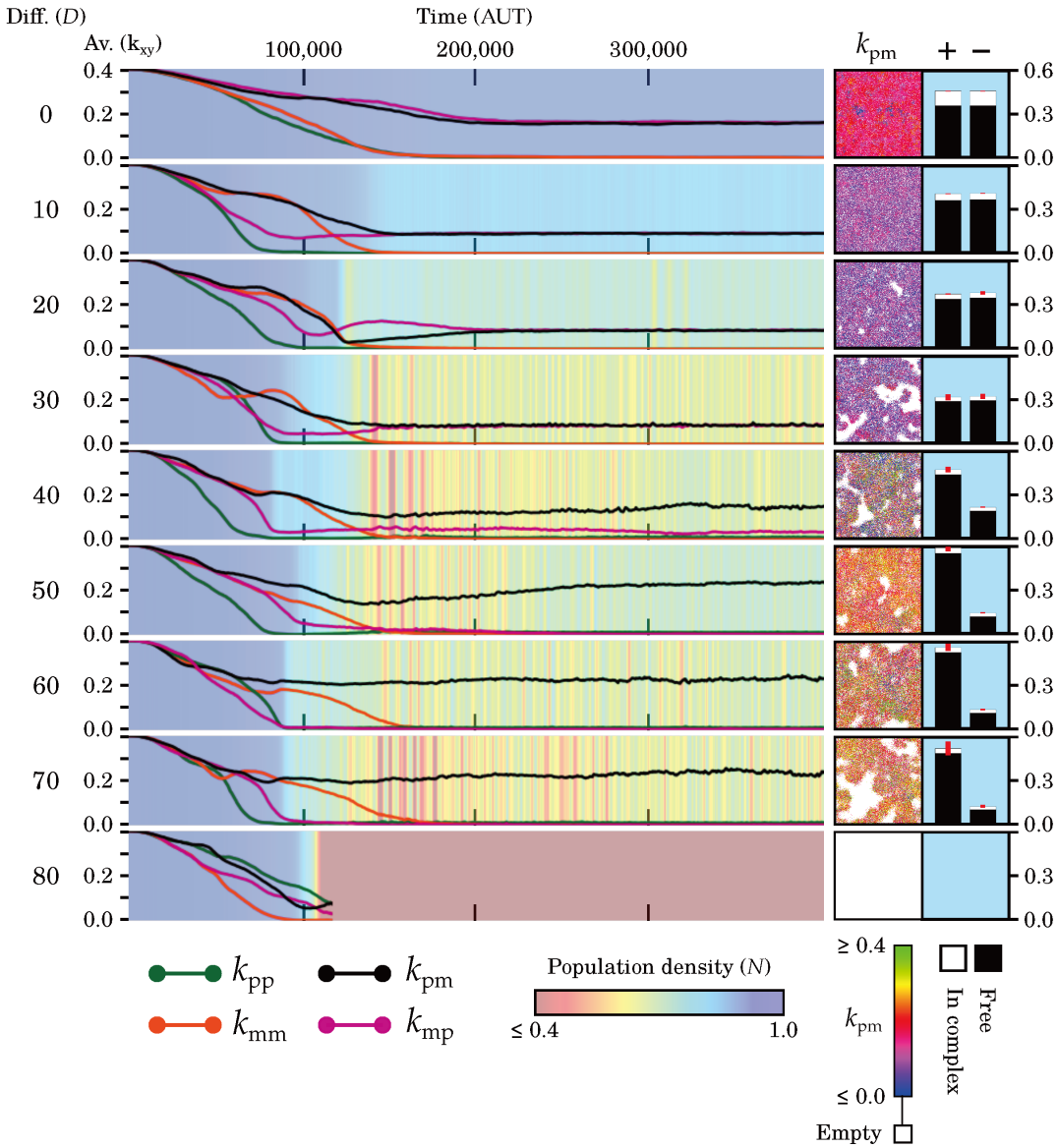


Figure 4.33: Evolutionary trajectories of catalytic parameters, population dynamics of strands, snapshots of the model space, and bar plots indicating rates of bound and unbound plus and minus strands. Taken from: (von der Dunk et al., 2017).

On the far left side are the diffusion values used. Right beside that are the averages of k_{xy} in the population, shown as lines. Each colour represents a certain combination of x and y. These lines are plotted against a backdrop that shows the total population density of the four molecules in the simulation, where cooler (more blue) colours indicate higher population densities, and warmer (more red) lower population densities. Right beside that are snapshots of the field, where the catalysis of the plus to the minus strand is visualised, along with bar plots that show what proportion of the plus and minus strands are free and in complex. Note that whichever strand gives lots of catalysis is denoted the plus strand.

Emergence of waves directs evolution

So what can we see? Until a diffusion rate of 20 there are almost no empty patches in the field (see the snapshots on the right), and replicators mostly compete locally, so that replicators with very low catalytic rates drive themselves to extinction locally.

If diffusion is higher than 20, individual-level selection first starts to win and replicators die out. However, this produces empty space, which creates a novel selection pressure for invasion of this empty space. Invasion of empty space is achieved by rapid replication (i.e. high catalysis), which can oppose the selection for decreasing catalysis on the strand level. Thus, emergent higher-level wave patterns (**mesoscale patterns**) counter individual-level selection pressure. You can see by the background that higher diffusion values coincide with much larger fluctuations of population density, which illustrates this point: individual-level selection wins, replicators start dying off, and the available space creates a new selective pressure. Interestingly, you can also see a development through time in these plots: the die-off is most extreme in the beginning (stretches of dark red to the left), and levels off over evolutionary time (this is most visible for a diffusion of 40 or 70: there are wide blotches of dark red in the beginning, while later, there is only yellow or light red, indicating less severe die-off of replicators). This means that the two selection pressures balance each other and the system stabilises itself over time.

If we look to the right at the rainbow-coloured snapshots, you see that higher diffusion values allow more diversity in the catalytic parameter. Why is this? This is similar to the **RP system**. There, there was selection for replication at the wave front, while the parasite advanced at the back. In this case, there is a selection for high replication (lots of catalysis) at the wave front, while low catalysis is favoured at the back (so that you get replicated more than you are doing the replicating). Depending on where you are, the selection pressure is thus different, and this causes more diversity in the parameters. In the long term, this can go so far that the difference between replicators at the front and the back of the wave becomes larger than just a few mutations. Then, two different lineages form. You can see this in the first figure we discussed. When $D = 40$ and $D = 60$, you can see a clear division of the k_{pm} parameter into two hubs: large for the wave front, small for the back (Figure 4.32).

Different types of symmetry breaking

You can see that until a diffusion value of 30, something appreciably different happens to the catalytic rates than after that. Before it, k_{pm} and k_{mp} take about equal (high) values, while after it, all values except k_{pm} are minimised. This is reflected in the proportions of plus and minus strands in complex: after a diffusion value of 30, the amount of minus strand falls drastically. The former behaviour is what we can call *directional asymmetry* or *reciprocal asymmetry*: both strands preferentially like to catalyse their complement (plus catalyses minus and minus catalyses plus, i.e. both k_{pm} and k_{mp} are high). This behaviour happens when small-scale interactions dominate (as is true for low diffusion). Then there is *catalytic* and *template* asymmetry: where one of the strands is a better catalyst or template, respectively. Both these asymmetries are seen between diffusion values of 40 and 70 at equilibrium of the system: the plus strand is the superior catalyst, the minus strand the superior template.

Evolution of DNA in an RNA world

Some earlier work explicitly studied the emergence of DNA in an RNA world. Why would DNA ever evolve? Hogeweg, Takeuchi, and Koonin tackled this question in 2011 (Takeuchi et al., 2011). Note that they focused purely on an *informatic reason*. That is to say that they ignored the fact that DNA could be a more stable molecule, or that there could be some chemical benefits or constraints. They purely looked at catalytic rates and information storage. In this system, there are two polymerases. One can make a string of DNA, the other can make a string of RNA. DNA can only be a template, RNA can be both template and catalyst. This gives rise to four different reactions:

1. RNA-dependent RNA synthesis
2. RNA-dependent DNA synthesis
3. DNA-dependent RNA synthesis
4. DNA-dependent DNA synthesis

Parameters such as the ability to recognise DNA and RNA can mutate, and a rare mutation can change a DNA polymerase into a RNA polymerase. There is a common resource that is used up for replication. Note that the model is working with many simplifications: there is no real strandedness, all DNAs are

just general DNAs and all RNAs are just general RNAs (i.e. though parameters of molecules can change, there is no underlying sequence), a rare mutation can cause a conversion from an RNA polymerase to a DNA polymerase, and the ability to recognise DNA is initialised as a neutral property in the population. Nevertheless, it can be informative to observe what happens given these simplifications. Two systems are compared: a vesicle-based system, with strands within vesicles modeled within a CPM framework (i.e. deformable vesicles), and a system in space.

Let's first survey the possibilities. In the image below, you can see that within the bounds of this system, three possible situations can occur (Figure **??**). The first is the starting situation, where an RNA-dependent RNA polymerase synthesises RNA. The situation in the middle is the transcription system: in the lower left, you see a DNA-dependent DNA polymerase (made of RNA), which catalyses the reaction to make more DNA-dependent DNA polymerase (made of DNA) and the reaction to make more DNA-dependent RNA-polymerase (made of DNA). Thus, this RNA uses DNA as template to create more DNA. Then, in the top left, there is the DNA-dependent RNA-polymerase (made of RNA), that catalyses the creation of more of itself (from DNA) and the generation of more DNA polymerase (from DNA). The last possibility is one with reverse transcription, where DNA can be made from RNA.

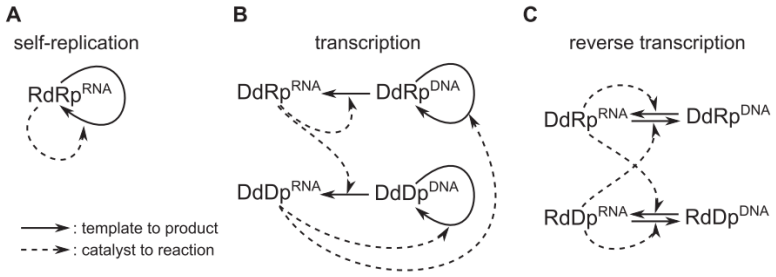


Figure 4.34: Modes of replication in the model. Taken from: (Takeuchi et al., 2011).

Although this might seem a bit confusing, the important point here is that systems including DNA immediately seem more cumbersome: they need more molecules to complete the same task of replication. Both system B and C require four types of molecules for replication. If the total concentration of molecules remains constant, this means that replication is slowed in these systems compared to self-replication depicted in Figure 4.34A. However, the saving grace of DNA could be that it takes away the implicit evolutionary battle between templates and catalysts: if RNA is both the template and the catalyst, there is a pressure to lose catalysis (thus being the template more often), which, if left unchecked, leads to extinction (as we have seen time and time again). Perhaps another (higher level) selection pressure can help the system survive (waves or vesicles)). If any parasite were to evolve that only serves as template, it would be supremely fit. This trade-off does not exist in the DNA system: RNAs do not need to be templates, so there is no problem if the DNA encodes for continuously better catalysis, and the DNA is the template all the time. Note, however, that this is not the case in the reverse transcription system: there, RNA still needs to serve as the template, so this trade-off is not alleviated, or only to a lesser degree.

The spatial model: co-existence of transcription and self-replication

We now look at what happens in the spatial model. The system was initialised with RNA-dependent RNA polymerases and parasites. When it was at equilibrium, the rare mutation from an RNA polymerase to a DNA polymerase was allowed, and the resultant evolutionary dynamics are shown (Figure 4.35). Rrec and Drec are recognition of RNA and DNA, respectively.

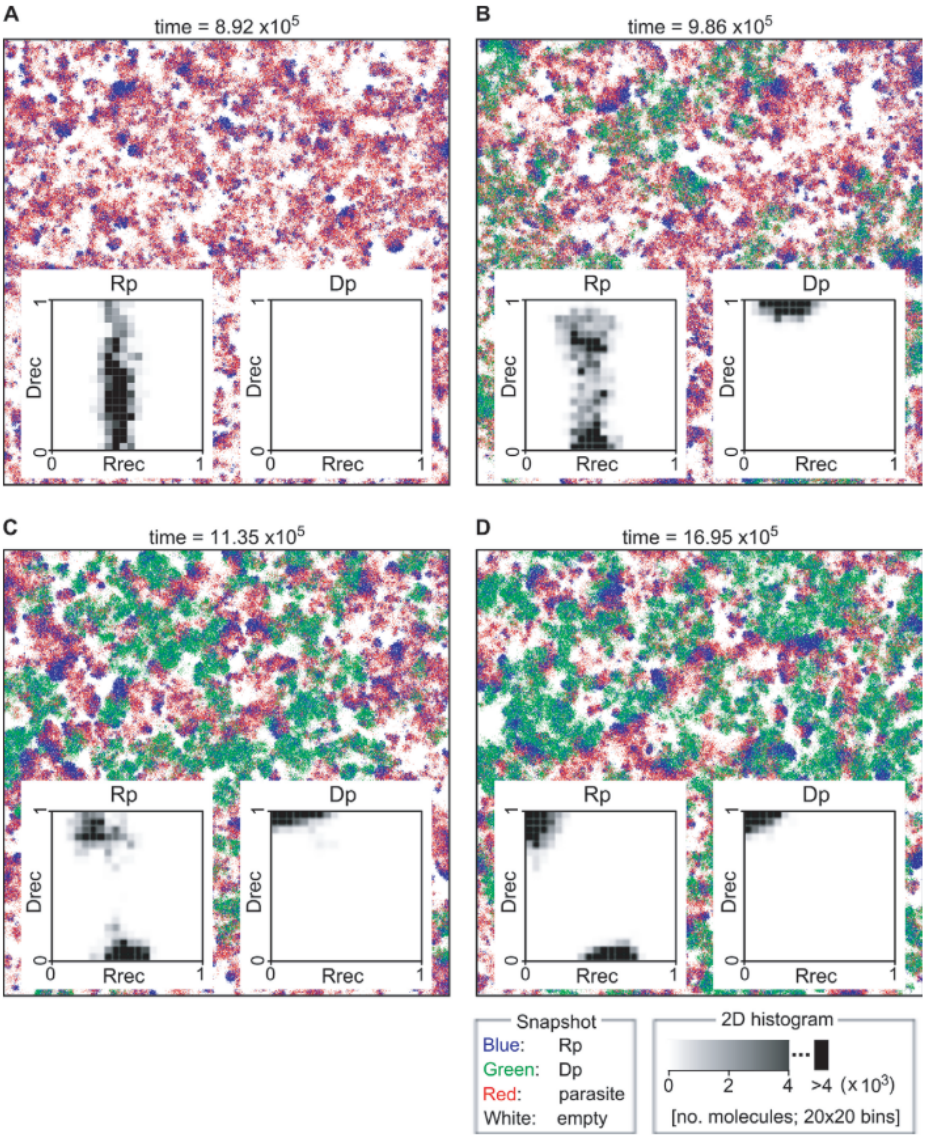


Figure 4.35: CA model of DNA and RNA replication. Taken from: (Takeuchi et al., 2011).

So what can we see? Over time, two subsystems evolve. The transcription system evolves, but the RNA system remains as well. You can see in the lower right that there are DNA polymerases that almost solely recognise DNA, and that RNA polymerases have split into two groups: one has lost the ability to recognise RNA (upper left corner; transcription system), the other recognises RNA. This is interesting: initially, the DNA replicase that evolves had dual specificity (you can see that DNA and RNA recognition exist in these molecules (Figure 4.35B)), but evolution swiftly causes specialisation. Interestingly, the transcription system could not supersede the self-replication system: if the simulation was continued, but all self-replicating molecules removed, the system re-invented a self-replication cycle. However, if an ancestor trace was done, in the long run, all molecules descended from DNA. This somewhat mirrors the central dogma, in that all in the long term, catalytic activity is descended from information in the DNA.

Vesicle system

The vesicle system assumes that there were cells before there was DNA (or, in other words: given that there are cell-like entities, what might we see then?). Let us see what happens there. A parasite (RNA that is only template) is put in at the beginning like before, but in the vesicle system, it is immediately filtered out by the **stochastic error corrector**. There is an evolutionary transient to

a transcription system (Figure 4.36). First, once equilibrium was achieved, the mutation from RNA to DNA polymerase was enabled. DNA polymerase (made of RNA) with a high recognition of DNA quickly invaded the system. However, it did not immediately evolve to a replicase, but maintained some reverse transcriptase activity (Figure 4.36B). After this invasion, the RNA polymerase (Rp) evolved high transcription activity. The system then stayed this way for a long time. In this phase, the fact that DNA polymerase exists at all is dependent on continuous mutations from Rp to Dp (if mutations were disabled, it went extinct). There is still a dual-function RNA-polymerase here. Later on, DNA polymerase evolved towards less reverse transcriptase activity in some vesicles, and Rp in those vesicles concomitantly became a stronger catalyst (Figure 4.36D). Vesicles with this transcription-like system then quickly outcompete others, and there is no reverse transcriptase in the system anymore (contrast this with the spatial system, where it remains).

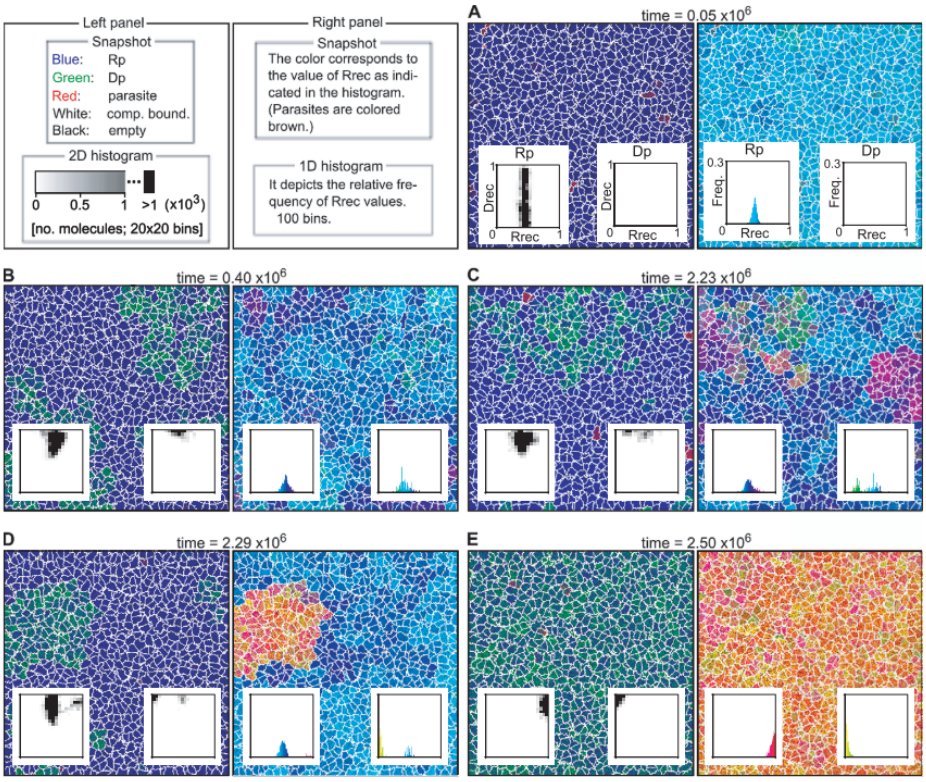


Figure 4.36: Dynamics of the vesicle model of DNA and RNA replication. Taken from: (Takeuchi et al., 2011).

Some vesicles randomly lose the transcription system, being left only with an RNA-dependent RNA polymerase: self-replication. These vesicles quickly expand and locally outcompete the DNA system, but in the long term, they disappear again (Figure 4.37). This happens multiple times in this system. Why is this?

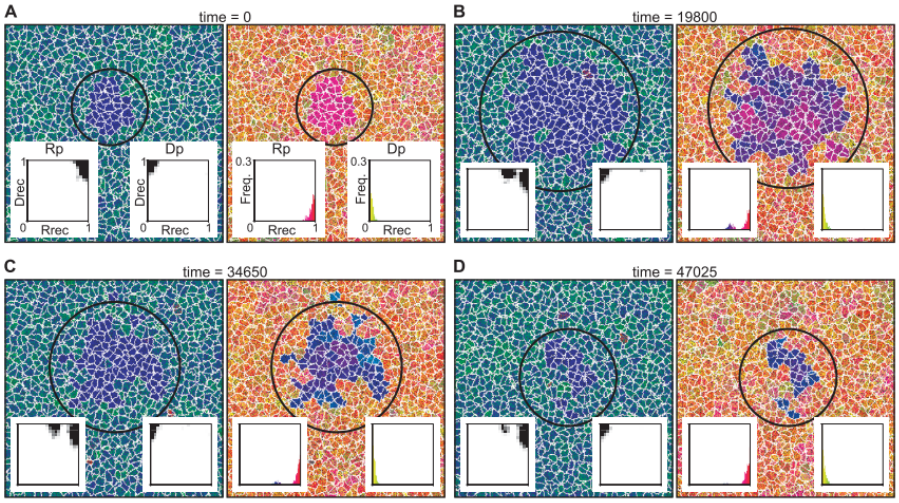


Figure 4.37: Loss of the transcription system causes short-term fitness advantage, but is competed out in the long term. Taken from: (Takeuchi et al., 2011).

If you lose DNA, recognition of RNA by RNA polymerase is decreased. Why? Well, this is the simple template-catalyst conflict: it is detrimental to replication to be a good catalyst. These vesicles thus grow slower, and are then outcompeted again by those with DNA. Thus, DNA evolves here because it is good for evolutionary reasons, not for dynamical reasons. That is to say, it only plays a role in keeping fit in the long term. DNA is not hampered by having a code that codes for strong catalysis, so this pressure that a self-replicating RNA experiences is (almost) absent. This causes molecule-level (within-vesicle) selection pressures to be less strong, so that the system with vesicles can survive. Thus, for informatic and evolutionary reasons, DNA can evolve, because a division of labour between storage and usage of information is good.

Though the caveats we discussed in the beginning are to be kept in mind (no chemical constraints taken into account, neutrality of mutation from RNA to DNA polymerase, no strandedness), this is very interesting: a modern transcription system can evolve from an RNA precursor system in vesicles. In space, the transcription variant evolves, but cannot efficiently rid itself of the self-replication variant.

Conclusion

This chapter has been quite a ride, and it will not be easy to recap everything that we have discussed in a few sentences. Let us look back on what brought us here. We started this chapter with the question of how life began, and then focussed on a workable example: the RNA world. We discovered that, at least in classical ODE models, there is a major hurdle in obtaining enough complexity: the **information threshold**. Best described by Eigen's paradox: it takes more information (a longer genome) to get a repair mechanism, but to get more information, you need a repair mechanism. That's a catch-22 for life, it would seem. Nevertheless, we exist, together with many more complex organisms. What gives? We looked at Eigen's initial (peculiar) solution: the hypercycle. An ecosystem-based solution where multiple replicators together can code more than an individual replicator. We saw that this presents a new problem: better hypercycles can never invade, because the amounts of established replicators are the largest driving force in the equations (remember the X^2 term).

We then switched gears and added space. Even though the hypercycle model is a somewhat nonsensical model that doesn't address the problem it set out to solve, adding space taught us many valuable lessons. We saw our first emergent higher level of selection: waves. These **mesoscale patterns** impart their own selection pressures, that can conflict with lower-level selection pressures. An example was the death rate of replicators: higher death rates of replicators can increase the fitness of waves, as empty space is needed for their successful propagation. Additionally, adding space made the system resilient to parasites. We also saw that evolution went to the **edge of chaos** or **border of order**: replicator death rate got as

high as possible without letting the waves go extinct. This is a general pattern of evolutionary dynamics. This was further explored in the minimal replicator-parasite system (**RP system**).

We learned that separation of time scales in models is often a very bad idea, especially if done from a position of 'I think we can separate this and it won't really matter to the dynamics'. The first example was the host-parasitoid system, where you saw that **fitness is a time-dependent function**. In time, all parasitoids were descended from those in spiral cores, but at shorter time scales, fitness for other parameter values seemed to dominate. Experimentally, you could never measure this many generations, which should caution all of us to be very wary of measurements of fitness. In the minimal eco-evo model, we saw that for the lowest mutation rate, the highest discrepancies between a strictly ecological (analogous to adaptive dynamics) and ecological + mutations simulation occurred. Therefore, the argument of 'Mutations are rare so they do not influence the ecological dynamics' does not hold at all in this case.

Next, we investigated vesicle-first scenarios of evolution, where replicators reside inside protocells that need replicators to grow and divide. We looked at Wilson's archetypal group selection model (an **existence proof**), that showed kin selection was not per se necessary for group selection to work. We then looked at **stochastic error correction**: how random sorting of replicators to daughter vesicles upon division can stochastically result in fit vesicles (where replicators are in the correct ratio) surviving, thus balancing within-vesicle fitness imbalance between the replicators.

We then looked at an explicit comparison of pre-defined (vesicles) and emergent (waves) higher levels in a RP system. We saw some interesting evolution close to the Hopf bifurcation, where at high mutation rates, the system concerns itself with not dying so fast once it invariably mutates over the **information threshold**. This model further taught us that waves are often more stable than vesicles.

Moving on to endless evolutionary dynamics in a stable population, we saw that continuous cell-level bottlenecks are what allowed the population to survive at high division volumes of vesicles. If they were removed, the system could not survive at as high division volumes. More generally, this shows that creative solutions emerge when selection pressures from different levels are about equal in strength. This might be a major mechanism allowing complexity to arise.

Finally, we looked at adding in some aspects of RNA to replicators. Specifically, we observed **symmetry breaking** when we added in plus and minus strands in both spatial and vesicle systems, which is reminiscent of the division of labour between DNA and RNA + proteins in present day cells. A final model showed that DNA can evolve for informatic (long-term) reasons only, though in the spatial system, a transcription-like system still coexisted with self-replicating molecules.

Phew, that is a lot to take in. Be sure to go over some of the sections that are still a bit fuzzy to you, or ask questions to the student assistants. Let's end on a more general note: in multi-level biological systems you should expect the levels not to be independent, but very much dependent on each other. And we should expect that micro- and macro-levels, as well as different time scales, affect each other. Then, asking the question "what level is causing the observed dynamics?" (e.g. is it the selfish gene, or the good of the species, or of your kin) is actually non-informative, because what happens at a certain level will always influence and be influenced by the other levels in the system. *What happens at the micro-level only makes sense in light of selection pressures from higher levels, and vice versa.*

Constructive evolution

We know that evolution can construct new things. New shapes and new solutions to problems. Somehow, life went from simple self-replicating entities to cells, multicellular organisms, and even the majestic elephant. Up until now, we have focused on the role of space and the effects of interlocking ecological and evolutionary dynamics (i.e. mutations happen while ecological processes play out, and both influence each other). We have thus allowed, and seen the importance of, **local interactions**. We have also seen **mesoscale patterns**. The ongoing mutations were contrasted with the classical approach of changing parameters. We looked at invasion dynamics of different phenotypes and asked what parameters (such as catalysis or death rate) were optimal in the system. We investigated endless dynamics of the **stochastic corrector**, and **symmetry breaking** in replicators with plus and minus strands, as well as the evolution

of DNA as an information carrier in a RNA world. Though these latter two systems went some way towards including more real aspects of RNA, there is a lot of ground left to cover. Now, we take a look at how novelty arises in evolutionary systems. How can evolution be constructive? For that, we need to change our models.

What we have missed until now

We have looked at RNA-like replicators in space. These replicators could change through **phenotypic mutations**: mutations that directly acted on a parameter such as death rate. However, that is not how mutations on a true RNA molecule work. Functional RNAs have a primary sequence, their genotype, that folds into a structure. Although computing the precise tertiary structure is still challenging (Major and Thibault, 2008), methods on predicting the secondary structure are very good (Figure 4.38) (Tsang and Wiese, 2010). This secondary structure is formed by interactions between the different bases of the RNA molecule. The secondary structure then folds into a 3D tertiary structure. This final 3D tertiary structure is the phenotype: the functionality of the completely folded RNA is what natural selection can act upon.

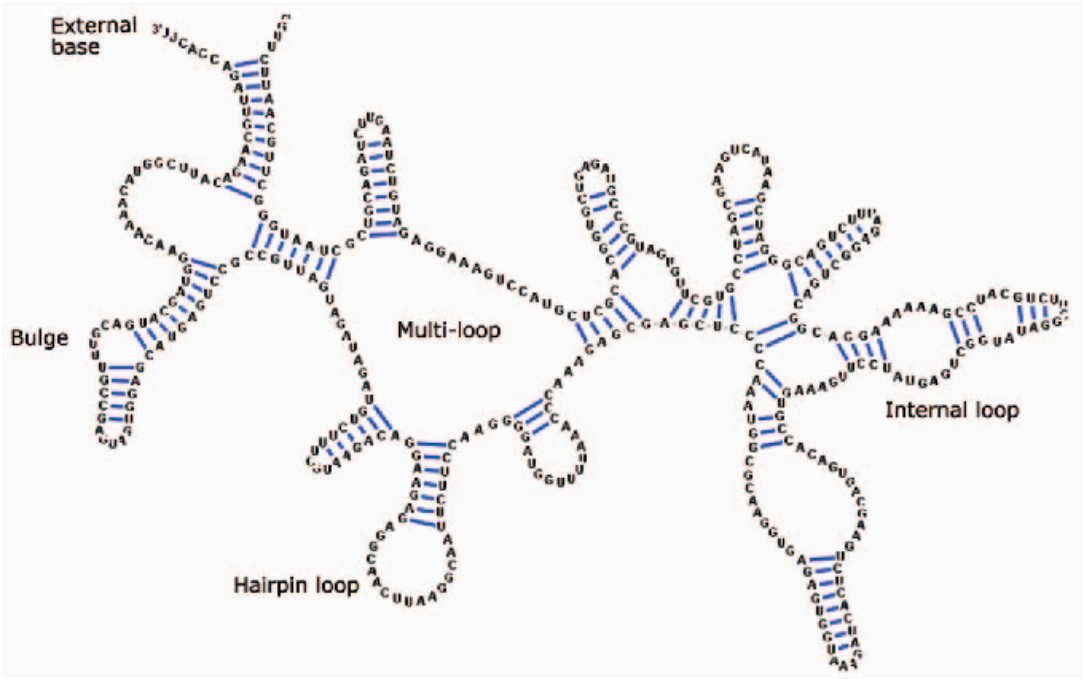


Figure 4.38: Structural complexity in the RNA secondary structure. Taken from: (Tsang and Wiese, 2010).

Our models up to now were simplified by omitting these levels between the mutations and the phenotype: the RNA sequence is mutated, which might change its structure. Given that it is reasonable to say that it is the structure that has a certain functionality, we should say it is this structure that affects fitness. We will now introduce this mapping from the genotype to the phenotype. As mentioned before, accurately predicting the tertiary structure is unresolved, although new Deep Learning methods are making large strides, as Alphafold 2 did for proteins (Pearce et al., 2022). When this work was done the situation was very different, and the secondary structure can be a good proxy for the real tertiary stucture. Hence, we will take the secondary structure as a proxy for the phenotype. Most folding algorithms use free energy minimisation (where the binding energies of bases are calculated and (globally) minimised) (Mathews and Turner, 2006), although here, too, Deep Learning methods are changing the game (Fu et al., 2022).

Using these two levels, the genotype and phenotype level, we introduce a **complex many-to-one genotype-to-phenotype mapping (GPM)**: genotypes fold into a certain secondary structure, which is what defines a fitness that can be selected for. Many genotypes fold into the same secondary structure. We shall see how this alters evolutionary dynamics.

Besides this mapping, we also introduce a **finite population**. In the replicator equation by Eigen, all possible genotypes existed. In real biological systems, possible genotypes far outnumber extant genotypes: a DNA strand consisting of 100 bases has 4^{100} possible sequences. The human genome contains approximately 3,000,000,000 bases. These have $4^{3,000,000,000}$ possible combinations. That number is astronomically high, and by far larger than the amount of humans that have existed. Thus, real evolution somehow finds good fitness while it cannot possibly sample all genotypes, or even a large fraction of them. We will see how this is possible.

We thus use the RNA **genotype-phenotype mapping** as a **paradigm system**: in order to understand how genotypes translate into phenotypes, we will study one particular system and draw conclusions from it. We will later discuss whether these conclusions are special (only true for RNA), or generic features of biology.

A first peek: genetic algorithms

The first to model and use the constructive power of evolution was not a biologist but a computer scientist named John Holland (Holland, 1992). He was the founding father of evolutionary algorithms. The idea was simple: evolution is clearly a powerful tool to construct solutions to problems. What if you replicate the workings of evolution to find the most efficient algorithm for a certain computational problem? If you want an algorithm that can efficiently sort lists of numbers from high to low, you could start with a random population of algorithms. These algorithms have a 'genome' that represents the functions they call on the input to produce their output. Unlike in real-life evolution, you know what the perfect algorithm should do beforehand: it should correctly sort every list of numbers you give it. So there is an **external fitness criterion**: it is defined a priori what the 'most fit' algorithm is. Now take a set of problems: in this case, sequences of numbers. You know the correct sorting of these numbers. Now unleash every algorithm in the population on the problems, and record how many they solve correctly. The more problems an algorithm solves correctly, the fitter it is. All that remains are three ingredients: algorithms should replicate according to their fitness, there should be a death rate, and, importantly: there should be mutations in the genome of the algorithms that change them. Voilà, you have created an *in silico* version of the great optimisation process we know as evolution.

In this way, algorithms that, by chance, solve some problems correctly will rise to dominance. Better algorithms arise by mutations and the **external fitness criterion** allows these to replicate more than others. Very importantly, Holland first recognised the importance of different **mutational operators**: besides point mutations, there are rearrangements, cross-over, deletions and duplications. These all affect the evolutionary process.

Evolutionary algorithms are used to find efficient solutions to complex problems, such as in (computer) network architecture, robotic control, the optimal placement of windmills for energy generation, and *in vitro* evolution of ribozymes (Walters and Smith, 1995; Biebricher and Gardiner, 1997; Fleming and Purshouse, 2002; Grady et al., 2005). These evolutionary algorithms align well with our change in perspective: we focused on parameter changes in spatial systems, and asked: *"What evolves?"*. Now, we too will define **external fitness criteria** and ask: *"How does evolution solve this problem?"*. For this, we will use the paradigm RNA system.

Adaptive landscapes

In previous sections, we have discussed spatial models of an RNA world. As has been noted, we left out a quintessential property of RNA: that the sequence is connected to a structure. In experimental studies, it has been shown that evolving a functional RNA ligase happens quickly. With small population sizes (in the range of 10^{10} - 10^{16}), efficient RNA ligases can often be evolved in a mere ten generations with random mutations and selection *in vitro* (Ekland et al., 1995; Ekland and Bartel, 1995; Jaeger et al., 1999; Joyce, 2007). This is astounding: if we take a length of 220 base pairs, there are 4^{220} options (about $2.8 * 10^{132}$!). How can this happen so quickly, when so few genotypes are sampled?

To think about this, we should think about the difficulties that evolution faces in finding high fitness. A tool that is considered useful here is the notion of **adaptive landscapes**: visualisations of the fitness for every genotype (given a certain environment). In such a space, you can see what the optimal genotypes are, and what difficulties might arise in reaching them via natural selection. Please note that adaptive

landscapes are always relative to a certain environment (i.o.w. fitness is dependent on the environment) which is not represented in the figure, but plays a role nonetheless. Conceptually, we could envision two different types of landscape. A smooth adaptive landscape has no **epistasis**: mutations all have independent effects on fitness. A smooth landscape with a single peak would be easiest to traverse: there is one defined optimal fitness, and every mutation independently brings you closer to it or farther away from it (Figure 4.39A). Then we could think of a rugged landscape: a landscape where mutations in different combinations produce different effects. If there are multiple slightly lower fitness peaks than the **global optimum** (the highest fitness peak in the field), we might think that evolution could easily get stuck there, because every mutation away from a **local fitness optimum** decreases fitness (but these mutations would be necessary to get the **global optimum**) (Figure 4.39B). This is especially the case given that we now think of finite populations: there are so many possible genotypes that they will never all exist. If existing genotypes by chance all surround a **local fitness optimum**, how can evolution ever find the **global optimum**?

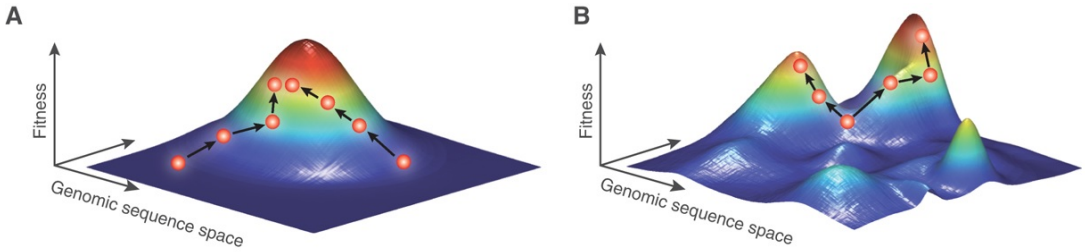


Figure 4.39: Two different adaptive landscapes we might envision. A: there is one, easily achieved global fitness peak. B: there are multiple fitness peaks. In theory, this might make it more difficult for evolution to find the best possible fitness in this landscape. Adapted from: (Conrad et al., 2011).

The adaptive landscape of RNA

Now let us zoom in on the RNA **adaptive landscape**, and see what the **genotype-phenotype mapping** means for those landscapes. Here, the genotype is the sequence of nucleotides (As, Us, Cs, and Gs), and the phenotype is the calculated secondary structure. Given this genotype and phenotype, what does the RNA **adaptive landscape** look like, and why can efficient RNA ligases so quickly evolve?

Neutrality in RNA landscapes

A core concept that makes evolution in RNA adaptive landscapes so efficient is neutrality. First defined by Kimura (Kimura, 1983). When DNA bases change but there is no concomitant change in an amino acid, this is neutral evolution. Similarly, in the RNA world, if the sequence changes but the structure is unchanged, this is neutral evolution.

Figure 4.40 illustrates the high **neutrality** of RNA landscapes (Huynen et al., 1993). Given that this image is 3D it might be hard to understand at first. Start with the x- and z-plane. These show the effects on the (calculated) secondary RNA structure that mutations can have. If you look at the "front" of the picture (a single mutation), you see that the vast majority of mutations have no effect on the secondary structure: they are neutral. However, there are also *single mutations* that change the structure by up to 100%!

This model finding is supported by experimental evidence: eukaryotic mRNA folding can be completely changed by just one mutation (Konings et al., 1987). Additionally, an RNA ligase exists where every position in the sequence is essential for the function (Schultes et al., 2000). Changing a certain residue changes every fold in the molecule and turns the molecule into *another functional ligase*.

Note here that most models do not take this effect of mutations into account. Mutations are often modeled as gradual departures from a starting value. For example, individuals can be modeled with a parameter that describes a certain propensity for feeding on a food source, and offspring can deviate by between 0-2% in that parameter. However, we see even in this simple model that this is a misrepresentation of

the evolutionary process: the redundant **genotype-phenotype mapping** means that a single mutation can have no effect at all, or have a drastic effect on secondary structure.

If we now look at the axis with the number of point mutations, we see that after about 20 random mutations from a given sequence, the distribution of the effects on the change in secondary structure stabilises: additional mutations are neither more nor less likely to affect the secondary structure. This defines the **correlation length**: how many mutations you can step away from a sequence of origin until the resulting structure is no longer dependent on the sequence of origin, but could just as well have come from a different sequence. This **correlation length** is at 20 mutations. There is thus a saturation in the magnitude of change of secondary structure.

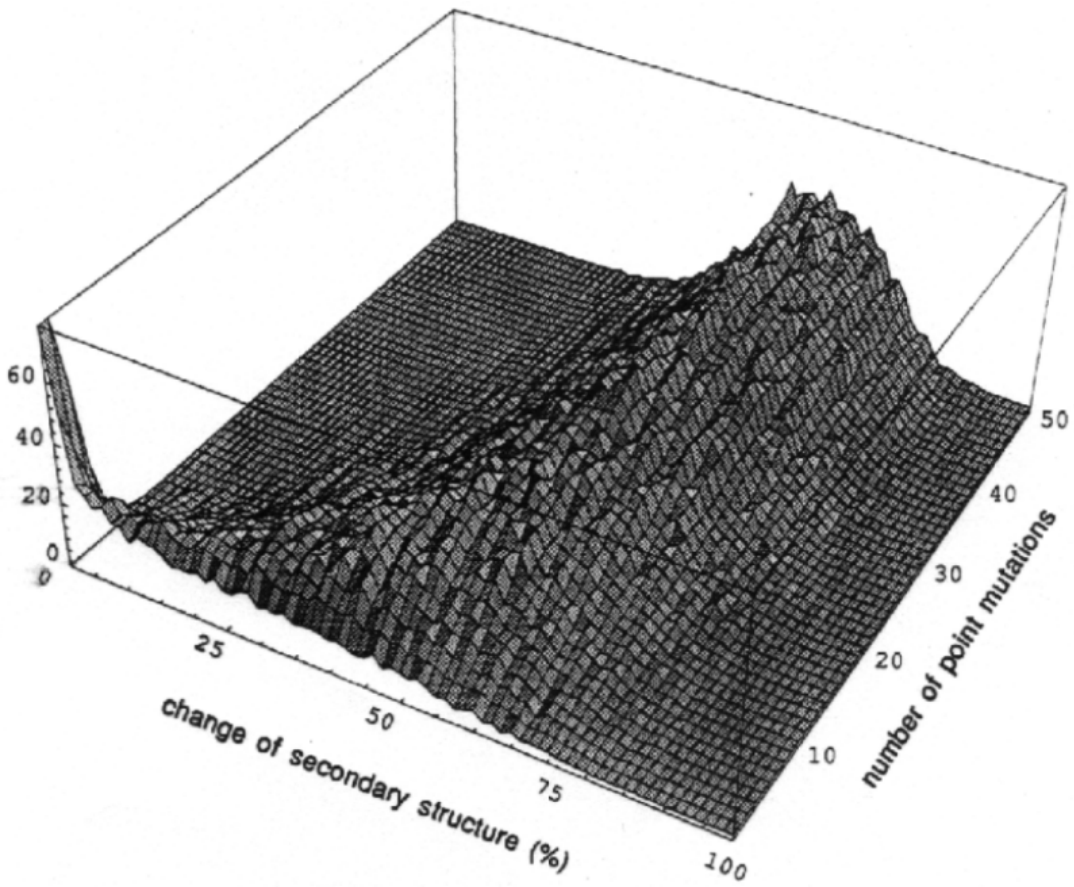


Figure 4.40: The high neutrality of RNA landscapes. On the x-axis, the percentage of change in the calculated secondary structure. On the y-axis: the number of point mutations added to a sequence. On the z-axis: the amount of times this occurred. Adapted from: (Huynen et al., 1993).

Looking back on the aforementioned smooth and rugged adaptive landscapes, where does RNA fit? Strikingly, it not fully smooth nor is it fully rugged: it is both:

1. One mutation can completely change the phenotype (so a single change can throw you right off from the fitness optimum). Contrast that with changes gradually moving you from the fitness optimum in a smooth landscape.

2. Many mutations are neutral (there is no single clear fitness peak on the sequence level).

The intuition was that rugged landscapes are difficult for evolution to traverse, yet experimental evolution of ligases was extremely efficient. How can this be? To understand that, we have to dive further into **neutrality**.

Percolation on neutral paths

We now focus on the work of Schuster (Schuster et al., 1994; Schuster, 1995). Figure 4.40 shows that the **one-mutational neighbourhood** (the pool of sequences that is one point mutation away from a sequence) is overwhelmingly neutral. You see that as you introduce more point mutations into a sequence, the chance of no change in secondary structure happening decreases rapidly. But another question arises: can you move from neutral mutation to neutral mutation until you completely changed the sequence, and yet retain the same structure? In other words, do neutral mutations percolate through sequence space?

In short, the answer is yes. Schuster et al. simulated RNA sequences and their structures. They started with a pool of sequences (of length 100) that folded into 500 different reference structures. They then allowed only point mutations in which the secondary structure was retained. The question: what is the distribution of the neutral path lengths; how far can you go on completely neutral mutational paths?. The vast majority of sequences could percolate fully through sequence space (Figure 4.41: the peak of the distribution is at n = 100): even if the sequence was completely different from the sequence of origin, the structure was still the same. The peak in the left of the figure is the answer to a different question: how many steps can you mutate towards a *specific* target sequence while retaining your original structure? To test this, a reference sequence and its structure were chosen, and a random target sequence was chosen. Sequences were then allowed to mutate only such that they kept their structure but approached the target sequence in sequence space. The results show that you can, on average, get to about 20 mutational steps away. The most important finding, however, is clear: sequences can percolate over neutral networks, and even completely different sequences can have the same structure. The genotype space is easily traversable and highly interconnected due to these networks.

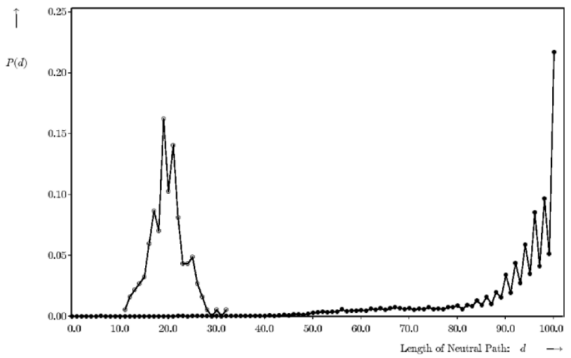


Fig. 4: Neutral paths. A neutral path is defined by a series of nearest neighbour sequences that fold into identical structures.

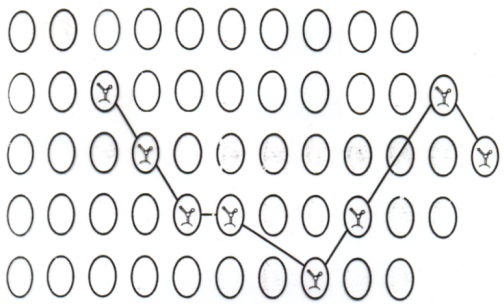


Figure 10. Percolation of sequence space by *neutral networks*. A neutral path connects sequences of Hamming distance h = 1 (singla base change) or h = 2 (base pair change) that fold into identical minimum free energy structures. The sketch shows a neutral path of length h = 9. The path ends because no identical structure was found with h = 10 or h = 11 from the reference.

Figure 4.41: Neutral paths percolate through sequence space. Left: if a pool of sequences with length 100 are allowed to mutate via SNPs that do not change their structure, how far can they mutate away? The peak of this distribution is at neutral path length $(d) = 100$. The sequence can thus change completely yet retain the same structure. The peak on the left shows to what mutational distance from a defined target sequence sequences can mutate without changing their structure. Right: illustration of neutral paths. The sequence space is percolated with neutral networks or neutral paths, interconnected networks of sequence mutations that are neutral with respect to the structure and span the whole sequence space. Adapted from: (Schuster et al., 1994; Schuster, 1995).

Novelty along the percolating path

The question arises what this means for the ease of evolving novel structures. The fact that neutral mutations connect vast reaches of sequence space will affect the evolutionary process. To investigate, Huynen calculated how many structures are seen in the one-mutational neighbourhood as a sequence meanders along a neutral path (Huynen, 1996). The amount of novel structures (phenotypes) seen along the neutral path continuously increases (Figure 4.42). Despite this novelty along the neutral path, there is a **shadow of similar structures**: structures which are continuously in the **one-mutational neighbourhood** of your sequence (i.e. now we count how many structures you see that you have already

encountered before along the path). Though the amount tapers off with increasing neutral path length, there are structures (which are closely related to the original structure) that are continuously seen along the neutral path.

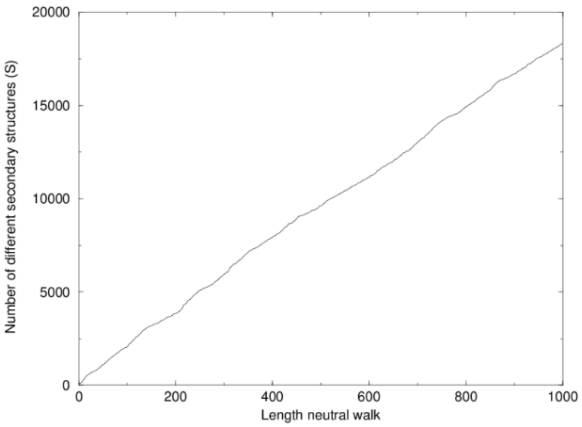


Figure 1: Perpetual Innovation along the Neutral Net.

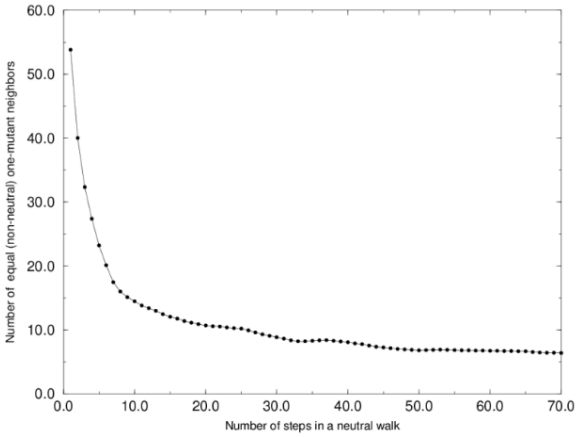


Figure 2: Conservation along the neutral net.

Figure 4.42: Continuous innovation and a **shadow of similar structures** along neutral paths. Left: if sequences are allowed to keep walking along neutral mutational paths, the amount of novel structures seen keeps rising linearly. Right: as the walk along the neutral path continues, the amount of structures that have already been seen in the **one-mutational neighbourhood** decreases, but never becomes 0. Adapted from: (Huynen, 1996).

Clustering on the neutral network

How do sequences percolate through the neutral network? If you start with a specific sequence, does the sequence randomly drift away from this point? Is there some kind of pattern to be found in how neutral mutations wander away over sequence space? This question was first explored in a flat landscape (a landscape where every sequence folds into the same structure). In that case, the neutral mutations are equal to a diffusion process **??**. Interestingly, it was shown that there is some sort of speciation in a flat (neutral) landscape (Higgs and Derrida, 1992). For this, the authors used a population of 1000 individuals. Population size was fixed, and each individual had an equal chance of reproducing asexually. Over the generations, the last common ancestor of sequences was followed. In this completely neutral landscape, there are nonetheless clusters of sequences with different most recent common ancestors (Figure 4.43). These can be seen as clusters moving along the time (x) axis, where there is larger chance that certain sequences share last common ancestors. Sequences thus percolate in random directions over neutral networks, but because population sizes are finite and some sequences reproduce more than others due to chance, defined subclusters form on the neutral network.

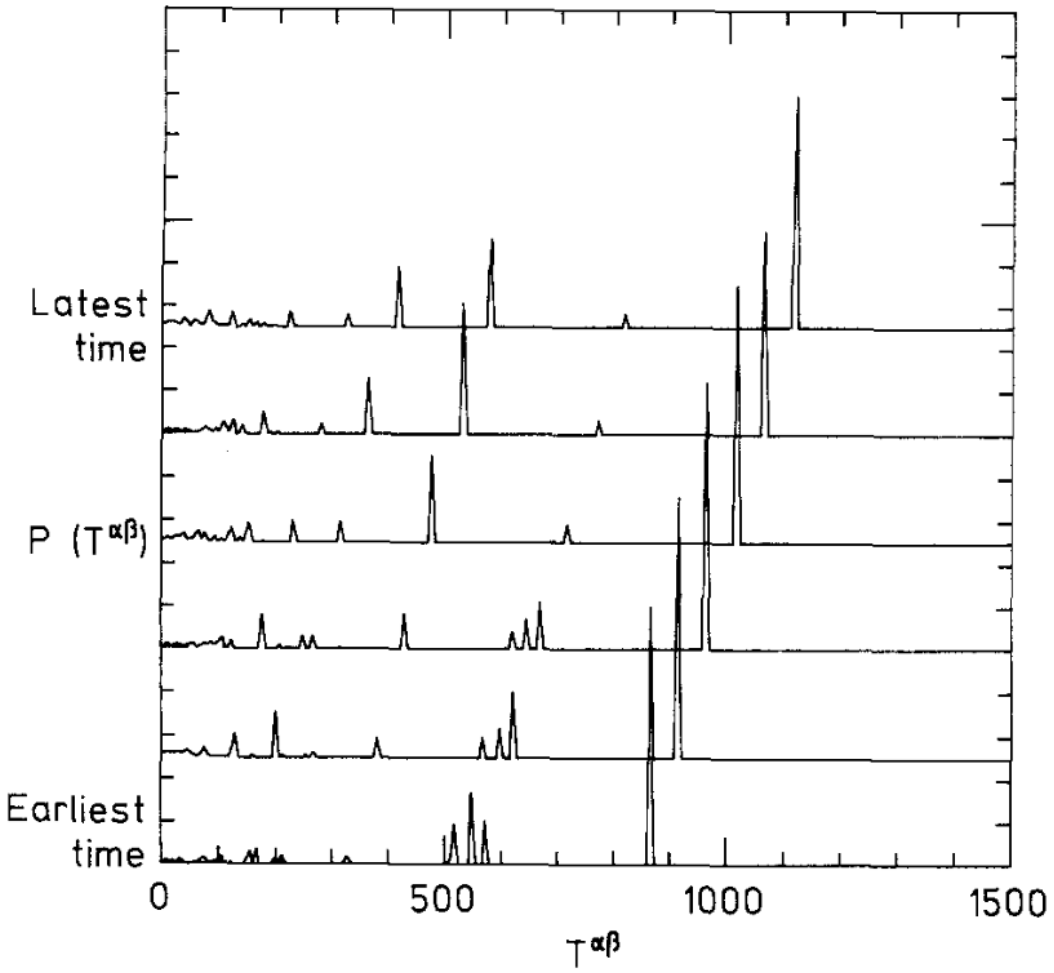


Figure 4.43: Clusters of defined neutral subpopulations (size shown on the y axis) move along the time axis (x) in steps of 50 generations. Clusters arise due to random effects and fixed population sizes, even though they are all neutral with respect to one another. Peaks can appear and disappear over time. Taken from: (Higgs and Derrida, 1992).

This analytical result is backed up by a computational result (Huynen et al., 1996). In this study, 1000 copies of the sequence coding for the phenylalanine tRNA were initialised. Population size was fixed. Mutation chance per base pair was 2% and the target structure was set to the phenylalanine tRNA structure. Thus, all sequences started with the correct structure, and those sequences which deviated in structure suffered fitness penalties. By taking the **Hamming distance** between all sequences and using **dimensionality reduction** to project sequence clusters into 2D, one can see that there are defined clusters of sequences in different parts of the genotype space (Figure 4.44). The Hamming distance is a simple metric: if you have the aligned sequences AABB and ABBA, the Hamming distance between them is 2, because they differ in two positions. It's a distance that adds 1 whenever two positions are not exactly the same.

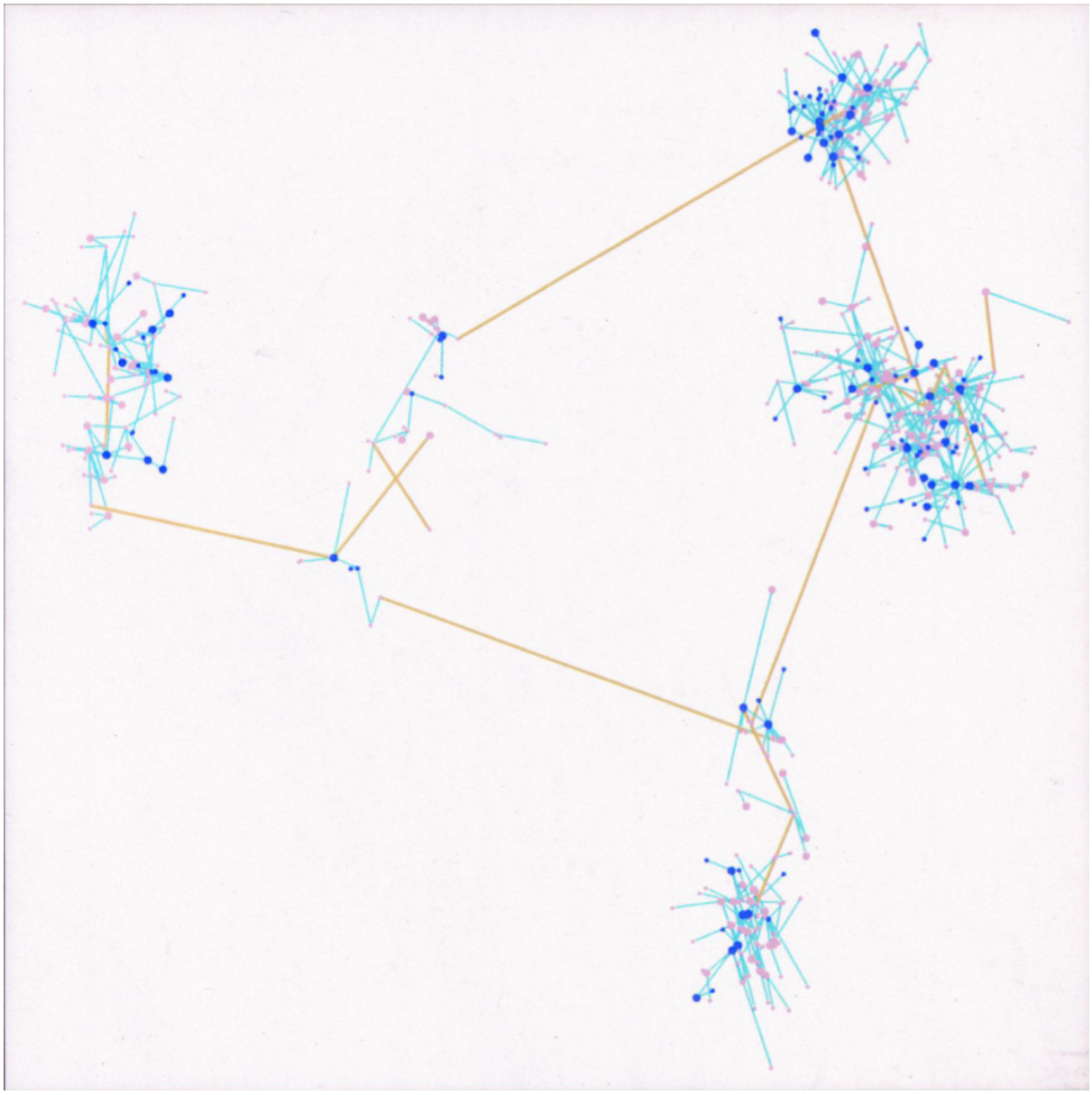


Figure 4.44: This is a **PCA** (a form of **dimensionality reduction**) done on a population of one thousand neutrally evolving sequences coding for the phenylalanine tRNA. Sequences are connected by lines to their closest **mutational neighbours** (blue: **Hamming distance** < 6, yellow: **Hamming distance** $>= 6$). Due to random effects, neutral clusters of sequences exist that can sample defined parts of the genotype space. Adapted from: (Huynen et al., 1996).

Consequences for evolution and the adaptive landscape

We introduced **adaptive landscapes** as a useful tool to think about what evolution encounters during the rise to optimal fitness. We thought about smooth and rugged fitness landscapes: whereas smooth landscapes are easy to traverse with their **single fitness optimum** and/or lack of **epistasis**, the **local optima** and/or **epistasis** of a rugged landscape make it hard to traverse. That, at least, was the intuition. We have found out that RNA landscapes are, in fact, both rugged and smooth.

If we map the immense **neutrality** in the sequence space back to **adaptive landscapes**, a different picture emerges: **local optima** are a bug in our thinking, rather than an actual phenomenon. In the true RNA adaptive landscape, **local optima** do not exist (or only extremely infrequently): structures with

intermediate fitness are retained while the underlying sequences change continuously. The sequences percolate along **neutral paths**, continuously seeing many new structures in their **one-mutational neighbourhood**. Not only that: these neutral sequences are also clustered into neutral subpopulations or **quasispecies**: groups of sequences that are close to each other in the genotype space. These groups are sampling different regions of the genotype space, and therefore potentially different structures. In other words, populations can do a large "parallel search" for the optimal structure, which is more efficient. Structures with better fitness can thus rapidly be searched *while the current best structure is retained*.

How does this translate to observed evolutionary phenomena? In 1972, Stephen Jay Gould revolutionised evolutionary theory with his theory of **punctuated equilibrium** (Eldredge and Gould, 1972). Darwin's original contention was that mutations with small fitness benefits accumulate over time due to natural selection, and that this culminates in the gradual formation of new species. He was confounded by the lack of gradual change in the fossil record: 'Why then is not every geological formation and every stratum full of such intermediate links? Geology assuredly does not reveal any such finely graduated organic chain; and this, perhaps, is the gravest objection which can be urged against my theory' (Darwin, 1859). The new synthesis (state of the art in evolutionary theory around the 1950s-1970s) still supported this gradualism, and breaks in the fossil record were held to be due to imperfect preservation.

Gould argued something different, and something that we can now understand: there is no gradual change. There are long periods of homeostasis (equilibrium), punctuated by bursts of innovation. His proposed mechanism was that of allopatric speciation: a species lives in a certain area, to which it is well-adapted. A breakaway population that becomes geographically separated and lives in a different area will, by chance, only carry a subset of the genetic traits of the original population, and will rapidly adapt to the new area. Thus, a new species can rapidly come to be (Eldredge and Gould, 1972).

What we have learned of neutrality supports the observation of punctuated equilibria, but not (per se) Gould's supposed mechanism. Instead, we have found that there are prolonged periods of **phenotypic neutrality**, while the underlying sequences percolate along **neutral paths**. This allows a population of sequences to sample many new structures, until such time as one is found that is better. The phenotype appears suddenly, but there is continuous change in the underlying sequences. This is illustrated beautifully in work by Huynen *et al.* where they modeled evolution in a flow reactor with capacity N = 1000, initialised with 1000 copies of a single random RNA sequence of length 76 (Figure 4.45) (Huynen et al., 1996). They defined a target structure (the phenylalanine tRNA), allowed for mutations and set a replication rate per sequence. This rate was dependent on the distance of its structure to the target structure, with lower distances yielding higher replication rates. The line shows the average distance of structures in the population to the target structure. Superimposed on the picture are dots. These indicate the **Hamming distances** between all sequences in the population that are present > 10 times, projected on to one dimension. If that sentence is unclear to you: fear not. It simply shows the *sequence-level* diversity in the population of sequences. These dots illustrate again that there are different neutral subclusters (collections of dots which are closer together). What you see are **punctuated equilibria** in optima forma: on the population level, nothing happens to the (average) phenotype for long stretches of evolutionary time. However, different clusters of **phenotypically neutral** sequences arise and percolate over the network, causing a gradual increase in sequence diversity. These sequences consistently 'see' new structures in their **one-mutational neighbourhood**. Then, a structure is encountered that is closer to the target structure, and the sequence coding for it quickly comes to dominate the population. At these points, the population diversity on the sequence level is notably reduced (the dots are much closer together). Then, **neutral percolation** begins again. This repeats until the target sequence is reached.

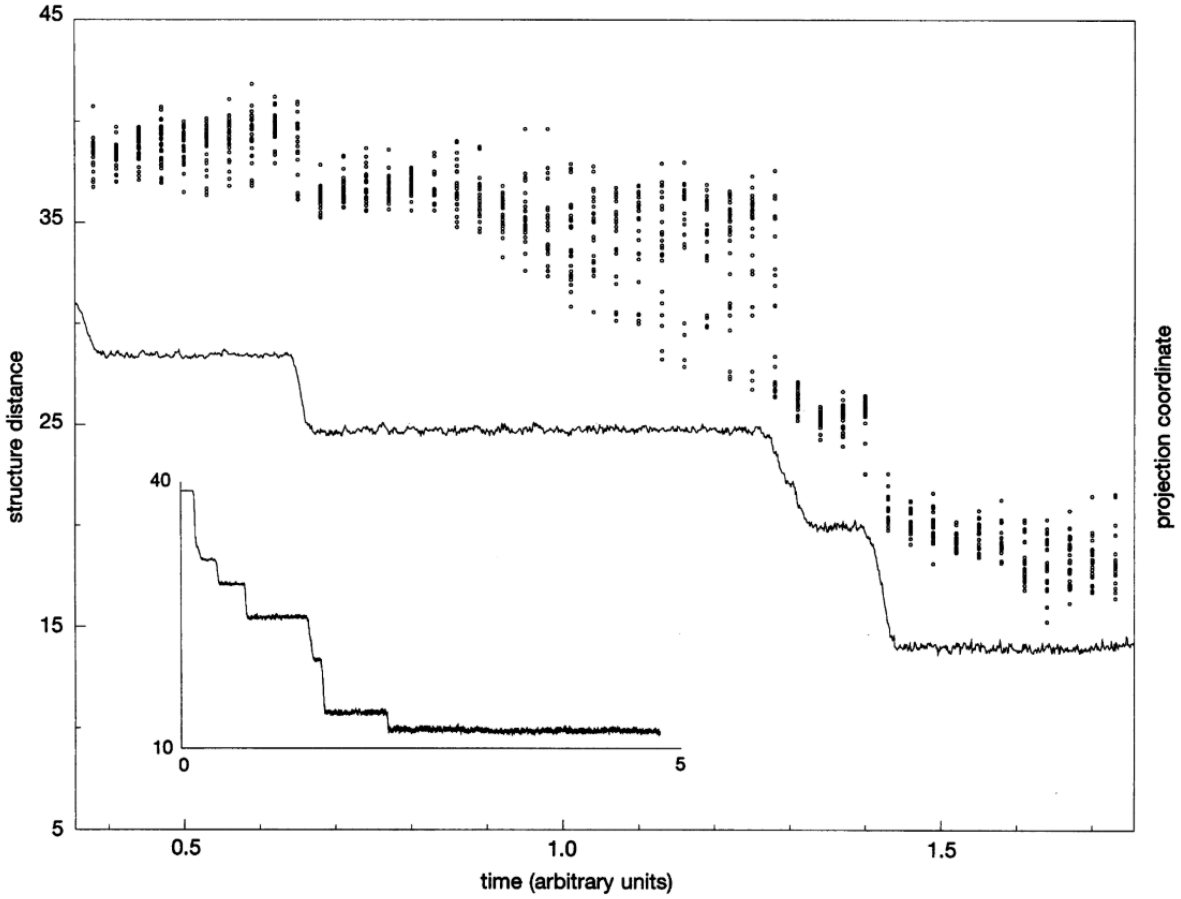


Figure 4.45: Evolutionary dynamics of 1000 DNA sequences of length 76 evolving towards a target secondary structure (phenylalanine tRNA cloverleaf). Dots show **dimensionality reduction** of the neutral sequence-level variety. Whenever a structure is found that is closer to the target, that sequence quickly takes over the population and **neutrality** is lost, only to be gained again during the search for the next favourable adaptation. Dynamics clearly show **punctuated equilibrium**. Taken from: (Huynen et al., 1996).

Similar behaviour has been found in experimental studies on evolution, such as in Lenski's long-term evolutionary experiment (LTEE) (Lenski et al., 1991; Lenski and Travisano, 1994; Barrick et al., 2009). This is an experiment that has been going on since 1988. *E. coli* are grown in flasks, fed at set times, and new generations started by sampling from the current generation daily. Samples are regularly frozen, so one can look back in evolutionary time, and strains are subjected to additional evolutionary experiments such as adaptation to novel media or nutrients. In the experiment, it was observed that cell size, which is correlated with bacterial fitness, evolved in a step-wise manner: long periods of phenotypic stasis, followed by a sudden jump in cell size (Figure 4.46) (Elena et al., 1996). A live report of **punctuated equilibrium**, if you will. Whether this is related to neutral networks, or a feature of rare mutations, is not yet clear.

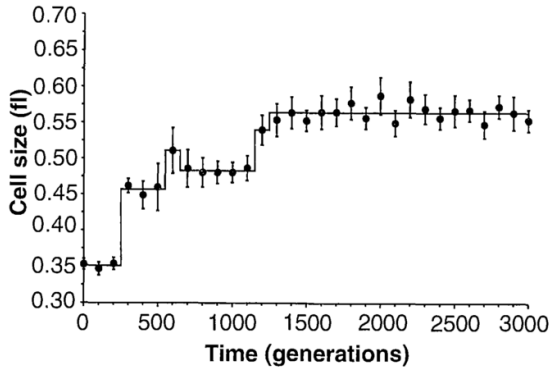


Fig. 1. Change in average cell size (1 fl = 10^{-15} L) in a population of *E. coli* during 3000 generations of experimental evolution. Each point is the mean of 10 replicate assays (*22*). Error bars indicate 95% confidence intervals. The solid line shows the best fit of a step-function model to these data (Table 1).

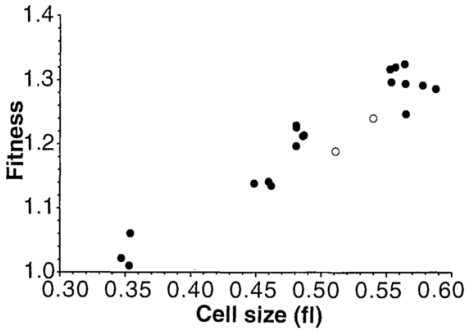


Fig. 2. Correlation between average cell size and mean fitness, each measured at 100-generation intervals for 2000 generations. Fitness is expressed relative to the ancestral genotype and was obtained from competition experiments between derived and ancestral cells (*6, 7*). The open symbols indicate the only two samples assigned to different steps by the cell size and fitness data.

Figure 4.46: Punctuated equilibrium in a population of *E. coli* experiencing rare mutations that alter cell size. Left: change in average cell size during 2000 generations of *E. coli* evolution. Each point is the mean of 10 replicate assays. Solid line is the best fit of a step-function to this data. Right: Correlation between cell size and mean fitness, measured at 100-generation intervals for 2000 generations. Fitness is expressed relative to the ancestral phenotype, and was derived from competition experiments. Adapted from: (Elena et al., 1996).

Taken together, the apparent ruggedness of RNA landscapes is nullified by the power of **neutral evolution**. Through evolution that is neutral on the phenotype level (**phenotypically neutral**), sequences can *percolate over a neutral network* and continuously 'see' novel structures. This *percolation is clustered*, with different subsets of the population sampling different regions of genotype space, further increasing efficiency. There is smoothness in apparent ruggedness. Because of this, phenotypic innovation is followed by neutral drift on the sequence level until a new phenotypic innovation is found: this is a possible explanation for **punctuated equilibria**.

Constructed adaptive landscapes

We have now looked at the adaptive landscape of RNA, with secondary structure as a proxy for the phenotype (keeping in the back of our mind that tertiary structure is a better indicator, and that other effects could play a role in reality, of course). To get more of a grip on adaptive landscapes, let's look at some constructed ones and the dynamics of evolution on them.

Constructed landscape 1: Kauffman's NK networks

We first look at Kauffman's NK networks. This work serves as a caveat about trying to be general but thereby missing the point, as has been alluded to early in the reader. Kauffman wished to study how ruggedness hinders the evolutionary process. In Kauffman's NK networks, there are a number of blocks (N) which can be either 0 or 1. In both states, each block gives a certain fitness benefit. Thus, the parameter N gives the dimensionality of the network (rather like genome positions). K is the connectedness of each block, i.e. on the value of how many other blocks a block's fitness contribution is dependent. In this system, Kauffman tried to investigate the optimal connectedness of blocks (i.e. how many genes should be connected for optimal fitness) and other effects (Kauffman, 1969, 1974; Kauffman and Levin, 1987). The system works as shown below for N = 3 and K = 0 (Figure 4.47). Beside the schematic representation of how it works, you see the fitness landscape to which it leads. This fitness landscape is known as a Mount Fuji landscape (Hayashi et al., 2006): it has continuously rising fitness and one **global fitness peak**. Without the knowledge of the power of **neutrality** that we now have,

this looks like the easiest landscape for evolution to navigate.

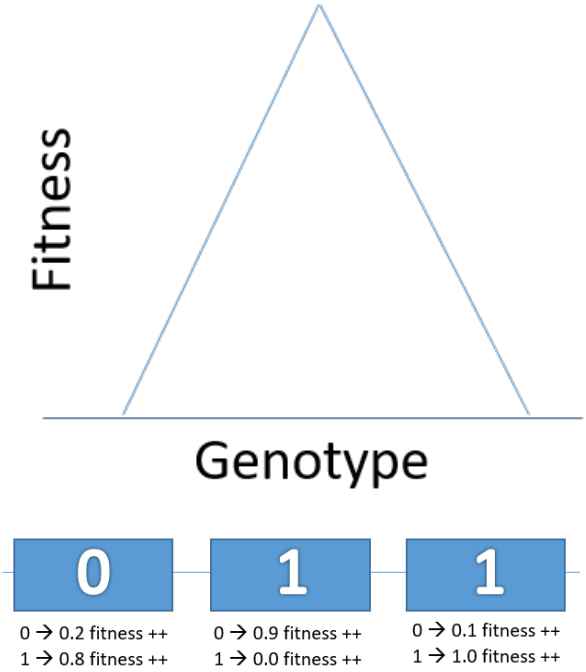


Figure 4.47: Illustration of the work by Kauffman on NK networks and the **adaptive landscape** to which it leads. Left: Blocks can be either 0 or 1, and have a specific fitness benefit in each case. The fitness a block gives can be made dependent on the state of other blocks. In this example however, there is **no** dependency (K=0). Right: This setup results in an **adaptive landscape** that is conceptually easy to traverse. This landscape is called a Mt. Fuji landscape, after the Japanese mountain with one defined peak and sloping sides (Hayashi et al., 2006). Made by Dieter Stoker, based on work by Kauffman (Kauffman, 1969, 1974; Kauffman and Levin, 1987).

If $K > 0$, then there are **epistatic interactions**. The simplification is that for every position in the 'genome', the amount of **epistatic interactions** is the same (so if $K = 1$, every position's fitness is dependent on the value of one other position) (Figure 4.48). These **epistatic interactions** give **ruggedness**, and allow study of how that influences the evolutionary process. Note that this system has no **neutrality** whatsoever: that idea came later, by studying the RNA landscapes that we have just looked at.

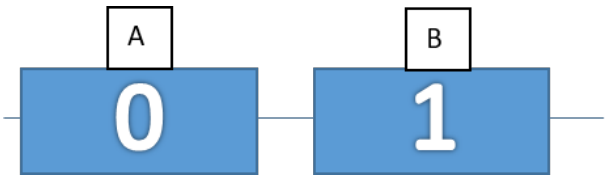


Figure 4.48: Conceptual illustration of NK networks by Kauffman with $K = 1$, such that the fitness contribution of each block is dependent on the value of one other block. Made by Dieter Stoker, based on work by Kauffman (Kauffman, 1969, 1974; Kauffman and Levin, 1987).

In this case, for two positions, we could get the following situation for the fitness contribution of A:

1. $A = 0; B = 0 \rightarrow 0.1$ fitness
2. $A = 0; B = 1 \rightarrow 0.8$ fitness
3. $A = 1; B = 0 \rightarrow 0.5$ fitness

4. $A = 1; B = 1 \rightarrow 0.9$ fitness

Thus, random patterns are assigned (randomly chosen which blocks are dependent on which other blocks). Interactions also need not be symmetric (the dependency of B on A can be different from that of A on B). By researching these landscapes, Kauffman found that a Mt. Fuji landscape is in fact not optimal for evolution. However, **neutrality** did not enter into this thinking at all. So, by studying a general example outright, he actually drew conclusions that were far away from generic. By studying the dynamics of specific landscapes (RNA evolution), we were able to draw much more general conclusions about the importance of **neutrality**.

Constructed landscape 2: Royal road landscape

A **royal road landscape** is an **adaptive landscape** where there is a rising line straight to the top possible fitness, interspersed with neutral areas. From studying the RNA **adaptive landscape**, van Nimwegen took the idea of neutrality and then applied that to genetic algorithms. He made a landscape where he added **neutrality**, a schematic representation of which is shown (Figure 4.49).

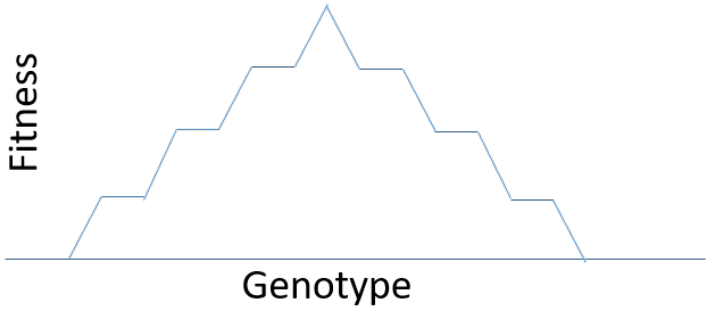


Figure 4.49: The artificial **royal road landscape** constructed and studied by van Nimwegen. Made by Dieter Stoker, based on work by van Nimwegen (Van Nimwegen, 1999).

How does this work? The genome is made up of blocks of 1s and 0s. There is a target sequence per block, for example: 011001. There are mutations, and the blocks only give a boost to fitness once their target sequence has been met. Thus, you can see how the landscape arises: periods of neutral mutation, until a block arrives at its target sequence, bumping up the fitness (Figure 4.49). Given that this idea was based on observations in the RNA landscape, does RNA have a **royal road landscape**? No, because after pro-longed evolution , the local RNA landscape becomes flatter (more neutral) which is not implemented here. Additionally, we defined here what is neutral and fitter *a priori* by having blocks of sequence, while in the RNA landscapes, you can only find that out by sampling the genotype space. Below, we see what the evolution on this constructed **royal road landscape** looks like. Here, q is the mutation rate, K is the size of blocks, and M is the population size (Figure 4.50).

We observe the following:

1. Epochal evolution/**punctuated equilibria** (Figure 4.50a-d).

2. Increased mutation leads to an increase in maximum fitness, but the system does go over the **information threshold**: it cannot keep the fittest. It does, however, keep the part of the string that can be maintained given the **information threshold** (Figure 4.50g and h, for instance).

3. Lower population sizes lead to more stochasticity and an earlier **information threshold**.

What we see here attenuates our idea of the **information threshold**. Rather than completely losing all fitness when it is crossed, we instead note that if parts of the sequence give some contribution towards total fitness, some can be maintained when you are over the threshold for the entire sequence length.

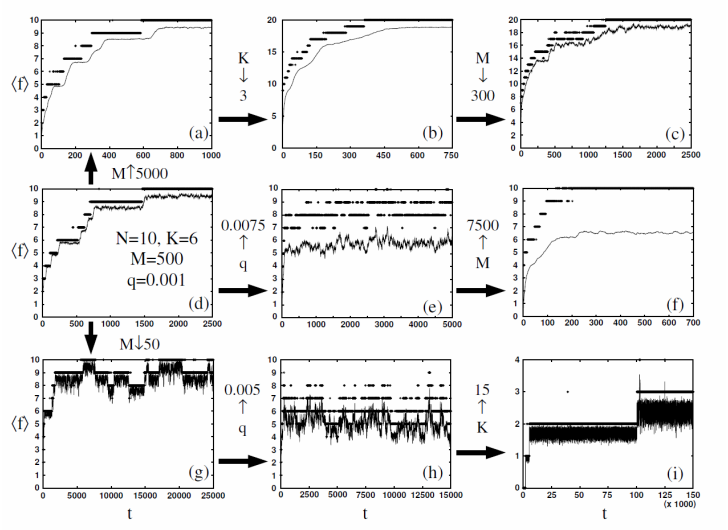


Figure 4.50: Evolutionary dynamics on the **royal road landscape** for different mutation speeds (q), block sizes (K), and population sizes (M). In the top, you see a classic view of **punctuated equilibrium**. For higher mutation rates, fitness is suboptimal, but some fitness can be retained. This attenuates our idea of the **information threshold**: even if the optimal fitness cannot be retained, a slightly lower fitness can be kept (panels g and h, for instance). Taken from: (Van Nimwegen, 1999).

In other words, this artificial landscape serves as a reminder that information accumulates up tot the **information threshold**, and the problem need not be an all-or-nothing scenario. Earlier, we thought of a catch-22, however, if some functionality can be reliably maintained in part of the sequence, that attenuates the problem.

AVIDA computer programme

We now turn away from constructed landscapes and to the AVIDA computer programme. In it, we aim to study what happens during evolutionary optimisation. Multiple programmes with 100 lines of code are instantiated, and the internal code can be mutated. The programmes need to perform a specific function, and there is a fitness based on how well it is done. The programmes compete in the population based on their fitness, which determines who gets to replicate. A colour-coded map of the evolutionary-trajectory is provided below, as is the epochal evolution that goes along with it (Figure 4.51). More red at a position means that there is more variation in the population, while cooler colours indicate less variation at that code position.

At low fitness, there is lots of variation in the population. The code is optimised by having more positions being of importance to the working of the programme (less variation at those positions, the image becomes more blue). So here, variation goes down, while we have just discussed that neutral evolution should increase variation. However, note that in this case we are evolving towards a higher fitness. If there is an innovation that increases fitness, that will be selected. Thus, variation goes down.

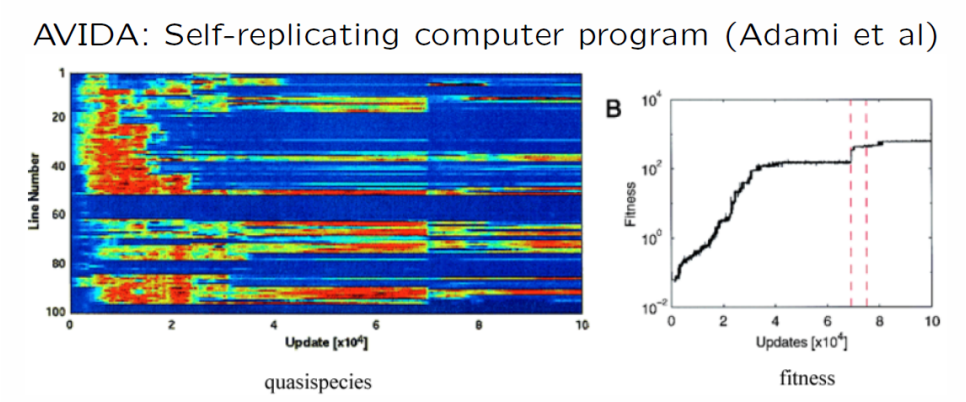


Figure 4.51: AVIDA evolution of programmes through time. When a new innovation is found, this reduces variation. Adapted from: (Adami et al., 2000).

So, what we see here is that if a programme becomes fitter, there is a variation bottleneck at that point as it quickly overtakes the population. During fitness increase, more **neutrality** does not accumulate. However, in the intermittent periods, you clearly see an increase of neutral variation (more red). Also, just after a bottleneck the rate of variation increase is higher. This underlines the point that when you go to a higher fitness, you are less

Mapping insights on RNA adaptive landscapes back to the information threshold

We have covered the **information threshold** in relation to Eigen's work on ODE systems of RNA-like replicators. There, we saw that for mutation rates higher than a certain threshold value, the replicator with optimal fitness would not be the most prominent in the system: in essence, natural selection broke down. Besides lacking space (whose importance we have previously seen), Eigen's model didn't include a quintessential property of RNA: there is a mapping of sequence to structure. This **GPM** (genotype-phenotype mapping) leads to a lot of **neutrality** on the genotype level: genotypes can diverge a lot, while maintaining the same phenotype.

If we map this finding back to the **information threshold**, we can say the following: given that the phenotype is what natural selection acts upon (it is the folded RNA that can catalyse replication with a certain propensity), and given that a phenotype can be kept while the underlying genotype neutrally evolves, there is a relaxed **phenotypic information threshold**. Why? Well, a phenotype can be maintained while the underlying sequence is constantly mutating. In fact, what used to be a problem (selection cannot maintain the fittest genotype in the population) is now a boon: the fittest *phenotype* at any time point can be retained, while maximum neutral percolation over the *genotype* space ensures that fitter phenotypes are found very fast.

Coding structures

Introduction

In the previous section we have discussed the concept of **neutral paths** and neutral networks and their role in allowing populations to discover novel structures. However, is there more going on? Is there a directionality on an RNA neutral path? And how neutral is a neutral path? In other words: which code evolves (given neutral evolution)?

How neutral is neutrality? The move towards robustness

In *Drosophila* (fruit fly), it was often observed that new strains had more deleterious mutations than well-established strains (Scharloo, 1991). RNAs, which are evolved, were known to be very mutationally **robust** (Huynen et al., 1993). This means that they did not suffer many deleterious consequences from mutations. If **neutrality** is higher in established strains of fruit flies, and evolved RNA structures are also mutationally **robust**, might it be that neutral evolution has a directionality towards more neutral regions of the genotype space?

The figure below illustrates this point. In it, the authors looked at RNA evolution to a target structure (Huynen and Hogeweg, 1994). They examined the average decrease in fitness after point mutations in two different situations: just after reaching a fitness peak, and after some neutral evolution from that peak. The mean decrease in fitness is higher when mutating away from sequences that have just reached the fitness peak (Figure 4.52, open circles) than when mutating away from those that have neutrally mutated for some time (Figure 4.52, squares and crosses). The effect is more pronounced if cross-overs are allowed.

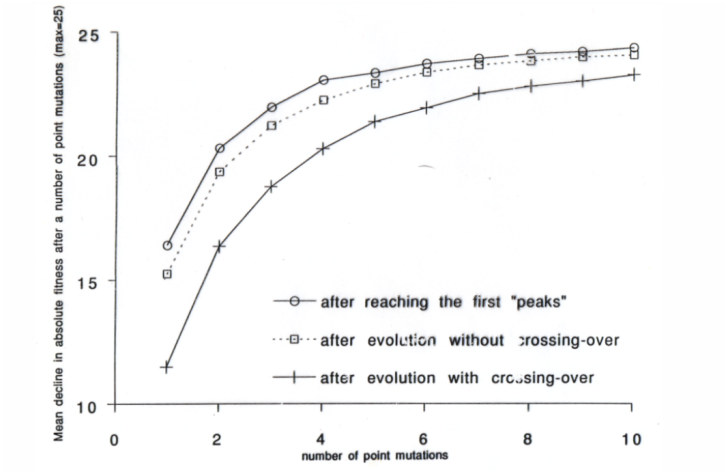


Figure 4.52: The effect of point mutations after just reaching a fitness peak (circles), or after some neutral mutation on that peak with (crosses) or without (squares) crossing-over. Taken from: (Huynen and Hogeweg, 1994).

Thus, neutral evolution leads to areas that are more neutral. Note that this is early work on the subject, and a *suboptimal metric* was used: the mean fitness decrease. Now we see an example where the metric is more in line with what we want to know (Figure 4.53). Here, λ is the fraction of neutral mutations in the **one-mutational neighbourhood**. We start from many copies of the real sequence of phe-tRNA, and the imposed **fitness criterion** is having the phe-tRNA structure. Hence, every molecule is already optimally fit at the beginning, but we observe what happens to the fraction of neutral one-mutational neighbours.

Depicted are the **master sequence** (red, the correct sequence and its neutral neighbours) and the **mutant sequences** (green, those who are not maximally fit, i.e. have a sequence that folds into a different structure). Though there are fluctuations in λ, it is clear that, over time, the fraction of neutral mutants of the **master sequence** increases. What we thus see is that neutral evolution (the **master sequence** and its neutral mutants continuously have the highest fitness attainable) leads towards *flatter, more connected, parts of the neutral network*. This also *increases the connectivity of the neutral network*: if you have more neutral neighbours, it is easier to percolate away over the neutral network.

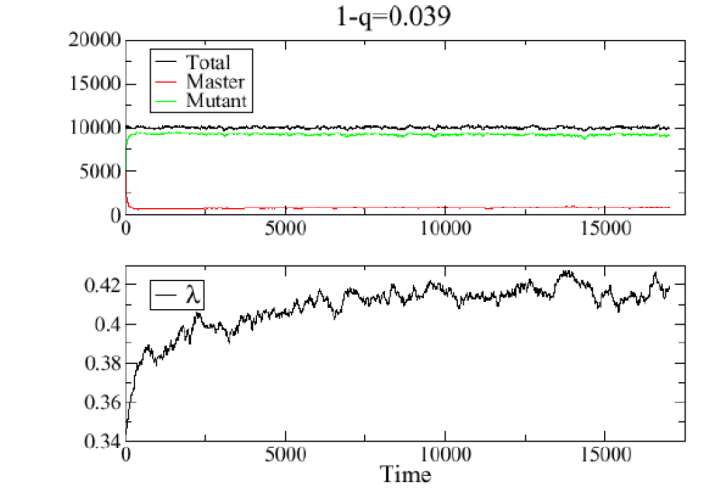


Figure 4.53: The effect of point mutations after just reaching a fitness peak (circles), or after some neutral mutation on that peak with (crosses) or without (squares) crossing-over. Made with RNAevol, which you will be using as well during the practicals.

You do not go to the highest robustness that exists, but you do go above the average robustness or connectivity that exists in the neutral genotype space. It depends on population size $*$ mutational rate where you end up (population size because 1,000 individuals can walk farther while taking neutral steps than 10 individuals, mutation speed for the obvious reason that exploration is limited by mutation rate). We can ask why real phe-tRNA is not more robust, if we can make it so in a measly 10,000 generations. We assume here that the only selection pressure is folding into the correct structure. However, we only look at the secondary structure, and fail to account for what phe-tRNA actually does: it needs to be able to undergo conformational changes to fit into the ribosome and get out of it. Therefore, complete robustness at the sequence level (mutational robustness) is, in real life, probably counteracted by the need to make conformational changes.

Why this move towards more **mutational robustness**? **Neutrality** is a double-edged sword. On the one hand, sequences with more neutral mutants will have more fit offspring: if a mutant occurs, it is less likely to be detrimental to fitness. Over the long term, **robustness** thus pays off. On the other hand, more **neutrality** allows a greater exploration of other possible phenotypes that might be fitter. What you would initially think to be opposites (being readily able to adapt vs being robust against mutations) are thus perfectly reconcilable. This leads us to the study of the **evolution of evolvability**: the idea that evolution can work to optimise *later* evolution. We have come far from the strictly adaptationist view. Now let that all sink in while I grab myself a cup of tea.

Recent experimental example of evolved robustness

Slurp Aaah..I do like me a nice warm cup of dried fermented leaf pulp. Proper hydration is key, kids. Where were we? Oh yes. We have introduced this move towards **robustness** based on work from some years ago by van Nimwegen. The decrease in **robustness** had already been noted in *Drosophila* (Scharloo, 1991). Some recent experimental work verifies these results.

In this study, the authors looked at the effects of single point mutations from the wild-type sequence of Hsp90 and from 7 of Hsp90's (almost) neutral mutants (mutations were induced between base pairs 580-590 of the sequence). Thus, you have a direct comparison of the **robustness** in the **mutational neighbourhood** of the wt and newly created (not evolutionarily optimised) Hsp90 sequences (Bank et al., 2015). The results are telling (Figure 4.54). The wt has many neutral mutants (high peak at growth rate of 1.0). Relative to wt, the mutants of neutral mutants are much less likely to be neutral: they are much more likely to be deleterious (higher peaks on the left). Once again, we see that the optimised evolved form (wt) is much more neutral than other nearly the same sequences, indicating that this property is something that evolves.

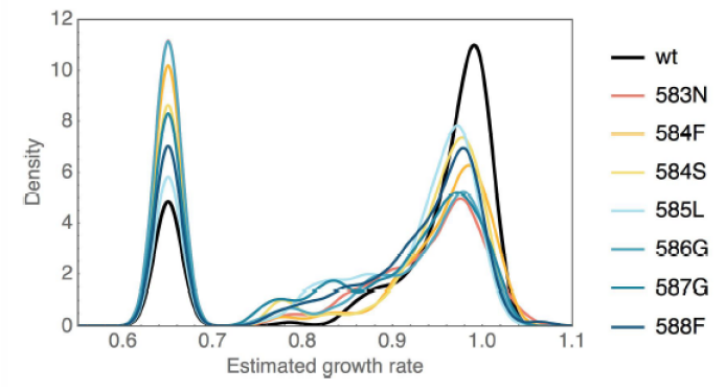


Figure 4.54: The effect of point mutations after just reaching a fitness peak (circles), or after some neutral mutation on that peak with (crosses) or without (squares) crossing-over. Taken from: (Bank et al., 2015).

We will here remark upon the fabled **'U-shape'**: mutations are either neutral or extremely deleterious. If we look more closely, we can see that, in fact, this effect is accentuated in the wt compared to all of its mutants. This is a first hint that perhaps something is up. Details will follow in later chapters, but know that this allows for high **neutrality** (and thus high **robustness**), and also for a high **selection coefficient** relative to the highly deleterious mutants, so that they can be quickly outcompeted.

How neutral is neutral?

We have now seen that neutral evolution is directed towards flatter parts of the fitness landscape: neutral evolution leads towards areas that are more neutral. This leads to both **robustness** and **evolvability**.

We are now forced to ask: what is **neutrality**? How neutral is neutral? If neutral evolution has a goal, is it neutral? What we can say is that a mutation is neutral when it is above the **information threshold** for selection. Yes, there it is again, our good friend and difficult concept the **information threshold**. If a neutral mutation causes an organism to have 0.001 less offspring, that is not something selection could ever act upon. Thus, neutral mutations can be slightly deleterious, but not more so. The fact that neutral mutations, over the long term, lead to more neutral areas of the genotype space which makes descendants of those individuals more likely to be fit does not change the fact that the neutral mutations themselves cannot be acted upon by selection.

Integrating pattern formation, coding structure, and evolution

In a CA we see **spatial pattern formation** which affects *ecological and evolutionary processes*. Higher order patterns arise by themselves and feedback on the evolution of replicators, i.e. *they enslave the evolution of replicators*. This is a two-way process whereby replicators interact to generate patterns which then feedback on replicators.

On the other hand we have seen that replicators have a coding structure which leads to a particular **genotype-phenotype mapping**. In organisms, there is therefore a complex transformation from a code to an organism, and we could well argue that this code, itself, is evolved.

These observations lead to the following questions:

1. *What is the best code for good evolution?* (Is that, for example, a smooth landscape, or something completely different?)

2. *Given a code, how does evolution proceed?* What evolutionary process occurs? In RNA, the landscape is not smooth, but we found out that its properties are in fact very helpful to the evolutionary process: redundant coding and **neutral paths** *help* rather than *hinder* evolution.

3. *Given an evolutionary process, what kind of code evolves?* Can evolution select for certain kinds of coding because properties of the landscape are not the same everywhere. In other words, can evolution select a place in genotype space such that the **adaptive landscape** there has certain properties?

We have already started to answer the first and second question. In the following pages, we will look at them more closely, before moving on to the third question.

Constructed landscapes and landscapes as a metaphor

The **adaptive landscape** metaphor is useful to give an intuitive idea of what kind of things could be going on, and was originally proposed by Sewall Wright (Wright, 1986; Skipper, Jr., 2004). However, we have seen by now that the concept has its flaws:

1. Too few dimensions: 2D or 3D adaptive landscapes are anything but realistic. Evolution has many more dimensions to work with.

2. The dimensions are fixed: bigger genomes have more dimensions (nucleotides) to change, and therefore have more dimensions to work with. Evolution changes the dimensions of the landscape: it increases or decreases the genotype space.

3. It only works well with point mutations, while there are many more mutational operators which make the "proximity" of genotypes hard to define (think about cross-over, inversions, or large duplications and deletions).

4. There is constant fitness: in the landscape metaphor, one assumes that the evolutionary pressures (environment) are constant. However, they would normally change during the evolutionary process (either from fluctuations outside of the system, or by means of influences from inside of the system!).

In the following we study constructed (very much artificial) landscapes that explore particular aspects of **adaptive landscapes**. Moreover, we aim to make a connection between **coding structure** and **spatial pattern formation**, both of which can leave **evolutionary signatures** in their own right. Here, our main question is: *what kind of traces do evolutionary processes leave in evolved entities?* We mean traces that are:

1. Not there because they are functional

2. Not there for biochemical reasons

3. Solely present because the system has evolved in a certain process/way

Concluding remarks RNA genotype-phenotype mapping

What have we seen in these RNA landscapes up until now? We have seen that there are not many true **local optima**. Instead, detours are available in the high-dimensional space that evolution can work with. There is *percolation over neutral networks*: sequences keep encountering new structures in their **one-mutational neighbourhood** as they amble through sequence space. There are **punctuated equilibria**: a mutation that is difficult at one time can, through **neutral percolation**, be simple to achieve at a later time. The underlying sequences for a certain phenotype are continuously changing, leading to a massive parallel exploration effort for better phenotypes. There is thus diffusion on neutral networks. During adaptive evolution, most mutations are actually neutral, with jumps in fitness due to non-neutral mutations. There are *intercalating neutral networks*: neutral wandering sequences might meet, and there are neutral subpopulations due to stochasticity. New structures are constantly sampled from different parts of the genotype space, and there is a **shadow of similar structures**: you keep seeing a small set of structures related to your initial structure throughout your neutral walk. We have seen The RNA landscape is ideal for evolution. Lastly, evolution on neutral paths proceeds towards flatter (more neutral) parts of the landscape. Why? Because a larger mutational robustness yields an individual with more highly fit offspring (more mutants are neutral). At the same time, on a population level, more neutrality yields more avenues to adapt (since a larger diversity of structures can be "seen" by evolution)!

What can we now say about what coding evolves in evolutionary systems and how they work?

1. Given a code, what are its evolutionary dynamics? For RNA, these dynamics are **punctuated equilibria**, **neutral paths**, etc.

2. Given a problem: how is it best to code? The expectation would be that a smooth, non-redundant search space is the best suited to find solutions. In reality, evolution in RNA landscapes is so powerful because there is **neutrality**! This redundant coding thus works very well indeed.

3. Given evolutionary dynamics, which code evolves? We see that a coding evolves that moves towards **robustness** and **evolvability**. **Neutral percolation** leads towards a more neutral area of the genotype space.

Caveats of the work on RNA landscapes

Though they do not invalidate the concepts we have learned from looking at RNA, we should note that there are some caveats.

1. RNA structure prediction is not perfect: if you look at the 16S RNA sequence for many bacterial species and assume that conserved parts have conserved folding, you get a different folding picture than when you calculate the minimum energy folding. This is probably, in part, due to the fact that RNA starts folding from the moment it is made. There are some prediction tools that take this into account and perform better (Hofacker et al., 2002). However, we should note that we did not go for complete accuracy, just for the general behaviour of the system.

2. Alternative basins of attraction: RNA structures do have one true minimum energy structure, but there are often many structures which are *almost* as good. In practice, RNAs in a chemical environment could well switch between several different foldings continuously, which might well impact our conclusions (Zuker, 1989; Fu et al., 2015).

An example of the power of neutrality: model of the insulin pathway by Wagner

To illustrate how important **neutrality** can be as a concept in our understanding of other biology, we now focus on an example by Wagner (Wagner, 2015). Here, they took a literature model of the insulin pathway (and resulting glucose uptake). This model contained mass action equations (i.e. reaction kinetics) and these were translated to a system of ODEs. Wagner then took 13 kinetic parameters and 2 feedback loops, and evolved them (Figure 4.55a). These parameters thus form a sort of pseudogenotype: changing the different parameters and seeing what happens will allow us to get a sense of the **adaptive landscape** of the insulin pathway. The parameters were varied by *six* orders of magnitude. Wagner then simulated a pulse of glucose (Figure 4.55b), and classified the behaviour of the system as normal or diseased based on the glucose uptake curve in time (he was quite liberal in what counted as healthy, but that does not change the findings) (Figure 4.55c). Though we will forego exact quantification here to keep things short, the findings were that though there were about four times more diseased phenotypes than healthy phenotypes, a huge range of parameter values produced healthy phenotypes.

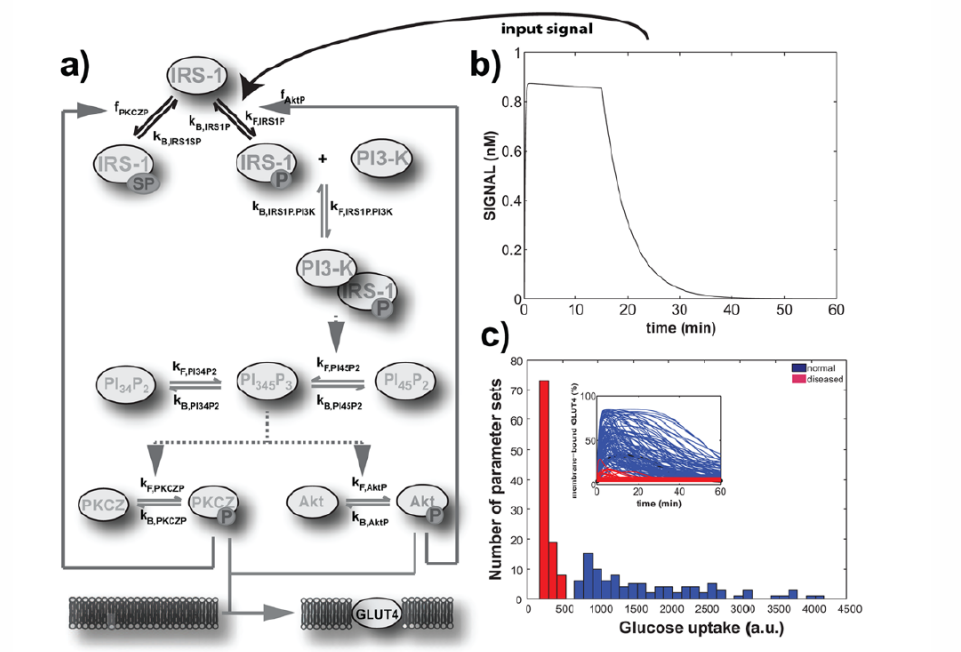


Figure 4.55: Illustration of the insulin pathway and its workings. a: reactions and parameters that Wagner used to parameterise his model and make his pseudogenotype. b: the input glucose signal that Wagner provided to all different parameterisations of the model. c: healthy (blue) or diseased (red) glucose uptake for all different parameter sets. Inset: visualisation of the glucose uptake curves generated by the different parameter sets in the model. Taken from: (Wagner, 2015).

He then took a large set of healthy parameter sets, and asked whether changing one of the 15 parameters randomly had a strong deleterious effect. As it turned out, most random changes to healthy sets of parameters gave no noticeable change in the glucose uptake curve at all. On the other hand, similar to what we have seen in the RNA landscape, in some parameter regimes changing a single parameter made the phenotype completely unhealthy. *Which mutations are strongly deleterious is very dependent on the genetic background!*

To explore this further, Wagner then allowed mutations in parameters to occur eac time step, while only allowing healthy individuals to survive. This means you introduce neutral drift into the insulin uptake pathway, and only individuals without detrimental phenotypic mutations survive. Now, he observed *strong drift in the causes of disease*: which parameter was most important for the generation of a diseased phenotype depended very strongly on genetic background, and that changed rapidly under this regime. In fact, after only about 10 generations, the correlation of what is a current causal disease mutation to what was a causal disease mutation 10 generations ago is close to 0. This leads to a very fundamental question: what is *the cause of a disease*, if, within 10 generation, the genes in which mutations are causal to disease can shift completely under neutral evolution?

The kicker is that Wagner also performed some *in silico* GWAS studies on these pseudogenotypes. In GWAS studies, you do logistic regression on effects (in this case, the parameters that cause a disease) and add all these effects up (i.e. you assume effects are additive) (Visscher et al., 2017). However, as has been shown in this chapter, there is a **non-linear genotype-phenotype mapping**: the genetic background matters very much for which parameter mutation is responsible for causing a diseased state. So just adding up the causal mutations in a population and inferring the cause of disease from that is a dangerous business. The result in this case was that in different populations where Wagner had simulated case-control groups (like one has in GWAS: healthy people and diseased people from certain populations), the significance of a parameter for disease could vary immensely: the parameter with the *lowest variation* in significance for disease causation varied over *47 orders* of magnitude between populations.

These findings thus cast quite some doubt on our ability to infer the causes of disease from non-human

model organisms: if we find the cause of disease in mice (note that lab mice are often extremely genetically similar strains), not only do we have to contend with the fact that there are actual physiological differences, we should also take into account that neutral evolution in all conserved mechanisms could have completely changed what is causal to disease.

This example thus provides insight into how important examining the landscape of (neutral) mutations can be. In this case, it shows us that GWAS studies should be interpreted very carefully, and that neutral evolution can greatly influence what is causal to a disease phenotype. Therefore, inferring causes of disease from model animals is a difficult task.

RNA world in space: structure-based modelling

Until now, we have done two things. They can be summarised as a world without true RNA, and RNA without a true world. That is to say, we simplified the form of RNA in spatial models into simple replicators with a few parameters such as catalysis and looked at the **information threshold** and **stochastic error correction** in vesicles. Thus, we studied the world, but without real RNA properties. In other words: we looked at ecosystem complexity. The last chapter has been about studying the **complex multi-one genotype-phenotype mapping** that true RNAs have, and how this influences what evolution does. However, we did not incorporate a world or any spatial structure: we just set fitness targets and looked at what evolved. Thus we looked at individual complexity without the ecosystem. Now we will incorporate the two and look at structural modelling.

Here, we will look at examples that incorporate structured entities (such as RNAs with true qualities of RNA) and ask whether we can obtain individual-/ecosystem-based complexity, and what sort of coding is needed to obtain it. While doing this, we will see that RNA is even more interesting than seen so far.

Model features

How do we model this? We take structured individuals. In this case, those will be RNA sequences, which are either a + or - strand. If these fold into a predefined (class of) structure(s), they are considered a replicase. There is no predefined target or fitness. There are also no predefined interactions, but there are predefined reactions. If we contrast that with the earlier replicator-parasite system (**RP system**), the difference is that in that case, one RNA was the parasite, and the other was the real replicator. Here, we make no such distinction beforehand. We simply model sequences that, when folded into a certain class of secondary structures, can function as replicases. This requires complex formation, where the 5' end of one string binds to the 3' end of the other. Complementary replication is also implemented. The system is modeled in space (CA formalism). This model thus allows complex formation of compatible neighbours, leading to replication of the complementary strand (Figure 4.56). Initial work was done by Takeuchi and Hogeweg, later extensions by Colizzi and Hogeweg (Takeuchi and Hogeweg, 2008; Colizzi and Hogeweg, 2014).

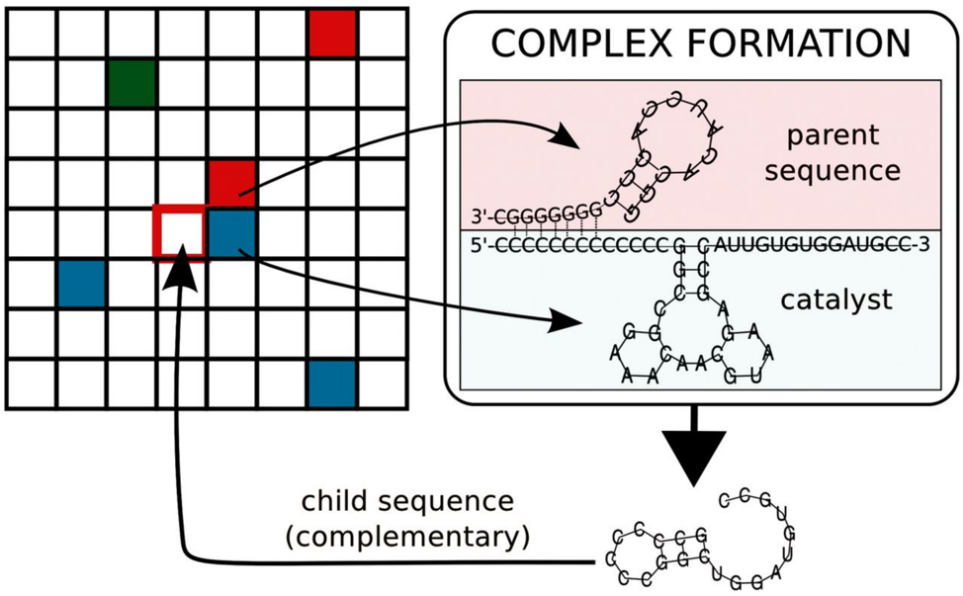


Figure 4.56: How the model of RNA in an ecosystem works. No replicators or parasites are defined beforehand. Rather, certain sequences can bind to one another and form a complex if they are close to each other on the CA field. Complementary replication can then take place if there is empty space nearby. Taken from: (Colizzi and Hogeweg, 2014).

For the next couple of paragraphs, we will thus work with these constraints/characteristics:

1. A class of predefined structures are assumed to be replicases
2. There is complex formation, and there are + and - strands. Therefore, replication yields a complementary string.
3. Complex formation happens with the 3' end of one string ligating to the 5' end of the other string. In other words, interactions between sequences *evolve* and this influences the genotype and phenotype (see Figure 4.56).
4. In all the below, sequence logos show the average genotype of the population.

By observing these behaviours and observing and dissecting what evolves, we hope to learn more of the mapping from genotype to phenotype to interactions in the ecosystem that evolves, and how these processes feedback on one another (Figure 4.57).

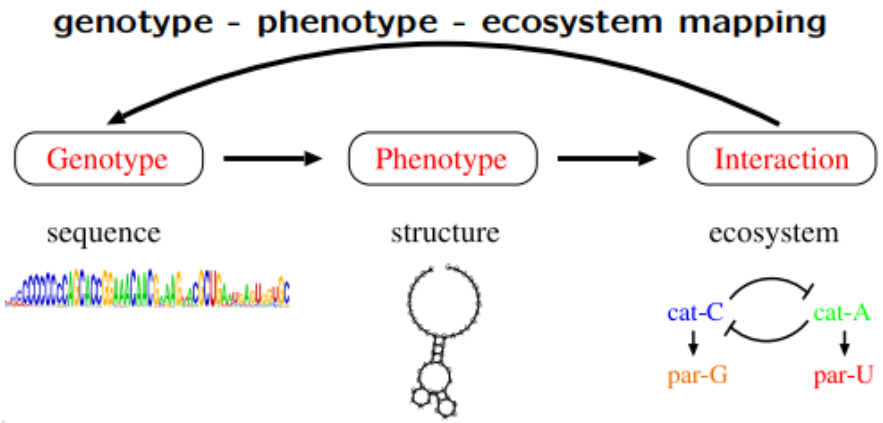


Figure 4.57: What the model investigates. We will look at how the mapping from genotype, to phenotype, to interactions within the environment shapes what is coded for emergently. Taken from: (Takeuchi and Hogeweg, 2008).

We will go about the results in a reverse order: starting with how the system behaves itself in a regime with high mutation rate and gradually lowering the mutation rate to see what happens. The system was started with a particular sequence that is a replicase. At the start, mutation rates needed to be quite low (so this model is not an immediate answer to the **information threshold** in that sense), but after evolving for some time, the system of RNA-like replicases could deal with ever-higher mutation rates.

Mutation rate of 0.015: a single quasispecies

Under this regime, if mutations are stopped at a certain time point, the resulting species is monotypic (i.e. has one genotype). In other words, everything we observe here is one **quasispecies** of fit sequences and their mutants. This was confirmed by looking at the phylogeny of these sequences: the relation between all sequences in the population was plotted in a phylogenetic tree, along with the information of whether or not these sequences folded into a catalyst. With this high mutation rate, there was clearly only one **quasispecies** (Figure 4.58).

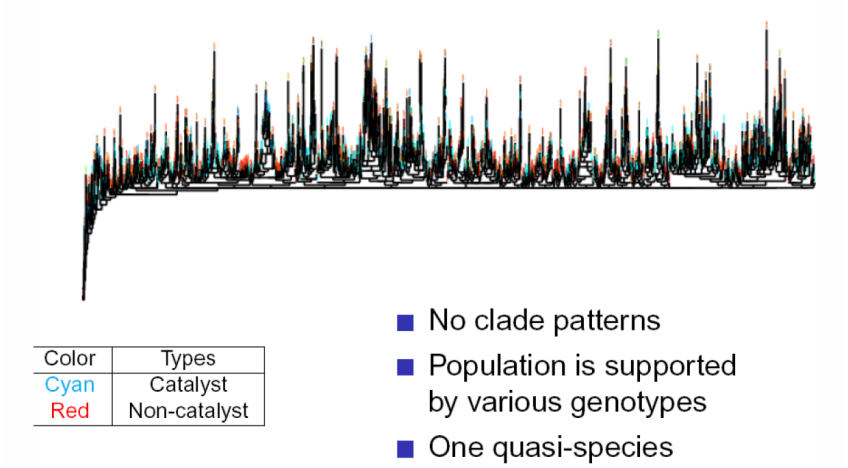


Figure 4.58: Single **quasispecies** at high mutation rates. Cyan and red are catalysts and non-catalysts respectively. (more colours are displayed, but are irrelevant for now). Based on: (Takeuchi and Hogeweg, 2008).

We can now look at the sequence logo to see what happens on the sequence level. There, we see that the replicase that exists is a very tight **quasispecies**, where most positions are extremely conserved (Figure

4.59). Below the sequence logo is a representation of the minimum energy folding of the consensus sequence in **dot-bracket notation** : matching parentheses mean bound bases, dots mean unbound bases. Red parentheses indicate the main folding/folding of the functional replicase, whereas black parentheses indicate other foldings that don't lead to a functioning replicase.

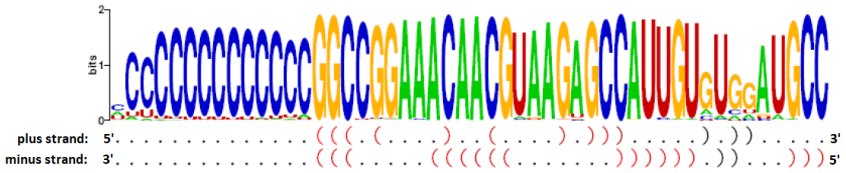


Figure 4.59: The single **quasispecies** at high mutation rates has a sequence that is very conserved, with many Cs. Based on: (Takeuchi and Hogeweg, 2008).

On the plus strand, there is a long open 5' end, which aids complex formation. On the minus strand, there is a long open 3' end, for a similar reason. Additionally, it lacks an open 5' end. This evolved to prevent non-functional complex formation. There is a majority of G and C in the sequence: when these bind, they bind strongly. This replicase can replicate itself. Finally, the Us interspersed in the 3' end of the plus strand make sure that it does not fold back on itself (the many Cs and Gc might otherwise allow self-binding, which is non-productive).

Mutation rate of 0.013: speciation

If the mutation rate is turned down to 0.013, two distinct classes of sequences appear in the phylogeny: one of which is a catalyst, another which is not: we see the emergence of a parasite (Figure 4.60).

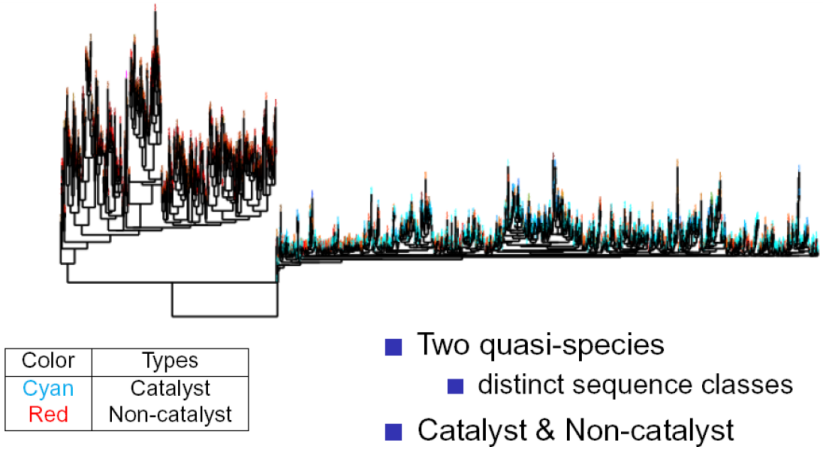


Figure 4.60: Slightly lower mutation rates lead to two distinct **quasispecies**, one of which is a parasite. Based on: (Takeuchi and Hogeweg, 2008).

If we examine the sequence logos of these two **quasispecies**, we see that the non-catalytic lineage (parasite) has just one helix (Figure 4.61). Additionally, it has a long open 3' end on both the plus and minus strand, and a hairpin in both strands. In this way, the 5' end of both strands is inaccessible, further enhancing the changes for binding the 3' end. Finally, it has many Gs at its 3' end, so it binds nicely to the replicator's Cs. The catalyst is almost completely the same as the one that survives in the high mutation rate regime.

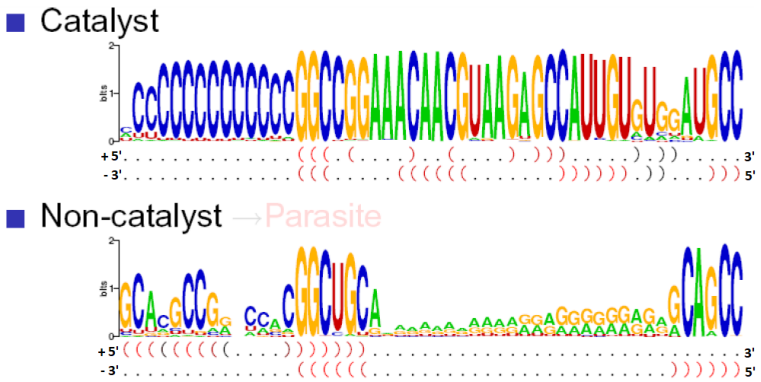


Figure 4.61: Slightly lower mutation rates lead to two distinct **quasispecies**, one of which is a parasite. Based on: (Takeuchi and Hogeweg, 2008).

Lowering the mutation rate yet further

If the mutation rate is lowered to 0.008, there is another newcomer: the A-catalyst. This novel replicase has an extremely high neutrality: 50% of SNPs still yield a functional replicase. At this point, the system encompasses two replicases which behave very differently. Finally, if the mutation rate is then set to its lowest value (0.004), this new replicase gets its own parasite: the U-parasite (Figure 4.62).

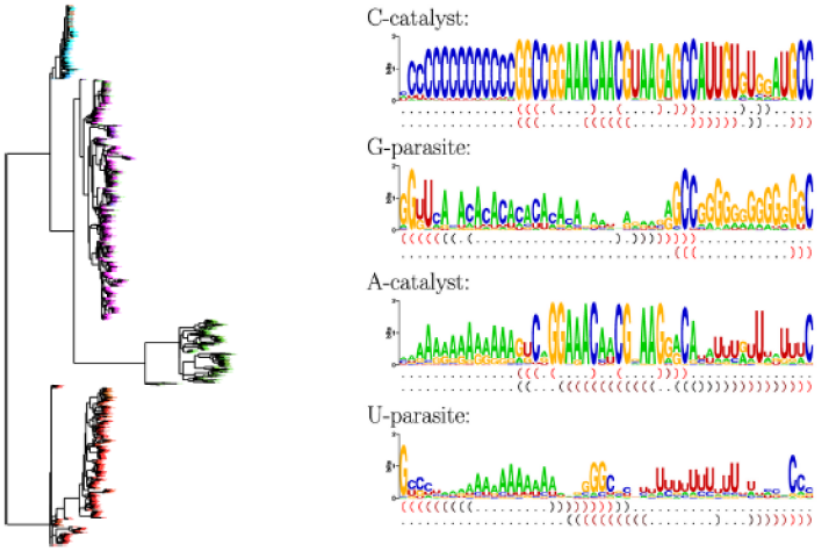


Figure 4.62: The lowest mutation rate leads to four different **quasispecies**, two of which are parasites, and two of which are catalysts. Based on: (Takeuchi and Hogeweg, 2008).

The spatial dynamics are such that the first catalyst replicates, gains a parasite, which generates space in the system. Upon lowering the mutation rate, this can be exploited by a different catalyst, which then gets its own parasite when the mutation rate is lowered again. If the mutation rate is set to 0 in the evolved system, the same ordering of different lineages still occurs. Thus, even when considering this as a static ecosystem, these are stable species. Their survival does not depend on mutation (as was true for the mutation rate of 0.015.). This is different from the eco-evo model earlier, where we saw that diversity was maintained by continuous mutations. The spatial pattern, in this model, is one of chaotic waves.

Let us cut to the chase and discuss what we have learned. It is important to remember that this is a *structure-based spatial RNA system without predefined interactions or fitness*. Nevertheless, we see

a very interesting phenomenon: there is only one true resource here, and it is space. In other words: there is only one niche. Speciation occurs, because we end up with a diverse ecosystem with different lineages. It was deemed impossible that different species could coexist on one resource: survival of the fittest means that the best exploiter of a niche wins. This is thus another **existence proof**: multiple species can, in principle, coexist in a predefined "niche", as they shape their own niches. Additionally, the replicator-parasite system (**RP system**) that we have looked at before was quite stable in space, but there we defined the interaction structure. Here, the interaction structure between the different catalysts and their parasites evolved itself: we merely imposed complex formation and a class of structures that could replicate.

Subconclusion structure-based modelling

We have seen that sequence, structure, and interactions do evolve. A very stable, multi-(quasi)species system evolves. The interaction topology is different from what we have studied before. The variability in the exosystem increases with decreasing mutation rate: lower rates lead to speciation. However, we did see that an ecosystem-based 'solution' is only found at lower mutation rates. In other words: only at low mutation rates can different species coexist and is there room for parasites. Thus, though this finding is interesting, it does not solve the problem of the **information threshold**. On the other hand, it does show that given time, evolution can cope with high mutation rates (the system needs to be started on lower mutation rates, but if they are increased afterwards, the system can cope). Lastly, we see that the evolved genotype, phenotype, interactions and spatial structure are all interdependent (Figure 4.63).

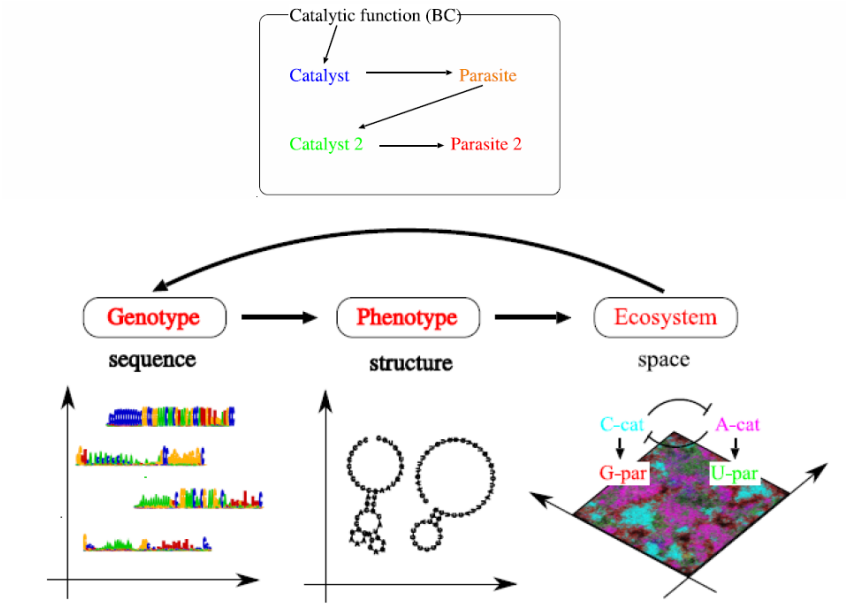


Figure 4.63: The evolved structure in this system depends on all levels. Based on: (Takeuchi and Hogeweg, 2008).

Evolution of coding structure at high mutation rates

If we look more closely at the situation with very high mutation rates, we see an interesting picture. As it turns out, the **master sequence** in the **quasispecies** is a minority in the population. It is a self-sufficient replicase, where both strands have an open 3' end. There is only a structure in one of the strands, and the strand with the structure has its 5' end free. We have learned that high **neutrality** is favourable, especially in regimes with high mutation rates: if more of your descendants have neutral mutations, the population is fitter. However, in the single **quasispecies**, the **mutational neighbourhood** is filled primarily with non-replicators (Figure 4.64a). If, instead, you take random sequences, find whether they are replicases, and for those replicases see how many **mutational neighbours** are still replicases, that is about 25%. The optimal replicator (optimal replication based on the constraints of the system) has more

than 50% of its mutants as replicators. Somehow, there is thus an evolutionary pressure towards a very high fraction of deleterious mutations for the **master sequence**. It does something completely different than random and optimal (in terms of replication potential) replicators. *This is a special property of the evolved system.* Why does it occur? That is what Colizzi and Hogeweg wanted to find out (Colizzi and Hogeweg, 2014).

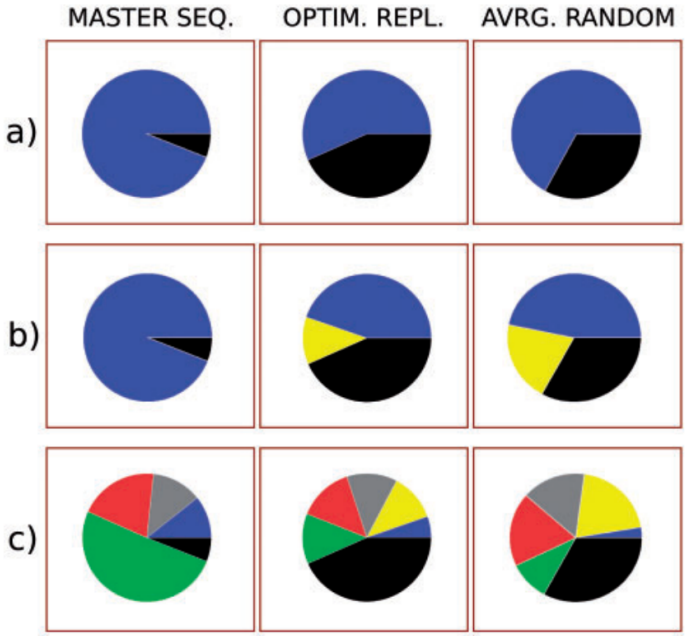


Figure 4.64: One-mutational neighbourhood of the **master sequence** evolved in the system compared to random and optimal replicators. Replicators (black) and non-replicators (blue) are shown, along with further functional subdivision. a: (non)-replicators in the **mutational neighbourhood**. b: (non)-replicators and parasites (yellow). c: (non)-replicators, parasites (yellow), helpers (green), stallers (red) and junk (grey). Taken from: (Colizzi and Hogeweg, 2014).

If we look at the fraction of parasites arising due to mutation (replicable molecules which are themselves not replicases, they are replicated on the + and - strand), there are quite some in the optimal replicase and average random sequence (Figure 4.64b). However, there are *none* in the evolved sequence. Why? A parasite close to you in space is deleterious. Since reproduction is local (this is a CA model, after all), parasites in your **mutational neighbourhood** will immediately compete with you for space. It is thus a very smart idea to make sure you do not generate parasites near yourself easily. Parasites are something we have seen in earlier models, so we expect them. However, what else is there in this evolved **mutational neighbourhood**? As it turns out, there are many sequences with slightly different functions (Figure 4.64c). *Helpers* (green) have a catalytic structure and an open 5' end, but no open 3' end. Their complementary strand can be replicated and forms a helper. However, they cannot replicate themselves. Thus, these sequences act like a worker ant or bee: they help to replicate the **master sequence**, but cannot replicate themselves and therefore do not compete. In the evolved sequence, about 50% of **one-mutational neighbours** are helpers, which has clearly been selected for. *Stallers* (red) have no catalytic structure, but have an open 5' end on at least one of the strands. They therefore bind, but don't replicate, other sequences. Junk (grey) doesn't really do anything.

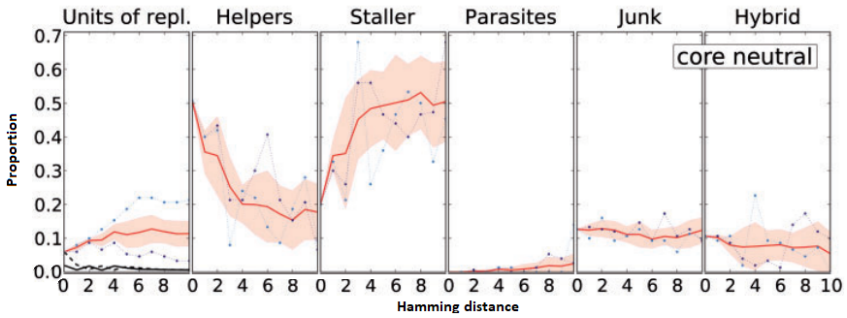


Figure 4.65: Mutational neighbourhood) of the master sequence evolved in the system compared to random and optimal replicators at larger Hamming distances. Taken from: (Colizzi and Hogeweg, 2014).

You see that the number of helpers drops quickly for larger **Hamming distances**, while the number of stallers rises sharply as more SNPs happen (Figure 4.65). That is interesting. Stallers are bad for everybody, so you wouldn't want stallers in your close **mutational neighbourhood**. However, we see here that somewhat further away, the amount of stallers increases. That is positive: the **mutational neighbourhood** of competitors (your mutants) has more stallers, and is therefore more stalled in replication than you are! This works because diffusion is low relative to mutation.

A question that remains is whether helpers really help. They can, but how necessary are they? To test this, the model was now changed so that any helper that evolves is immediately turned into junk or empty space. In that case, the system goes to extinction. This is a pretty strong hint that they are crucial. To further test this, an ODE system was made whereby the fraction of helpers in the **mutational neighbourhood** could be altered, and the dynamics of the system were observed (Figure 4.66). The question was at which mutation rate the system would die out, given different fractions of helpers in the **mutational neighbourhood**. If parasites were present, higher fractions of helper mutants actually hastened extinction. If parasites were absent, more helpers in the **mutational neighbourhood** increased the mutational rates that the system could survive. So, in this sense, the fraction of helpers determines where the **information threshold** is. In this case, the **information threshold** is on a functional level, not on a sequence level. Without parasites, helpers extend the area before you are over the **information threshold** relative to the **master replicator**. If parasites are there, helpers actually hasten the demise. This is why the evolved sequence has no parasites in its **mutational neighbourhood**.

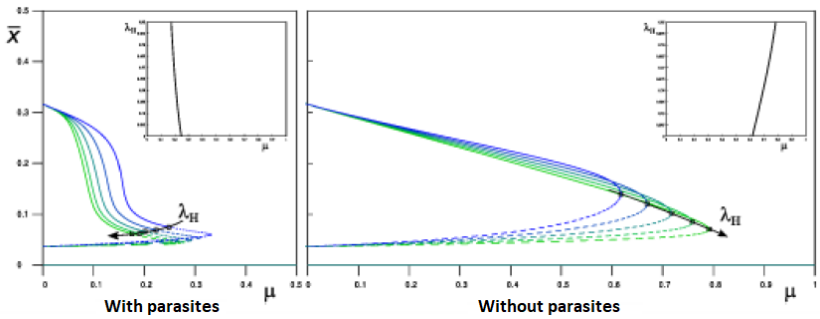


Figure 4.66: ODE model to investigate survival of the system with and without parasites, if the fraction of helper mutants in the **mutational neighbourhood** is altered. Taken from: (Colizzi and Hogeweg, 2014).

Lastly, we will discuss the part that stallers play. In some sense, they are bad for the whole system: whether parasite, replicator, or helper, stallers can bind you for some time while nothing happens. If all

stallers that arise are changed into junk, the density of replicators is increased. However, the **master sequence** is also changed, and some form of pseudo-stallers evolve, which are less strong stallers that were not in the definition of stallers. Bear in mind that the system can produce all sorts of structures and sequences, but all these classes are defined *in hindsight*! In this case, by disallowing what we had defined as stallers, the system nonetheless managed to make sequences that also stall a bit. If these pseudo-stallers are also removed, a parasite lineage evolves! Thus, having stallers in the system prevents parasites.

What we have just described evolved in 6/8 simulation runs. There are thus two qualitatively different cases, where the **quasispecies** is incredibly steep: everybody except yourself is non-viable, so the selection coefficient is very high. However, in competition experiments between a **quasispecies** from 2/8 simulations and from the other simulations, the steep **quasispecies** (that we have just observed, with many helpers) wins. Thus, those 2/8 seem a suboptimal result.

RNA world at high mutation rates and meaning for the information threshold

What can we now conclude about this *RNA world at high mutation rates*?. We saw the following:

1. In this system, where the mapping from sequence, to structure, to interactions can change, a very specific coding structure evolves.

2. One **master sequence** codes for a functionally diverse ecosystem (in its **mutational neighbourhood**).

3. This functional ecosystem was decoded by looking at the effects that mutations had. Hence, this was most clear at high mutation rates.

4. This specific steep **quasispecies** (6/8 cases), shows this effect most clearly. There, the **master sequence** has most control.

This sequence of events is similar to Dawkins' extended phenotype (Dawkins, 1982, 1984): the idea that genes of one organism might be selected for the effect they have on or in another. In this case, the genome of the **master sequence** is selected for the effect that it makes many helpers which help it replicate, and possibly also for stallers further away in the mutational space. We see now that a **quasispecies** can be selected for specific attributes, which is contrary to Eigen's initial assumption for making the **hypercycle**: that other selection pressures would be needed to keep the good replicator in the population. In this case, the good quasispecies is selected for certain attributes, even at high mutation rates (though the system did need to be initialised with lower mutation rates initially). Thus, a **quasispecies** can be selected for certain favourable characteristics (a functional **mutational neighbourhood**, in this case), without additional ecological mechanisms. Whereas classical models don't *evolve the effect that mutations can have*, that is what happens here. We will further explore the consequences of this evolution of the effect of mutations later on. To summarise in one easily digested sentence: individually-coded, but ecosystem-based diversity evolves and persists close the **information threshold**.

Incorporating more RNA features: protocell system with adaptors

We have looked at a functional **mutational neighbourhood** in RNA, and that a sequence can be selected for by virtue of its mutational neighbourhood. Now, however, we will incorporate more features. As has been mentioned before, minimum energy structures are not all there is to RNA. Structures fold and unfold in thermodynamic equilibrium and there are many structures with about the same minimal energy. Additionally, different structures can be formed by binding to something (*e.g.* binding clay surfaces or binding eachother). In populations of RNA, there might well be "adaptors", RNA molecules that bind another RNA, and shield some of its positions during folding.

To investigate what this does, we will look at a new model (de Boer and Hogeweg, 2012). In this model, we simulate protocells with RNA. 25 RNA structures which are functional are predefined so they give fitness. The higher the amount of functional structures in a protocell, the higher its fitness. However, all other RNA structures are toxic. This is thus one of the few models that incorporates that most RNA

structures might actually be negative for a cell (they could interfere with the workings of needed RNA molecules, for example). Cells compete in space. Also, adaptors are predefined as a possible interaction between RNAs (they can bind to other RNAs, and shield the bases that they bind from the minimum energy calculations, thus changing the folding of RNAs that they bind). The RNAs in this model do not replicate by themselves in the cell: in a sense, this model is therefore not a true **protocell model** as we have seen it before. Here, instead, the RNAs replicate upon cell division. The question: how does the system cope with high mutation rates? The figure below is a schematic representation of the model.

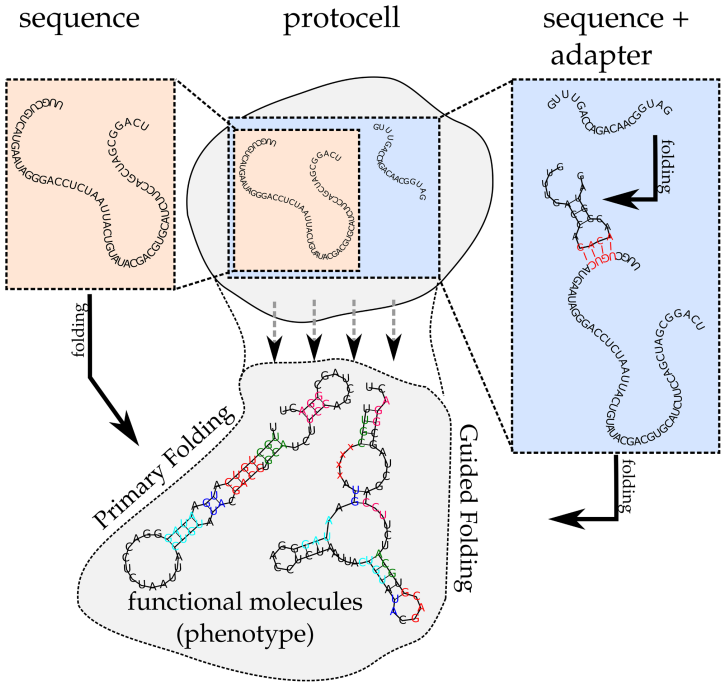


Figure 4.67: Workings of the model with adaptors. Predefined RNA structures yield fitness, whereas all others are toxic to the cell. Adaptors can bind to RNA sequences, blocking some bases from the minimum energy folding calculations, and allowing them to fold into different secondary structures. Taken from: (de Boer and Hogeweg, 2012).

The first difficulty is defining a set of RNA target structures. These structures should not be too easy, but also not too hard to obtain. Additionally, they need to be different enough from each other. Then, there is the question of how strictly to enforce a target structure: should one base that is differently bound matter, or can we be more coarse-grained? To implement the latter, only the so-called Shapiro structure was taken into account. This only counts the number and relative positions of helices, loops, hairpins, and other higher-level structural elements of RNA folding (Shapiro, 1988). Thus, sequence length is not important, and a string can be translated into helix-hairpin-loop-helix etc. Therefore, each *target structure* in this model is actually a set of structures with the same general shape. To define sets, many random sequences and their foldings were simulated. For set A, a set of structures with intermediate difficulty was taken (Figure 4.69a). For set B, structures from a database of real RNA structures were selected which looked different from each other (Figure 4.69b). Some of these structures were very hard to obtain, while some were very easy (Figure 4.69c).

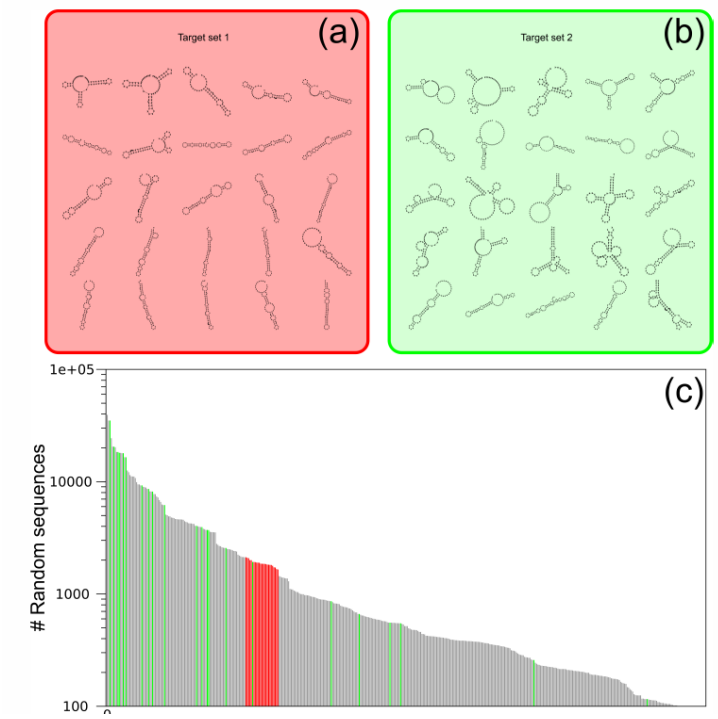


Figure 4.68: Structure sets chosen as targets. a: random structures of intermediate difficulty. b: existing structures taken from a database that cover a range of difficulties. c: number of random sequences that fold into a certain Shapiro structure (a measure for how difficult certain foldings are to obtain). Taken from: (de Boer and Hogeweg, 2012).

The first thing that was checked in this model is the behaviour under various point mutation rates. Genome size and fitness were observed as a function of SNP mutation rate. Fitness per cell is calculated as the number of correct structures - the number of toxic structures. The genome size is the total length of all RNAs. Fitness decreased with increasing mutation rates, but not immensely. In a way, this system is similar to the **royal road landscape**. There, there were bit strings that needed to be correct for every block, and only at that point did the block grant fitness. Here, each block requires a correct structure before fitness benefit, and you have maximum fitness as a cell if all the structures are present. Genome size goes down enormously as mutation rate increases, but fitness does not decrease at the same rate. This is very interesting: somehow, at high mutation rates, functionality is achieved with much smaller genomes. If we map this back to the **information threshold**, we here have a situation where, if mutation rates are higher, acceptable fitness can be *coded in smaller genomes*. Seeing as only small sequences could be kept under high mutation rates, that is interesting: evolution might be able to manage even with short sequence lengths.

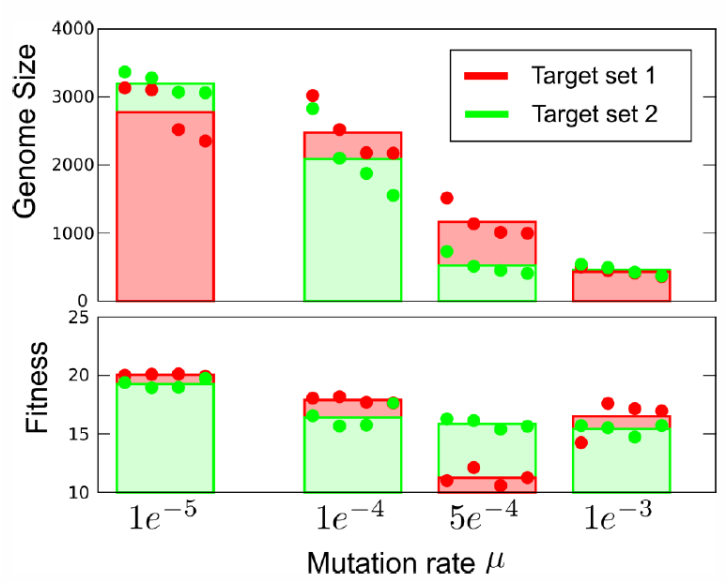


Figure 4.69: Genome size and fitness under different mutation rates. Fitness does not decrease equally with genome size; at higher mutation rates, acceptable fitness can be coded for in shorter sequences (smaller genomes). Red: evolution towards target set 1 (medium difficulty) Shapiro structure. Green: evolution towards target set 2 (real RNA structures). Dots: averages from a simulation. Taken from: (de Boer and Hogeweg, 2012).

So, how does the system work? Two different solutions evolve. The system can evolve one adaptor, and RNA sequences that fold into a certain functional structure with the adaptor and another functional structure without the adaptor (Figure 4.70A). The other, perhaps more interesting, solution is one whereby there is one RNA sequence that folds differently based on many adaptors (Figure 4.70B) (without it folding into any non-functional/toxic structures!). This is an example of **multiple coding**: one RNA sequence can code for many structures by using different adapters. Intermediate mutation rates do not find this multiple-coding solution: it is enforced by a need to encode as much information as possible in as little sequence as possible. Note that in real biological systems, viruses have extremely efficient coding, with many overlapping ORFs and polycistronic RNAs and more, such that even the well-studied Herpes Simplex Virus 1 still had surprises in store when its gene expression during infection was sequenced with Nanopore sequencing (Depledge et al., 2019). Viruses are the real MVPs of retaining fitness while compressing the coding structure to an extreme degree.

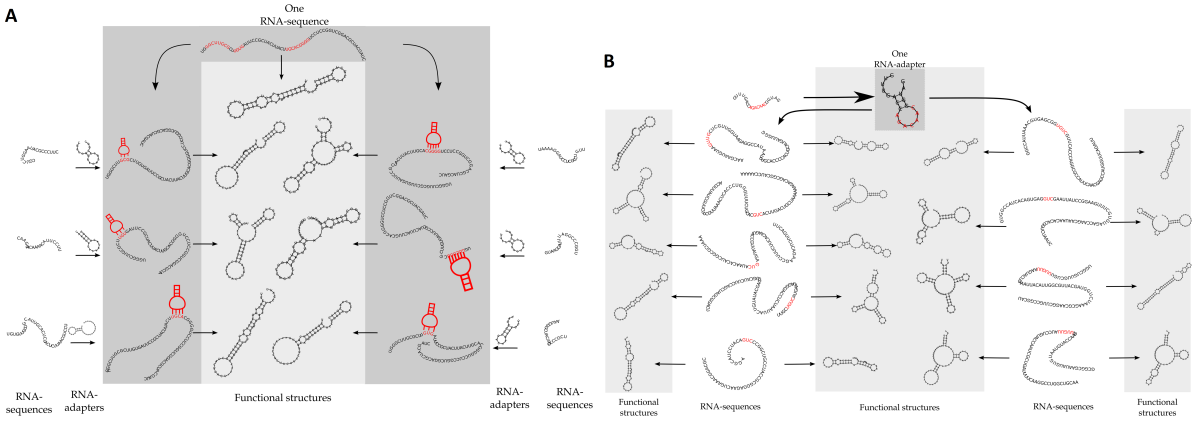


Figure 4.70: Different solutions the system finds to achieve fitness. A: One adaptor evolves that allows a plethora of RNA sequences to fold into two distinct structures. B: One sequence evolves that can fold into many structures based on many adaptors. Adapted from: (de Boer and Hogeweg, 2012).

We can now conclude that RNA is an even more ideal molecule for evolution: **multiple coding** can arise and alleviate the **information threshold**. Thus, the **information threshold** does not necessarily limit functionality, rather, it shapes how information is coded. As a final note, if the system is set to well-mixed conditions, the fitness achieved is much lower, so *local competition helps*, also in this case. This is something you will look at (or have looked at) during the practicals as well. The overarching conclusion (what we can take away as a sort of ground truth from this model) is that *coding structure* can adapt to mutation rate. We see here that at higher mutation rates, coding adapts to encode more fitness in less sequence. Thus, we can say that *evolution converges to being close to the* ***information threshold***.

Chapter 5

Genome evolution, gene regulatory networks and metabolism

Introduction

We now move away from the RNA world, and move onto genome evolution, and eventually move to gene regulatory networks and metabolism. We have seen from all of the model discussed so far that evolution, given enoug degrees of freedom, can solve problems in very creative and surprising ways. In this chapter we will attempt to address questions such as:

- Why are certain genomes very small and other very big?
- Why do certain genomes have a lot of non-coding DNA, while other have very little?
- Why do certain genomes organize their genomes in very particular ways (gene clusters, overlapping operons) while other do not show this trend?
- Why do we see certain network motifs more frequent than others? (feed-forward loops)
- Does evolution evolve to code information in any specific way?

AEVOL: a computational model of genome evolution

AEVOL is a simulation of genomes where the genome structure is fine-grained and abstract (Knibbe et al., 2007; Misevic et al., 2012; Batut et al., 2013). The genomes are bit strings, which have predefined start and stop codons, as well as termination codons for transcription. There are promoter sequences and the bit strings are double-stranded. There are thus coding and non-coding parts of the bit string genome. Genomes replicate based on fitness and incur different types of mutations (Figure 5.1A). Fitness in this model is measured by how well all genes together produce a proteome and phenotype that match a given landscape imposed by the environment (Figure 5.1B, this is a bit complex, read the papers for more information if you so desire). In this model, it is easy to evolve new genes and there is a coding structure on the genome. There are different **mutational operators** that can happen upon replication: SNPs, duplications, deletions, and gross chromosomal rearrangements (GCR) (Figure 5.1C). Note that the chance of these events happening is a per base pair chance. Thus, a larger genome has a larger chance of SNPs or GCRs happening. The ***genotype-phenotype mapping*** *is determined a priori.*

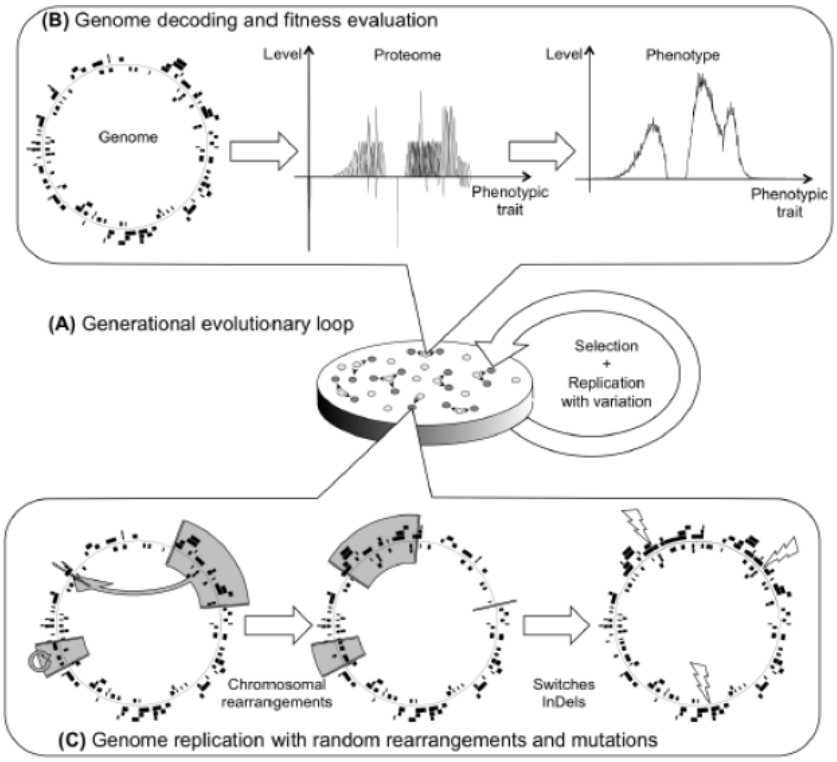


Figure 5.1: Mechanisms implemented in AEVOL. A: genomes in AEVOL are selected (based on some environment-induced fitness) and replicate with mutations. B: the genomes are decoded into a proteome signature and phenotypic signature, which needs to match what the environment imposes for survival. This is the fitness. C: replication happens with a per base pair chance of different mutation types, such as SNPs and GCRs. Taken from: (Biller et al., 2016).

One question we could ask of this model is what happens if, under a predefined fitness regime, mutation rates are varied. What genome structure evolves? Under a low mutation rate, a compact genome evolves, with closely packed genes (Figure 5.2). This strain is bacterium-like in its make-up. If the mutation rate is set to high, genomes become small, genes overlap, and there is only one start site (virus-like genome).

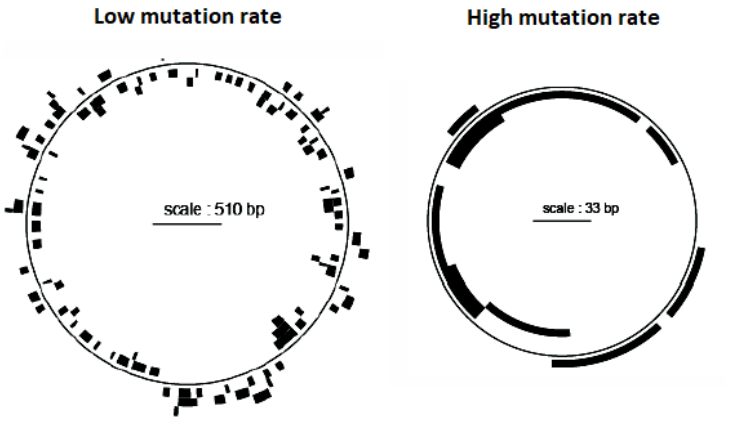


Figure 5.2: Genome structures under different mutational regimes. Left: a bacterium-like genome. Right: a virus-like genome with overlapping genes and one start site. Adapted from: [REF NEEDED].

A few times in the previous sections, we have foreshadowed the **U-shape**. In AEVOL (and actually, in

many more computational models), evolved genomes have a tendency to have a high **neutrality** and a high proportion of very deleterious mutation in their direct **mutational neighbourhood** (Beslon et al., 2017). How come? Well, we have already gone over the fact that high **neutrality** allows both innovations (percolation over a **neutral path**, with continuous novel phenotypes along the way) and robustness (fewer mutations away from the fit phenotype). That is thus clear. Why the many highly deleterious mutations (or why not slightly deleterious mutations)? This is to help purifying selection: if a negative mutation does occur, an organism is most likely to compete locally with this less fit mutant. If the selection coefficient is higher, it is easier for purifying selection to remove this mutant (because it is *much* less fit). Slightly deleterious mutants will be hard to remove because their fitness disadvantage is less clear, and could lead to mutational decline of fitness that cannot quickly be corrected. That is why evolution moves towards a **mutational neighbourhood** with the shape of a U. In a virtual cell model where the **mutational neighbourhood** is observed over time, both highly deleterious and neutral mutations increase, whereas mutations with intermediate effects decline. If you generate a deleterious mutant close by, you have a high fitness advantage over them. If we think back to the **information threshold**, we might remember that the formula for it is $ln(\text{selection coefficient})/(1-\text{quality of replication})$. Therefore, higher **selection coefficients** are better, as they move the **information threshold** (remember, the information threshold is defined as one over the selection coefficient). Thus, by making more mutations deleterious while also increasing or maintaining **neutrality**, fitness can go up. A similar pattern was noted in experimental studies in yeast and viruses. Thus, the **U-shape** is the increase of lethality and **neutrality** concomitantly, while mutations with intermediate effects are minimised.

If we step away from the current discussion a little bit, we can think about what this *U-shape* means for population models of mutations. Most population models assume a decrease of fitness proportional to the amount of mutations (Figure 5.3). However, the **U-shape** invalidates these assumptions: most mutations will be either neutral or highly deleterious, there is no nice downward fitness trend with the number of mutations, and the effects of mutations are in fact themselves, evolved. Furthermore, the specific detail of the **U-shape** (its depth, neutrality, etc.) could be an evolved property. This means that most of the models that assume the below effect of mutations on fitness are wrong.

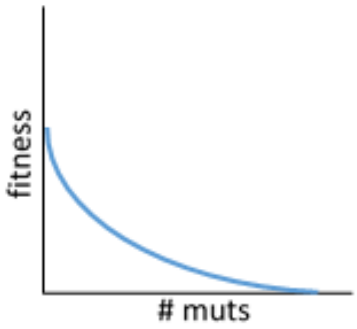


Figure 5.3: The fitness effect of mutations that population models often assume. Made by Dieter Stoker.

Evolution of the U-shape in mutator strains

Let us see how the quest for the **U-shape** manifests itself in the AEVOL system. We take evolved bacteria in this model (i.e. the entities under low mutation rates that have bacteria-like genomes), and turn them into "mutator strains" (Rutten et al., 2016). These mutators happen in real populations, where bacteria lose repair genes and therefore mutate much more often (Gross and Siegel, 1981; Giraud et al., 2001). In this case, the chance of SNPs is set to one hundred times normal levels to simulate the mutator state. There is an immediate decrease in fitness in these circumstances (Figure 5.4; the graph shoots up in the figure below, as the y-axis is the error relative to ultimate fitness, i.e. the distance from the target fitness). After some time, however, the fitness reaches pre-mutator levels, or sometimes even higher levels. What is remarkable is that genome size *increases* under this regime of higher mutation rates. Somehow, having a larger genome buffers against the effects of mutations. How is this possible?

In a mutator, lethal mutations are increased a lot, whereas neutral mutations are increased a little. By inflating the genome, the proportion of the genome that is coding decreases. Thus, SNPs are less of a problem, as are small deletions and duplications. However, large chromosomal rearrangements happen more often (remember that these depend on genome size). If a large chromosomal rearrangement hits a coding region, it is very likely to create a lethal mutant. In this way, the increase in genome size facilitates a **U-shape**. It creates more regions without coding genes, where SNPs and duplications/deletions do not matter (increasing **neutrality**), while simultaneously increasing the chance of large chromosomal mutations. Large deletions are lethal if they hit coding regions (increasing **lethality**). This effect as it were "hollows out" to the **U-shape**, which is *indirectly* beneficial for the evolved genome.

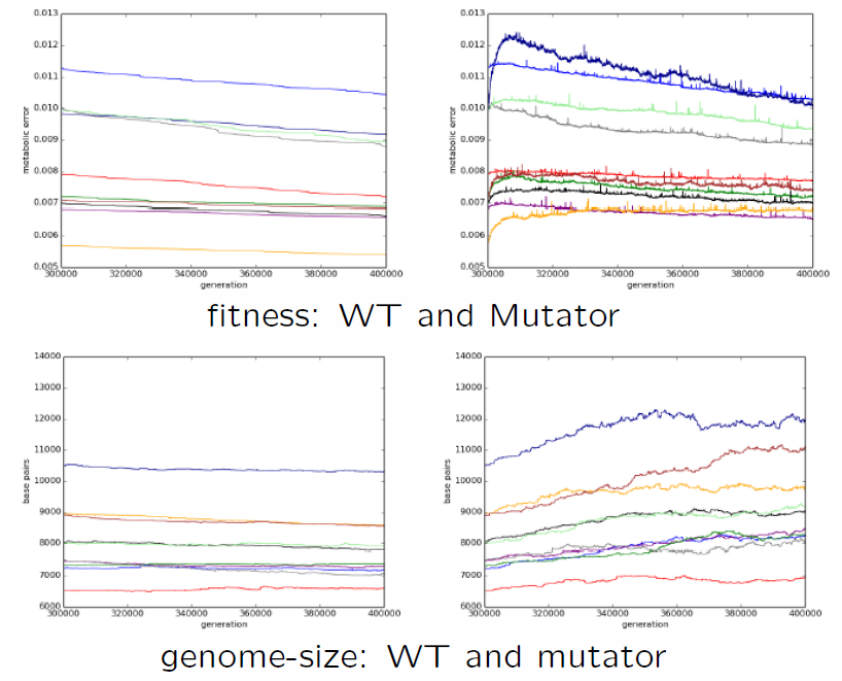


Figure 5.4: The fitness effect of mutations that population models often assume. Based on: (Rutten et al., 2016).

If a mutator strain is cropped, such that non-essential genome is reduced to the size of pre-mutator times while mutation rates are kept high, it again increases its genome size, to increase deleterious mutations, to increase its fitness (because it has a higher **selection coefficient** relative to its mutants). The system thus prefers a higher **selection coefficient** over being more neutral.

Conclusions genome evolution in AEVOL

We have now seen that genome evolution selects for a **U-shaped mutational neighbourhood**, where there is high **neutrality** and high lethality. The latter leads to a high **selection coefficient** relative to mutants, which, given you have to deal with mutants anyway, can be a good thing. Under high mutation rates, small genomes with overlapping genes were selected for, whereas under low mutation rates larger but compact genomes with an operon structure (bacteria-like) evolved. We saw that mutators, however, increased their genome size to increase the amount of deleterious mutations to increase fitness.

Evolution of gene regulatory networks

Feed-forward loops

We will now zoom in on transcription regulation networks. The results we present are qualitative. It has been noted that transcriptional regulation networks often have many feed-forward loops (FFLs) (see image).

Figure 5.5: A feed-forward loop. One of the regulatory connections found to be very common in **gene regulatory networks** (network motifs) (Alon, 2007). Made by Dieter Stoker.

It has also been observed that there is very fast evolutionary adaptation of yeast to novel environments. After a week, changes in gene expression of up to 10% were realised (Ferea et al., 1999). This involves gross chromosomal rearrangements, often the same ones in multiple environments (Dunham et al., 2002b). The question we can ask now is whether these observed structures of genomes and gene regulatory networks (GRNs) are generic, and whether they can be expected from random mutations. The short answer is yes, so let us see how it works!

We will explore this question with the Pearls on a String model. This is a more coarse-grained model than AEVOL. Populations are in space on a grid. Each grid point is occupied by a genome, and has a **GRN** derived from that genome (Figure 5.6A). There are blocks which define certain transcription factors. There are also transcription factor binding sites (TFBS), and because the genes code for TFs, there is a **gene regulatory network**. **Mutational operators** act on the genome level: there are SNPs, duplications, and deletions (Figure 5.6B). These always affect a number of blocks or a single block, whereas in AEVOL, these operators act on the single base level. TFBS can also change from binding one gene product (transcription factor) to the next, which changes the network derived from the genome. The mapping of genome to network to phenotype is evolvable. Fitness can be defined on the basis of which genes are on and which are off, and can be checked against a target imposed by an environment, which can change over time (Figure 5.6C). This could thus force the evolution of a network that can be regulated to be in two different states. The fun part about this "pearl-on-a-string"-formalism is that it is easy to come up with new "pearls". For example, in the work by Crombach and Hogeweg (2007), retrotransposons were added to study how the mutations that these transposons generate evolve to specifically result in duplications and deletions of genes that frequently needed to be either duplicated or deleted. We will now however, first discuss the evolution of the aforementioned FFLs.

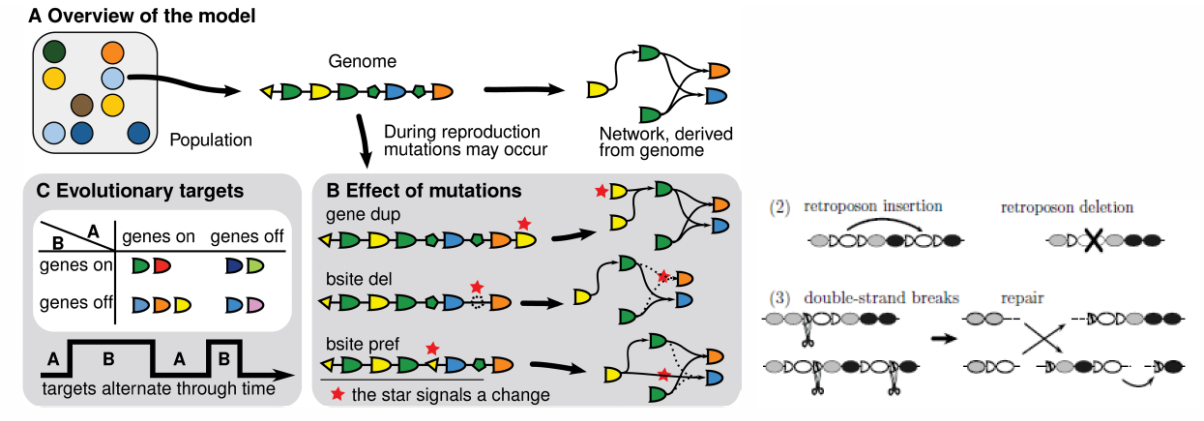


Figure 5.6: The Pearl on a String (PoaS) model. Based on: (Crombach and Hogeweg, 2008).

The observed network structure in biological transcriptional networks has a global property, namely that the **degree distribution** follows a power law. The **degree distribution** is the number of connections per node in the network. Many nodes have only one or few connections, whereas some very small number of nodes have a huge number of interactions. In biological systems, this relation can be modeled using a power law. If the distribution of degrees would be random, one would expect an exponential curve,

but there are more genes with few connections, and more genes with many connections, than expected. The local property of networks is that there is an overrepresentation of certain motifs: relations between nodes in the network that are seen very often. An example of one such motif is the FFL seen above (Figure 5.5) (Alon, 2007).

Now our tendency is often to ascribe significance to such findings: surely, if such a motif is overrepresented in **GRNs**, it must be beneficial to the system. This assumption was tested in a simplified PoaS model _without a **fitness criterion**. There is thus no direction imposed by selection at all. If these mutational dynamics are unleashed upon a random network, one winds up with an interesting result: the power law distribution and hierarchical ordering of observed **GRNs** is obtained (data not shown). After 2,000 time steps, many FFL have appeared without any selection for them in the system (Figure 5.7!)

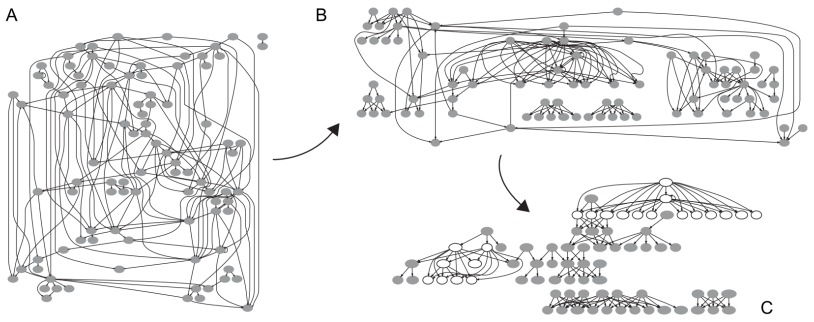


Figure 5.7: Effect of mutational dynamics on a **GRN** in the PoaS formalism after 0 (A), 1,000 (B), and 2,000 (C) time steps. Open circles denote feed-forward loops (FFL). Taken from: (Cordero and Hogeweg, 2006).

How come that these FFLs pop up? If one looks at this evolution over time, there is a sudden increase in the number of FFL (Figure 5.8). How does this happen? The reason almost sounds like cheating. Over time, hub nodes (that regulate many genes) can duplicate. All that is needed then is a mutation that connects the original hub node to its duplicate and voilà: many FFL are created. That might not seem like 'real' feed-forward loops. After all, if we take the schematic of FFLs as a guide (Figure 5.5), and compare it with this case, we see that here, X and Y are constantly the same (duplicated hub gene), while only Z changes. However, if you discount exactly these types of FFL, they are not overrepresented in **GRNs** any more. Moreover, real examples of this structure (duplicated hub nodes leading to FFLs) exist in yeast (Lee et al., 2002; Cordero and Hogeweg, 2006). Thus, the overrepresentation of feed-forward loops in biological networks is something that arises for free from mutational dynamics with multiple different **mutational operators**.

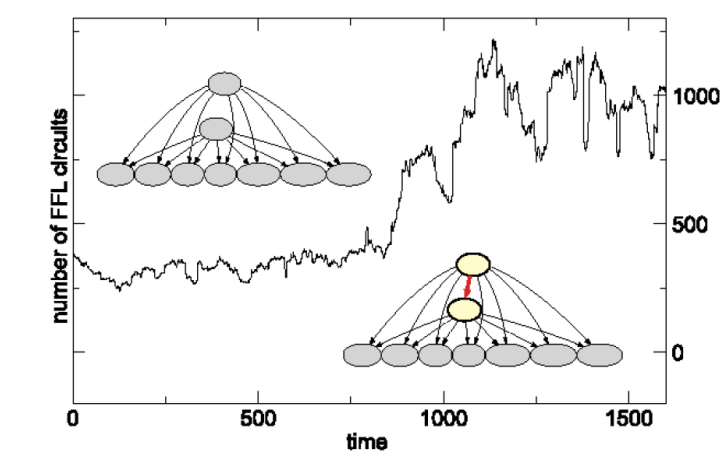


Figure 5.8: Sudden increase in the amount of FFL after some evolutionary time. Taken from: (Cordero and Hogeweg, 2006).

Evolution of evolvability

Randomisations of gene regulatory networks and the effect of mutations

We might also wonder why FFL were found as overrepresented in **GRNs** the first place. These network motifs were tested against randomised networks. How are networks randomised? The normal practice in randomisations is to keep every feature of the network the same except the feature to be tested. So, to test whether FFL were overrepresented in networks, the **degree distribution** was kept the same, while the connections between nodes was switched (Figure 5.9). This way, you have the most "fair" control group. When comparing the actual amount of FFLs in **GRNs** to these random shuffled networks, a significant difference was found. Seems like a clear cut case, but is this truly the right way to test for a difference?

Mutational dynamics would never lead to swapping of connections between nodes: such mutations do not readily occur. If you check what random mutations do, the in-degree of nodes (# of connections) is *not* kept the same. In other words: random mutations $\neq$ randomisation. If a special structure is found in the **GRN** and you check whether it is significantly different from a randomisation of the network, you still don't know whether what you observed could have been caused by random mutational processes: these processes are constrained in other ways. Thus, randomisation of a network controls for a random network, but not for a random network affected by random biological mutations.

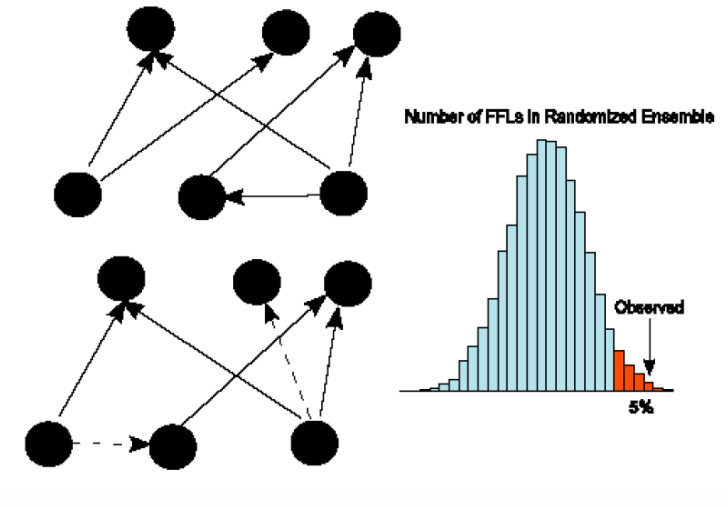


Figure 5.9: Randomisations to test for FFLs. The changes to the network that are performed would not occur in this way due to mutations, and the procedure therefore does not test for a significant difference from random networks under the influence of random mutations. Taken from: [REF NEEDED].

A mini-conclusion here is thus that *random mutations do not give randomisation, but neutrally mutate towards a certain structure*. This is an example of an **evolutionary signature**. In this case, the signature is that there are lots of FFL in evolved organisms. However, the evolved structures are not necessarily selected for, though purifying selection could, of course, act on these structures if they happen to be deleterious.

Evolution of evolvability in yeast

We will now focus on some surprising conclusions from work on yeast evolution. Efficient adaptation was observed in a short period: over 600 genes changed their expression in a period in which no more than 7 mutations were expected to occur (Ferea et al., 1999). This was not regulatory change (i.e. rapid changes in gene expression due to changes in regulation) because the ancestor was already accustomed to the medium. It had already had time to change its regulation. Hence, this was *extremely rapid mutation* with widespread changes in gene expression, even resulting in changes in the TCA cycle. How can it be that yeast so easily adapted to a new medium?

Years later, genome level changes in yeast during adaptation to a new environment were observed (Dunham et al., 2002a). Large changes in chromosome structure occurred, often at the same break points. As it turned out, there were transposon-related sequences at these break points. The evolutionary change in some ways resembled regulatory adaptation, because genes that were duplicated were nonetheless expressed more lowly, whereas some that suffered deletions were more highly expressed (the exact opposite of what one would expect). The question that faces us: *is this* ***evolved evolvability***? Has the evolutionary system yeast somehow evolved such that it can more easily evolve to fit novel environments (i.e. environments that it might have seen during its evolutionary history)?

To investigate these questions, we go back to the model that was made by Crombach and Hogeweg (2007), which was in many ways similar to the "pearl-on-a-string" model discussed so far, but now includes the aforementioned retro-transposon dynamics. These transposon dynamics cause breakpoints, whose repair can lead to gross chromosomal rearrangements. The selection criterion was that of a fluctuating external environment, where in one environment you needed two copies of certain genes to be active, while in other environments, only one should be active. This system was allowed to evolve, and the rate of adaptation to novel environments was observed over time (Figure 5.10). After thousands of generations, the rate of adaptation to a change in environment is much quicker. This is the definite proof that **evolution of evolvability** exists: evolution will evolve a genome structure such that it is easy to cope with environmental changes *that were seen before in evolutionary time*. This is **long-term information**

integration. The how of this all is that genes that needed to be duplicated and deleted often were, over time, surrounded by breakpoints (caused by transposons), so that they were easier to duplicate.

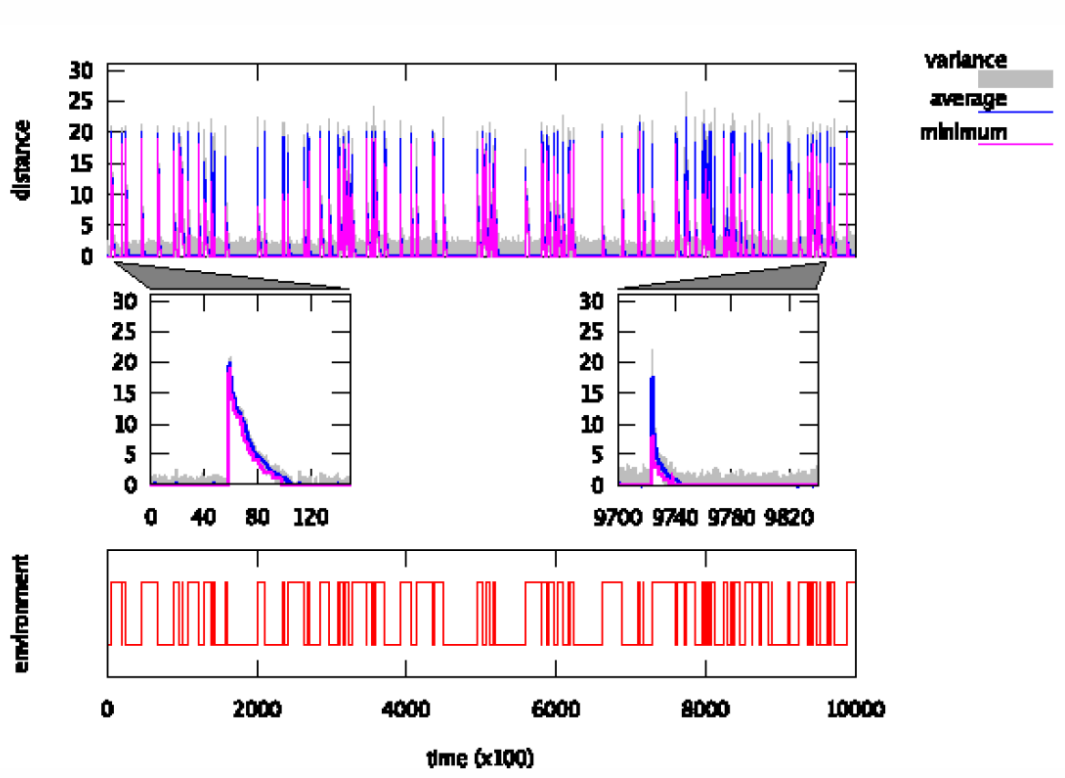


Figure 5.10: Evolution of evolvability in a simulation. Top: constant switches in the environment (bottom) cause distance from target gene expression, which is rectified by mutations. Left: time steps until adaptation to the other environment early in evolution. Right: time steps until adaptation to the other environment late in evolution. Adaptation to the other environment is much faster later in evolutionary time: the system has thus evolved to make it easier to adapt to the other environment. Based on: (Crombach and Hogeweg, 2007).

What about the *time scales* of this system: when does this genome organisation evolve? To investigate, the frequency of environment switching was varied (Figure 5.11). If the frequency of change is low, most individuals are mostly adapted to the environment in which they are the majority of the time: the distance to the required gene expression is chiefly 0 (Figure 5.11A). If the frequency of change is very high a "**bet hedging**" strategy (stochastic switching between phenotypic states (Beaumont et al., 2009)) evolves: the genomes that evolve are somewhat fit in both environments, with an average distance to the target expression of around 10 over time (Figure 5.11C). Under an intermediate regime of environment change, adaptation is not fast enough in the beginning, but over time, it is faster (Figure 5.11B). At that point, most individuals are very fit (distance close to 0). Thus, the system has evolved such that it is easier to adapt to an environment that was already seen. You also see a band of individuals that is very unfit (distance of about 20): these are individuals that made the right mutation (i.e. the mutation that is made more likely over time and switches the system to the expression needed in the other environment) at the "wrong" moment (when the environment is not changing).

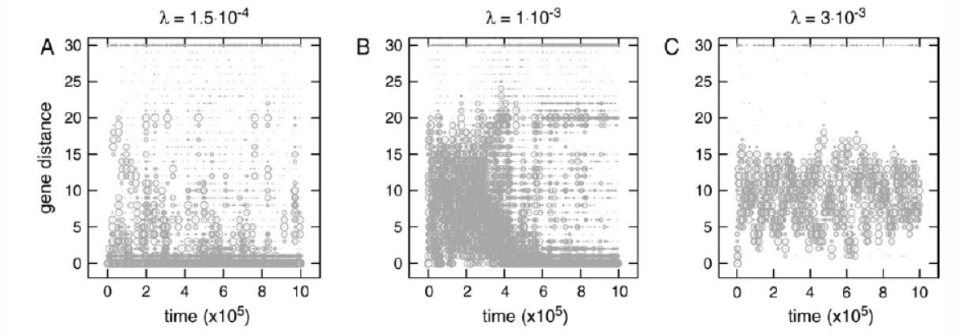


Figure 5.11: Distance from the target gene expression in the population for different frequences of environmental change. Adapted from: (Crombach and Hogeweg, 2007).

Note that there is no short-term benefit encoded into the system. There is no direct fitness benefit of having this genome organisation. The structure is there solely because it leads to better adaptation. This was a major taboo in biology. Short-term fitness benefits reign supreme in the new synthesis. Nevertheless, Darwin said that those who adapt can survive (Darwin, 1859). That is true here. Note that there is still a need for the environment to have been seen many times, and semi-regularly. However, for yeast and bacteria, we can imagine that circumstances such as starvation have been encountered many times, and the system has thus evolved to be more adaptable under such conditions. This also sheds an interesting light on the LTEE experiment by Lenski: evolution there is proceeding in an evolved system, and it will therefore be more evolvable for certain conditions (encountered over evolutionary time) than to others (that have never been seen before by *E. coli*).

Subconclusion evolution of evolvability

The process we just described and showed is called **mutational priming** or the evolution of evolvability: evolution makes the occurrence of certain mutations that can be beneficial more likely. Yeast has transposon remnants that cause breakpoints, just like in the model we just saw, and it has been observed to rapidly duplicate a whole chromosome in response to a changed environment (Dunham et al., 2002a). Older transposons are also often found in important (regulatory) regions (Crombach and Hogeweg, 2007). Though this was long a no-go in evolutionary circles, **evolution of evolvability** exists, and needs no short-term fitness benefit to persist.

Mutational priming in GRNs

We now turn to a threshold network to investigate **mutational priming** in a genome with a regulatory layer. The model is in the family of the PoaS models, so genes on linear chromosomes can serve as transcription factors and various **mutational operators** are included (Figure 5.6). The network is similar to a Boolean network, only genes become activated or not dependent on a threshold function (if more than X of the incoming nodes are on, you are on, otherwise off, or you remain the same). So to recap: the genome has TFBS and codes for TFs. Genes are turned off or on based on how many incoming connections are on. Fitness iEarly in time, switches of the environment are rare, while they happen more often later in time.

What happens in this case? We look at the frequency with which mutations have a certain effect (Figure 5.12). In a certain environment you are at the attractor (so at 0). If mutations happen, do they push the network away from the attractor in the other environment (negative values) or towards the attractor in the other environment (positive values)? The gray area shows the initial effects of mutations (i.o.w. before evolution). The blue line shows the evolved effects of mutations. After prolonged evolution, all mutational operators except one shows a larger proportion of positive effects. That is to say, these mutations are more often than expected, biased towards the network state that is required in the *other* environment (positive values). This is thus **evolution of evolvability** in a PoaS model with a regulatory layer.

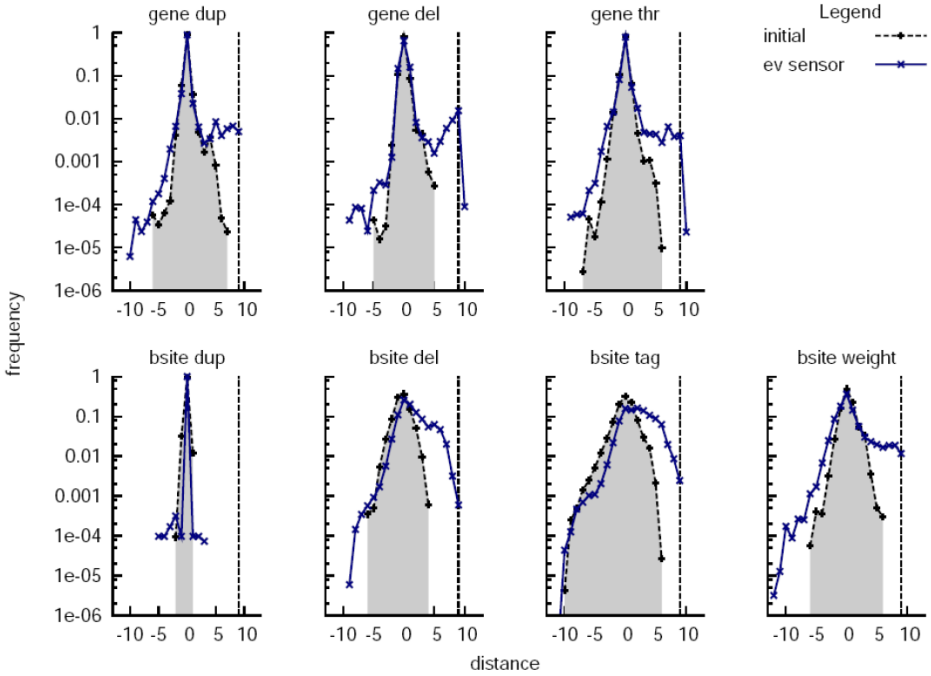


Figure 5.12: Distance from the target gene expression in the population for different frequences of environmental change. Adapted from: (Crombach and Hogeweg, 2007).

Within the network, there are still a lot of changes in which gene or which places on the genome are important for the switching behaviour. The gene that does the switching when the environments change can change over time, but the switch does remain in the genome. Thus, neutral changes to network structure do still occur. In fact, neutral mutations were by far the most abundant in this system (data not shown, see Crombach and Hogeweg (2008)). So, the system stays on a fitness "ridge" where it can easily change to the necessary target in the other environment. So, while the specific gene or position that is crucial to switching behaviour in different environments does change, the behaviour stays in the system.

In the beginning of this reader, we talked about Kauffman's investigations of Boolean networks. There, we mentioned basins of attraction that can be found in networks. Here, the network has two different attractors. Mutations need to act such that:

1. The current attractor is not the attractor anymore.
2. A new attractor is created, corresponding to the new environment.
3. It brings you into the domain of attraction of that new attractor.

Evolution manages to do all of that rather nicely.

Final yeast example of the evolution of evolvability: rapid duplication to adapt to high temperatures and Ph

Just to drive the point home fully, we shortly discuss a study from 2012 where yeast was subjected to different environments and its genome was sequenced (Yona et al., 2012). Lo and behold, for this yeast duplicated a specific chromosome (in multiple independent replicates). As it turned out, the duplication of chromosome III conferred immediate fitness advantage (better growth) under heat stress, but was detrimental under other types of stresses (Figure 5.13).

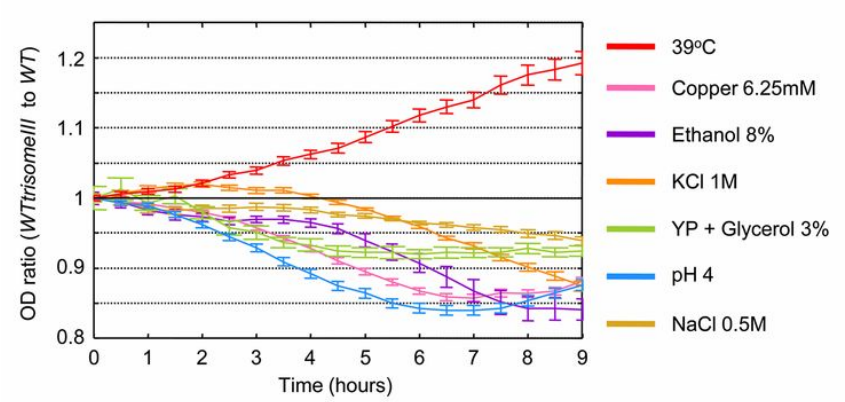


Figure 5.13: Growth rate of yeast under different stresses compared to wt yeast. Yeast with trisomy III performs considerably better under heat stress, and worse under all other stresses. Adapted from: (Yona et al., 2012).

The interesting finding here was that this chromosomal duplication was a transient evolutionary solution. It happened in all 4 replicate lines that were evolved for 450 generations in rich medium with heat stress (39 degrees Celsius). In two lineages that were evolved further, the trisomy was eliminated after 1700 or 2350 generations (Figure 5.14).

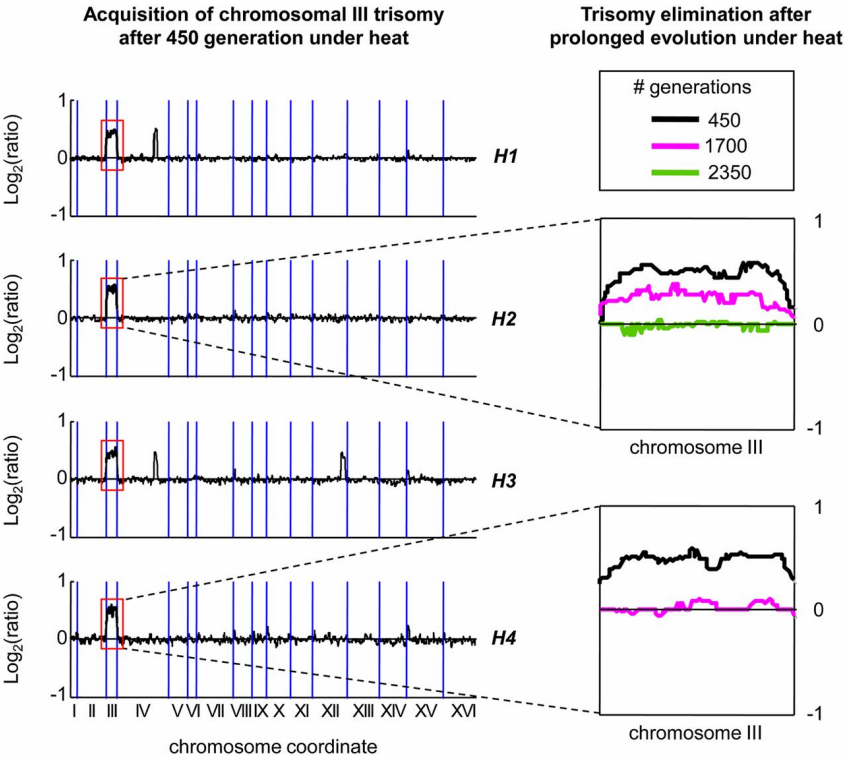


Figure 5.14: Trisomy of chromosome III happens in four replicate lines of yeast. Trisomy is eliminated in two lines that were evolved further for many generations. Taken from: (Yona et al., 2012).

This is interesting, but what is perhaps more interesting is that this evolutionary change is followed by refinement that maintains resistance to high temperatures: while the trisomy is undone in the end, there are lasting changes to gene expression. You can see that descendants of trisomic ancestors that have lost the trisonomy show improved growth under heat without the large fitness cost that having a whole

extra chromosome incurs. Interestingly, a specific group of genes from chromosome III retained high expression. In subfigure C, you can see what insertion of each of a group of lastingly upregulated genes and control genes in wildtype yeast does to temperature resistance relative to trisomy III yeast. On the left, you can see that much of the effect is mediated by HSP genes.

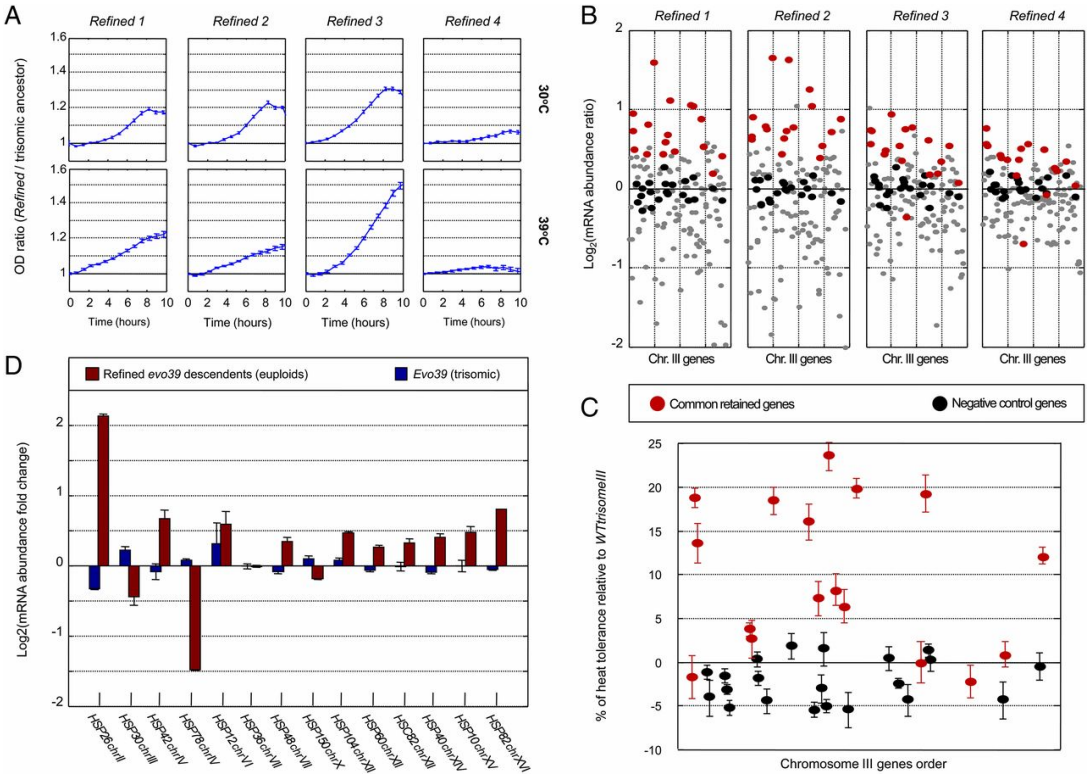


Figure 5.15: Yeast remains more resistant to high temperatures through lasting changes in gene expression after evolutionary elimination of trisomy. A: Refined mutants (without trisomy III) grow better than their ancestors at both 30 degC (top) and 39 degC (bottom). B: Despite elimination of the trisomy in Refined 1–4, a group of genes from chromosome III retained high expression levels. Dots represent log2 ratios of mRNA abundance of chromosome III genes over a diploid wild type. Genes that retain high expression (in at least three of the four refined evolutions) are marked in red, and from the majority of genes that went back to wild-type–like expression (gray dots) a control group was selected and marked in black (used in C). C: The group of genes that retain high expression levels after the elimination of trisomy confer increased heat tolerance when introduced into wild type. Each of the highly expressed genes (red) and the negative control genes (black) was inserted into the diploid wild type on a centromeric plasmid, and heat-shock tolerance was compared with that of a wt with trisomy III. D: The refined solution replacing the trisomy is characterized by changes in expression levels of most HSP genes. Log2 expression ratios over wild type are shown for all HSPs, for trisomic yeast (blue) and its descendants that eliminated the trisomy (red). Data are presented as mean and SEM. Taken from: (Yona et al., 2012).

Al in all, this is very interesting. It shows that a quick response of yeast to a novel environment (though one that yeast has undoubtedly encountered many times in its evolutionary history) is a major genomic change, that is apparently made easy due to the evolved genome structure (in this case that structure consists of chromosomes). Then, over time, gene expression is changed and refined, resulting in a yeast fitter at higher temperatures with lower detriments to fitness than a trisonomy III yeast. This is a nice, experimentally observed, example of the **evolution of evolvability**.

Conclusions evolution of genome structure

We have seen that the **mutational neighbourhood** of an evolved system tends to have a peculiar shape: the **U-shape**. Highly deleterious and (almost) neutral mutations occur more often relative to somewhat deleterious mutations. This increases long-term fitness: highly deleterious mutations allow a higher **selection coefficient** (fitness advantage) relative to mutants, and those populations with more mutations that are neutral will, on average, be more fit (less detrimental mutants, and more sampling of phenotype space via **neutral paths**). This **U-shape** also means that assumptions that are often made in population genetic models do not hold: it is not so that fitness decreases evenly with the number of mutations. Thus, when thinking about **mutational neighbourhoods**, it is extremely important to keep in mind that evolutionary systems optimise theirs to have a **U-shape**.

We have also discovered the important point that *random mutations do not lead to a random structure*. Rather, neutrally occuring mutations grant some structure for free. This is a very important result. You will encounter many a paper where certain features of a genome or network are compared to randomisations to determine whether a feature is special. Even though this can be insightful: a randomised network is not the same as a network that resulted from random mutations. Thus, one still does not know whether an observed characteristic is special relative to random mutations, i.e. whether it is truly selected for. *Random mutations do not give random results*. Thus, testing a resulting structure against randomisations does not tell the whole story.

We have just now learned that *random mutations are not random in evolved genomes*. These can ironically be called 'non-random random mutations'. These genomes have undergone evolution for many generations. Over time, evolution thus optimised the genome structure to work best for evolution. This is called **mutational priming** or the **evolution of evolvability**. In yeast and a PoaS model, we have seen that transposon dynamics are used: breakpoints border regions with genes important for survival in certain environments. Long-term evolution has thus facilitated short-term evolution without direct fitness gains. We have also seen, in the second example, that **genotype-to-phenotype mapping** evolves such that attractor switching occurs in a regulatory network: there is a blow-up of single mutations to large scale effects in the network. This mechanism, too, appears to occur in yeast.

It is important to realise that the insight regarding the **U-shape** and the **evolution of evolvability** were, for a long time, considered taboo under the new synthesis. Immediate fitness effects drived change under that doctrine, and something so mystical as the **evolution of evolvability** was considered blasphemy. Only in the last decade or two have experimental studies started to validate early modeling results, and have these concepts become more accepted (see, for example (Pigliucci, 2008; Janković and Ćirković, 2016; Nuño de la Rosa, 2017)).

Evolution of complexity: early complexity

We will now focus on another aspect of genome evolution. One that has proven to be quite counter-intuitive. We used to think that the group of multicellular animals, for instance, had many important gene innovations at the root of its tree, which allowed all the following complexity to arise. However, as research goes on, one trend stands out: the more phylogenetic analysis we do, the more it turns out that ancient organisms had huge genomes and were extremely complex (Koumandou et al., 2013). While one might initially expect that gene innovations are what drive differentiation, time and time again the opposite seems true: complex, multifunctional ancestors lose genes, and *gene loss is a major factor in adaptation*.

Another interesting observation is that whole genome duplications (WGDs) are rare, but important. Though they occur quite often (especially in plants), they are rarely fixed. Nevertheless, they are often at the root of major radiations of species, and might happen during environmental shifts (Edger and Pires, 2009; Eric Schranz et al., 2012). One might imagine that the duplication of a whole genome is a rather destructive affair, considering the regulatory upheaval and the detrimental effect on fitness of having double the amount of DNA to replicate. The fact that we see many WGDs in the phylogenetic evidence nevertheless shows that they are important to evolution and might drive success.

To better understand the phylogenetic observations, we turn once again to dynamical models (Cuypers and Hogeweg, 2012, 2014; Cuypers et al., 2017), and see what is necessary to recreate these observations.

Long-term genome structure dynamics in a minimal multilevel cell model

To investigate, a minimal or plausible model of a multilevel cell was used (Figure 5.16). It can be summarised as PoaS + metabolism: the gene network defines a metabolism. Rather than the fitness criterion of certain genes being on or off, metabolic homeostasis is the fitness criterion here. There is a varying environment, and the genome codes for pumps and enzymes that import resources and drive catabolism (energy generation) or anabolism (creating building blocks), respectively. The goal for a cell is then specifically to keep the internal concentration of a certain resource constant. The mutational operators in the system are segmental duplications and deletions, rearrangements, as well as point mutations (SNPs).

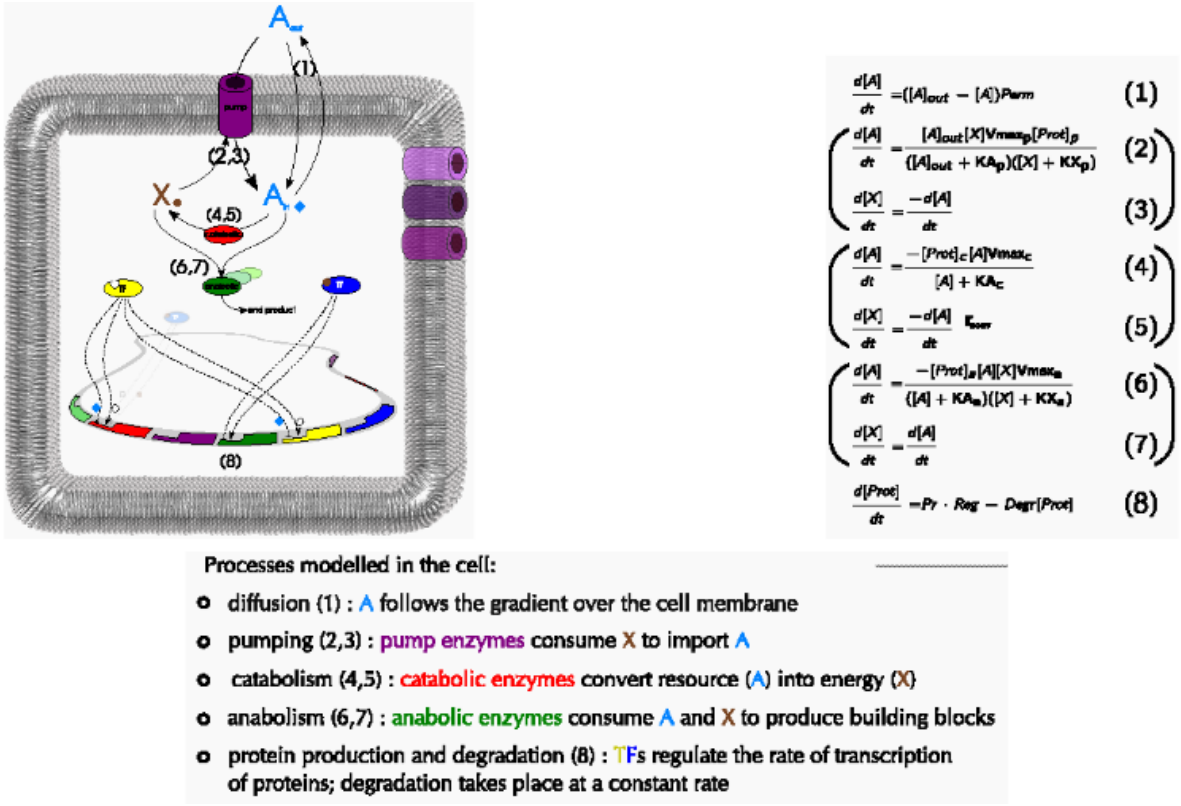


Figure 5.16: Working of the Pearl on a String model with metabolism. The genome has TFs and TFBS, and codes for enzymes and pumps. There is a fluctuating external concentration of a resource, and the **fitness criterion** is cellular homeostasis. Based on: (Cuypers and Hogeweg, 2012).

Populations are initialised with 1,000 cells and are allowed to evolve for 10,000 generations. External concentrations of the resource A fluctuate between 0.003 and 30 (4 orders of magnitude). The homeostasis criterion is that the internal concentration should be kept at 1. Initial genome size is about 10 genes. Each individual sees about one to three environments in its lifetime. The differences with previous models is thus that fitness is not expressed as gene expression itself, but as the *effect of gene expression*. This hinges on the environment and allows regulatory adaptation. Additionally, gene expression is not simply on/off, but there is a range of activation.

So, what are the typical evolutionary dynamics? From Figure 5.17 we can see that there is often an initial genome inflation followed by "streamlining" (gradual decrease in genome size). Furthermore, those genomes that underwent the initial genome inflation eventually ended up fitter! The interpretation here then is: early genome inflations are a generic pattern of organisms that eventually end up being fit!

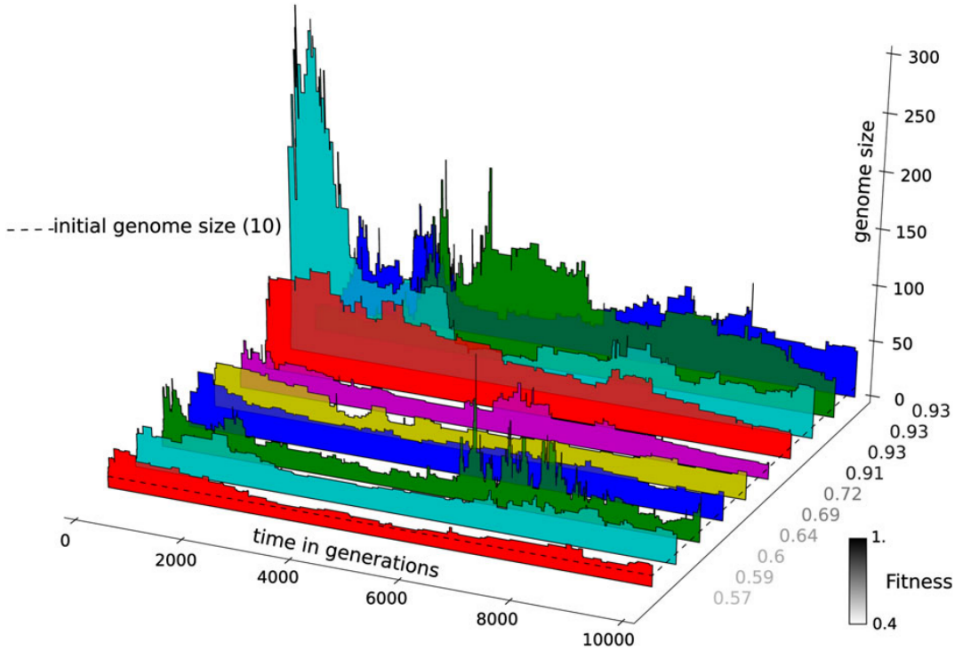


Figure 5.17: Dynamics of genome size and fitness in ten replicate simulations. Simulations are ordered from fron to back based on eventual fitness achieved. Height corresponds to genome size. There is a clear correlation between early genome inflation and eventual high fitness. Taken from: (Cuypers and Hogeweg, 2012).

We also see the aforementioned evolution of the **U-shape** in this study. If we compare the initial fitness landscape of mutants with the landscapes further in evolution, we can see that **neutrality** is increased, slightly deleterious mutations are somewhat lowered, while lethality is reduced somewhat, but still maintained at quite high levels (Figure 5.18.

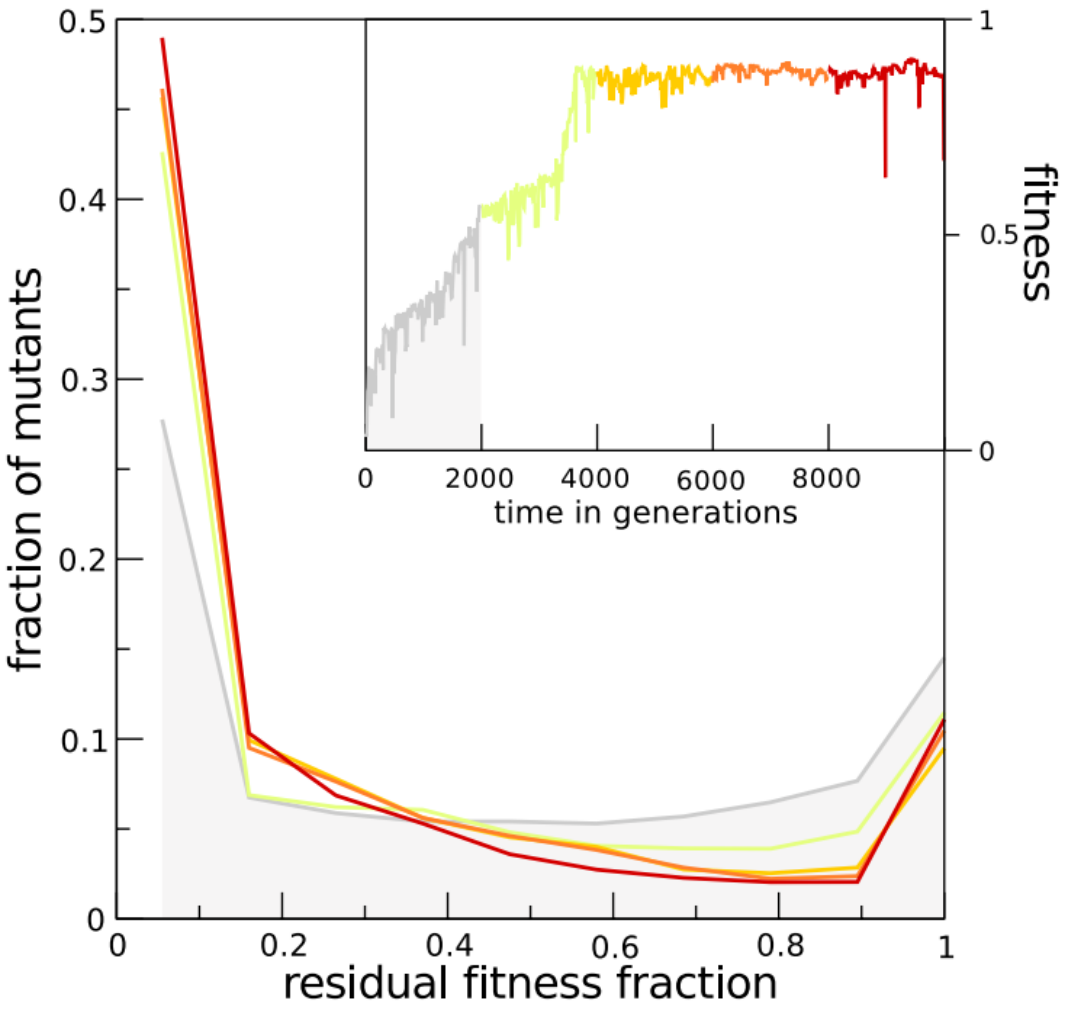


Figure 5.18: Evolving to a **U-shape** over evolutionary time. Inset: fitness of a run, where different time periods are coloured differently. The effect of mutations is found by inducing rounds of mutations in individuals from these different time periods. Taken from: (Cuypers and Hogeweg, 2012).

Studies of the effect of deletions and duplications on final fitness show that duplications are more often advantageous than deletions. Early duplications in particular are a boon to eventual fitness. This happens because other genes hitchhike along with large duplications, and they might only become functional later. Higher degrees of freedom (more genomic raw material) thus increases adaptability. Nevertheless, there is always streamlining of the genome. Why does this happen?

Over time, the network function becomes concentrated in fewer genes. Whereas fitness is first attained by many genes with small individual contributions, the end result over evolutionary time are few genes that carry a lot of the functionality. Mutations in the model have a per-gene chance of happening. Thus, by concentrating functionality in a few genes, the fraction of neutral deletions (and duplications) is increased: non-essential genes can be deleted without problems. At the same time, the fraction of lethal mutations increases, for mutation of the very important genes is a catastrophe. Streamlining counteracts the high mutational load that early genome expansion brings: if many genes together create fitness, then mutations arise more often and can be deleterious more easily. By streamlining and concentrating the fitness effect in few genes, fewer mutations happen and they are either neutral or lethal. Once again this strongly implies the evolution of the **U-shape**!

WGD and switching to a novel environment

Using the same model, we look at readaptation to a novel environment (Figure 5.19) (Cuypers and Hogeweg, 2014). Now, cells are allowed to evolve to a certain target, but after thousands of generations, the environment is changed completely. This means that extracellular concentrations of A change suddenly, without the system ever having seen these changes before. Whole genome duplications are implemented, and we ask: what is the effect of WGD on adaptability to a completely novel environment?

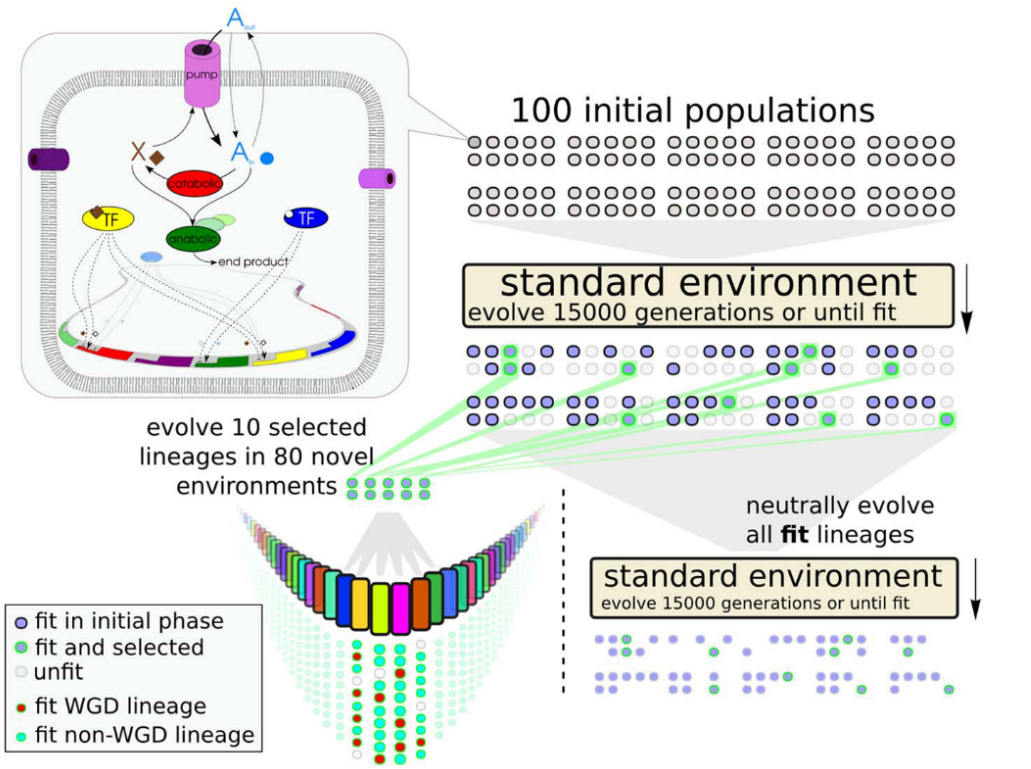


Figure 5.19: Protocol for studies of in silico adaptation to environments that were never seen before. Again, there is a resource A that can diffuse and be pumped in, there is a genome that codes for TFs and metabolic gene, and the fitness criterion is homeostasis of A within the cell. 100 populations of 1,024 cells are initialised, allowed to evolve for many generations or until fit. Then, 10 populations are sampled and subjected to 80 novel environments. As a control, all 100 populations are allowed to mutate further in the original environment. Taken from: (Cuypers and Hogeweg, 2014).

We firstly see that almost all lineages that turned out to be fit had an early WGD. Initially however, no such increase in fitness is observed. As we have discussed in an earlier chapter: fitness is a time-dependent function! Lineages without WGD barely ever reach high fitness within 15,000 generations (Figure 5.20A). In a minority of cases, WGDs happened after or before the switch of the environment, and could still help fitness in that way. WGD helps to speed up readaptation: the increase in genomic raw material allos these lineages to cope with the sudden change of environment more quickly (Figure 5.20B). Lineages with ancestral WGD are more likely to become fit eventually than those without a WGD (Figure 5.20C). WGDs were never observed in intermediately fit lineages. i.e. they happened in the beginning or right after switching, but not at other times (Figure 5.20D and E). Nevertheless, some lineages without WGD re-adapted using less than 5 mutations, so there are always exceptions to the rule (Figure 5.20F). Overall, though, WGD thus helps cope with unknown environments.

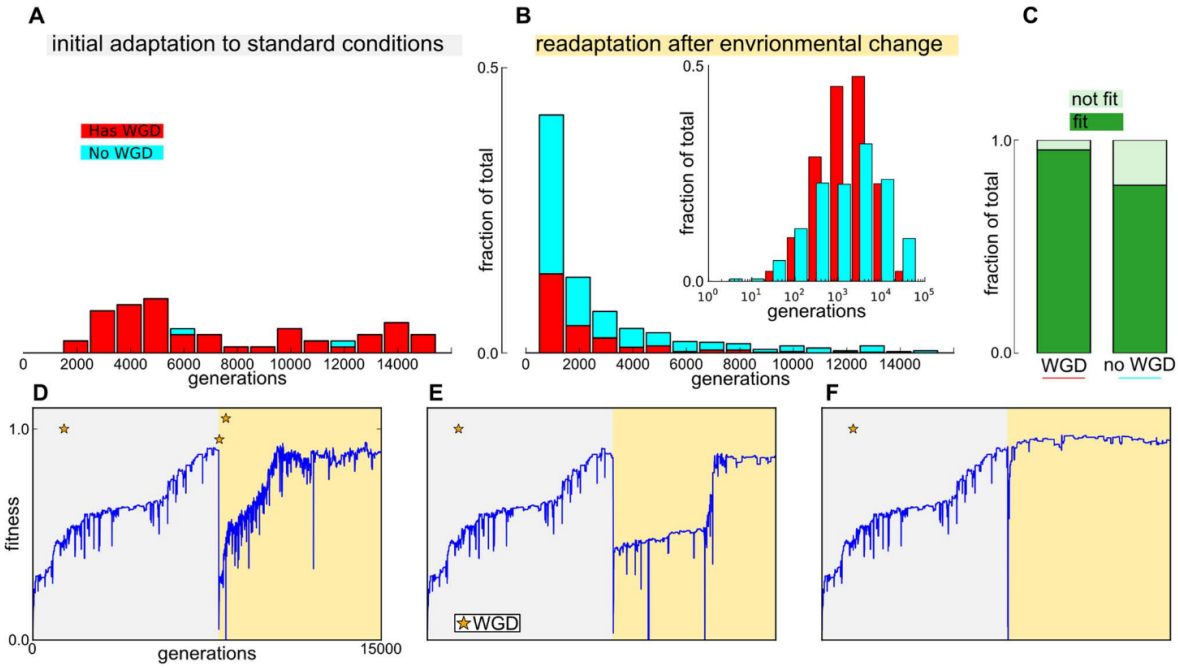


Figure 5.20: Fitness and WGD in re-adaptation. (A,B): lineages with (red) and without (cyan) WGD binned at the time of reaching high fitness (0.85) in the initial environment (A) or when readapting to the novel environment (B). Note that lineages without WGD almost never reach high fitness within 15,000 generations (A). Inset in B: distributions of time until reaching fitness on a log scale shows that lineages with WGD reach fitness earlier (larger peaks in intermediate times). C: eventual fitness in the new environment for lineages with and without an ancestral WGD. Lineages with WGD more often become fit. D-F: examples of fitness trajectories before and after environment change. Taken from: (Cuypers and Hogeweg, 2014).

If we look at what is retained after WGD, we see two trends. Firstly, in this model, TFs are preferentially retained over enzymes or pumps (data not shown). There is no sub-functionalisation. Adaptation happens by peripheral TFs. The second thing we notice is that, despite the streamlining process, absolute size of the genome remains larger in lineages that underwent WGDs (Figure 5.22). The fraction of genes retained is lower, but there is some **irremediable complexity**: after genome duplication, more genes are needed to retain fitness. A good mental model of **irremediable complexity** is one where you have a gene A that performs functions X and Y. If this gene is duplicated, one copy can lose function X while the other loses function Y. This complexity is not an increase in efficiency, and it need not be beneficial at all, but it can happen and now both copies are needed, where at first one sufficed. That is **irremediable complexity**.

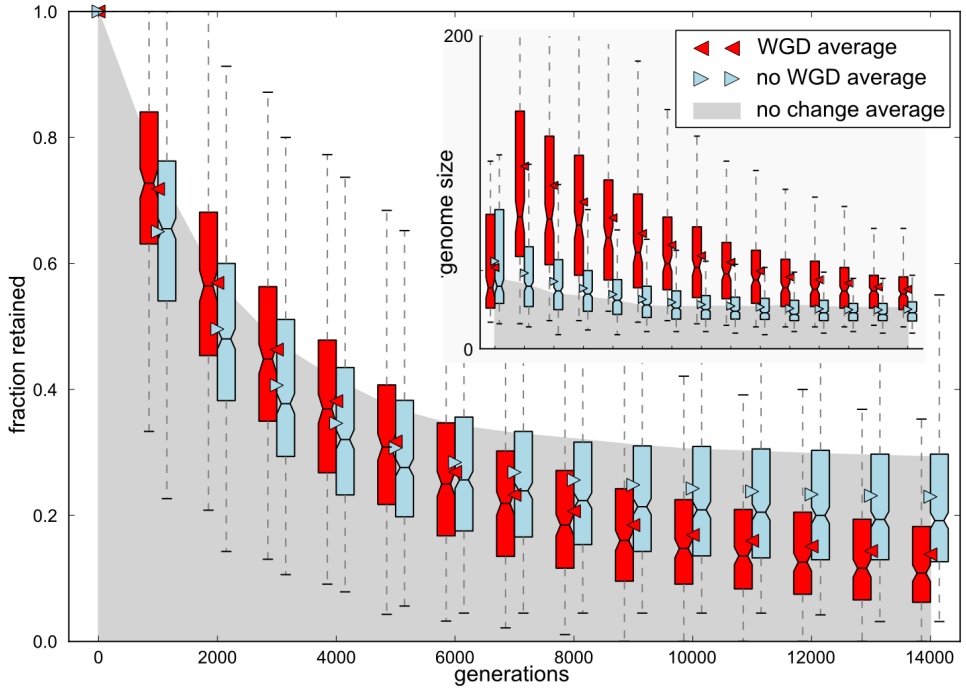


Figure 5.21: Fraction of conserved ancestral gene content through evolutionary time. The gene content of ancestors at the time of environmental switch was used as reference. At 1,000 generation intervals, the overlap in gene content of descendants with the reference was measured. All genes inherited one-to-one from the ancestral reference (not counting copies from subsequent duplication events) count towards the retained fraction of the total ancestral gene content, in WGD lineages (red), non-WGD lineages (blue), and a neutral control set where the environment was kept the same (gray). Boxes and whiskers show the 50% (box) and 75% (whiskers) ranges of the data around the median (line). Triangles and the upper edge of the shaded area show the averages of the environmental change and neutral evolutionary runs, respectively. The inset shows the distribution of genome sizes. Taken from: (Cuypers and Hogeweg, 2014).

Evolvability vs. regulation

If we now look at recurrent switching to novel environments (modelled by suddenly increasing or decreasing conversion factors, diffusion, or decay parameters), we see that ease of readaptation increases when the environment changes multiple times. Additionally, the time scale of change is important. If we look at switching between just two environments, and vary the time scale of changes (10-1,000 generations), we see two different solutions (see figure below). At shorter time scales of change, regulators evolve: these can regulate expression based on conditions. At longer time scales, evolvers evolve, who can quickly mutate to fit novel environments. [VRAAG OVER FIGUUR MET BRUIN BLAUW REGULATOR EVOLVER SOLUTIONS ONE WT].

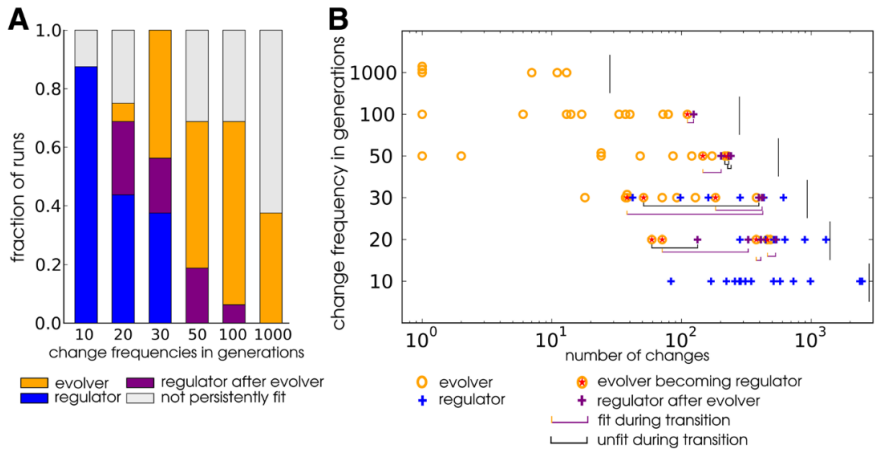


Figure 5.22: Fraction of conserved ancestral gene content through evolutionary time. The gene content of ancestors at the time of environmental switch was used as reference. At 1,000 generation intervals, the overlap in gene content of descendants with the reference was measured. All genes inherited one-to-one from the ancestral reference (not counting copies from subsequent duplication events) count towards the retained fraction of the total ancestral gene content, in WGD lineages (red), non-WGD lineages (blue), and a neutral control set where the environment was kept the same (gray). Boxes and whiskers show the 50% (box) and 75% (whiskers) ranges of the data around the median (line). Triangles and the upper edge of the shaded area show the averages of the environmental change and neutral evolutionary runs, respectively. The inset shows the distribution of genome sizes. Taken from: (Cuypers and Hogeweg, 2014).

Conclusions virtual cell modeling

Phylogenetic reconstructions often suggests that the ancestor of all modern eukaryotes actually had a huge genome. We have now seen that early genome inflations are something we should perhaps expect for evolving systems. This is probably not because those expansions were fit at that time, but rather, because those lineages surviving such a likely quite catastrophic event later found themselves with more raw material for evolution, increasing adaptability and eventual fitness. WGDs occur often, but are rarely accepted. Only if they happen early in evolution or after environmental change are they kept. There is an intricate interplay of adaptive and neutral processes. Adaptation leads to **neutrality**, and **neutrality** influences the potential for adaptation. Evolved **genotype-phenotype mappings (GPM)** maximise both **neutrality** and selection. This is the **U-shape**: mutations in evolved systems are chiefly either almost neutral or very deleterious. We have now also seen that the evolved **genotype-phenotype mapping** increases evolvability for *novel* conditions that were not seen in evolutionary history: evolved systems could adapt remarkably fast to change to an environment that had not been sensed before. We have seen that evolvability and regulation are equal alternatives to cope with fluctuating environments. However, we did see that the time scale of environmental change (relative to the speed of mutation) mattered: if environmental changes are very rapid, a regulator is better. However, evolvability to different environments is easier to evolve than being a regulator.

Of course, we now say to have derived these general principles of evolution. We say that the **U-shape**, **evolution of evolvability**, and the dynamics of genome inflation and streamlining are real and general properties of evolved systems. How can we be sure? One might state that we have been looking at artificial systems that lack some complexity, or that have strange constraints. It is perhaps best to compare these findings to model organisms: no one would argue that all organisms function exactly like yeast, but we have learned much about the cell cycle and regulatory mechanisms through studying yeast. Here, by using different models, we see that the **U-shape** occurs and that evolvability and regulation are two different solutions to the problem of changing environments. While the specifics may differ in real organisms, the overall patterns we observed here are to be expected in many organisms. We do not state that X will surely happen in the real world, but rather that these are the patterns that emerge in arbitrary and plausible evolutionary systems. We do not purport to know what exactly did happen

exactly in evolution, but we do now have an idea of what we can expect to happen by mutation and selection in evolutionary systems. Thus, these models have granted us new insights that can become **search images**: we now know what sort of patterns we might expect, and can look for them. They have broadened our horizon of what can, and will probably, happen in evolutionary systems.

Chapter 6

Ecosystem based solutions and sparse fitness evaluation

The phage example of misguided intuition

In fact, let's sprinkle in one more example. Bacterial DNA is often modified (methylated) to distinguish self from non-self by restriction-modification systems. Any invading phage DNA is not modified, and thus can be attacked. The system contains two parts: the restriction enzymes cut DNA with a certain sequence, and modification enzymes methylate DNA with that same sequence to protect it from cutting. These systems come on plasmids. The function is thus supposed to be antiviral defence. Note that this doesn't always work: sometimes, invading phage DNA is not cleaved by the restriction enzyme, and so becomes methylated at these DNA sequences itself. If the phage then multiplies, all copies are duly methylated, and the system does not work for this phage.

What would we think to be the best strategy in this system? If every RM plasmid causes you to cleave only a certain sequence, and phages have lots of different sequences, we might want to have many of these plasmids to defend against as many of these phages as possible. Sounds good, yes? Pagie and Hogeweg modeled this in a CA system (Pagie and Hogeweg, 2000). It has bacteria and phages, and there is no penalty for how many plasmids you have. Two alternative attractors exist, which you can see below. On the left, you can see our clever bacteria, which can cut lots of phages and have many plasmids. On the right, however, every bacteria just has one or two plasmids (and is thus resistant to only one or two phages).

By adding the plasmids one-by-one, Hogeweg and Pagie found out that having these clever bacteria gives clever phages, that is to say: if a phage by chance evades cutting, it gets modified (methylated) by this system. Then, that phage is immune to the restriction system, and any new bacteria it is in will propagate the modification on the phage DNA (because it comes in methylated, new copies will also be methylated). Thus, what you end up with is a system where everyone has many RM systems, but they don't do much against the phages, because when a phage is methylated, it will stay methylated (as (almost) all other bacteria also carry the RM system that first did this). While you might think that the RM systems would then fade out of the population, they do not, because there is an asymmetry of infection, called *unidirectional infectivity*: if everyone has 10 RM systems, and you too, you can be infected by 10 methylated phages. If everyone has 10 RM systems and you have 9, you can be infected by 10 methylated phages and 1 phage can also infect you if it is not methylated. This pressure is large enough to keep the systems in, but phages are relatively successful, and relatively little bacteria are present.

In the population-based model, however (which you can see occurring after some time in the lower right picture), bacteria only have one or no RM. How does this work? Phages lose their methylation over time, so if a phage gets methylated by chance, it will lose it again. If bacteria only have one or no RM system,

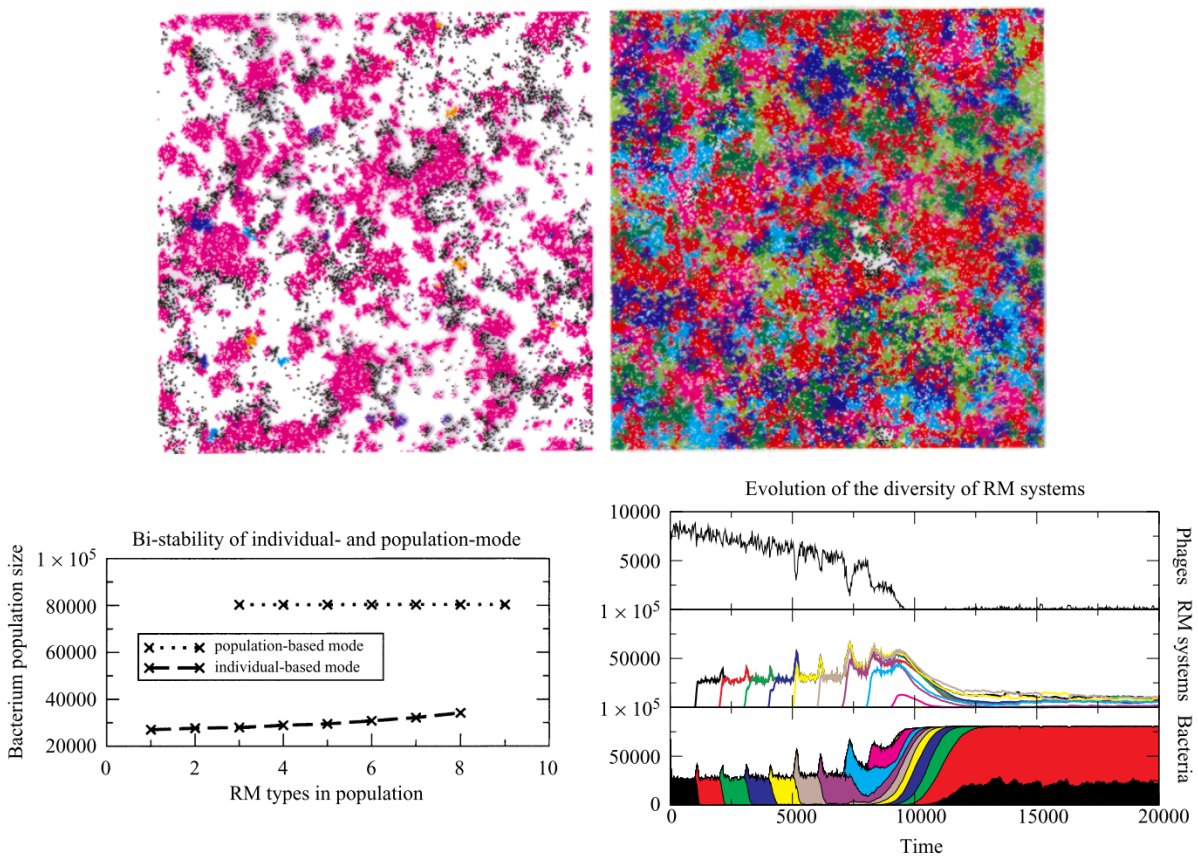


Figure 6.1: Individual vs. ecosystem based solutions (Pagie and Hogeweg, 2000)

a phage methylated in one bacterium will probably be no better off in many other bacteria, and lose its methylation. Thus, when bacteria have less RM systems, but many RM systems are present in the population, they protect each other. Many more bacteria can grow, and phages are barely alive (they mostly subsist on wildtype bacteria and exist because of influx).

So what does that teach us? What is good for whom is not *a priori* definable. You cannot say what is good or what is bad. Here, we see that having less restriction-modification systems actually works out better for the bacteria. For immediate benefit, you can say what is good or bad, but long-term benefit, and what the immediate benefits might lead to is very unclear. If we compare this to The Major Transitions: because they were theorycrafting, they *had* to choose what was good for whom. Otherwise, they would just be proclaiming baseless hypotheses. However, their assumption is a constraint and might be a problem!

time scales and fitness

We have seen in the host-parasitoid example that fitness is no defined value. This was a MAP lattice system with parasitoid wasps migrating to their hosts with a certain tendency, where many spatial patterns emerged. For an inclusive fitness over 50 generations, the best fitness value came from a *high* migration parameter. Over 300 generations, inclusive fitness was by far highest for the *low* migration parameter. Why? Because that is good behaviour in the spiral core: if you are there, you had better stay there! If you are offspring and not in the core, your migration parameter had better mutate to hunt for hosts. Thus, not everyone got the low migration parameter, but almost everyone was descended from those that did have it. Conclusion: *Fitness is a time-dependent function.* Be wary what you think of fitness of behaviours, characteristics, and traits.

On the genome level, we know that genome duplications are neutral at best, and a burden (both energetic and because of deregulation) at worst. Nevertheless, those organisms who had a genome duplication 1000

generations earlier will often be most fit later, as we have seen. That cannot easily be squared with a philosophy that focusses on immediate benefit.

We're going to spruce things up some more with another example. This time about **sparse fitness evaluation**. What is this? Well, let us think about what evolution is: it is a long-term process of information integration. Information on what works, and what doesn't, among other things. Crucially, you will not be infected by all possible viruses in your lifetime. There are many challenges that an organism might face, but it will only ever face a subsection of all challenges and obstacles that an organism of its species can face. In other words: we are only ever confronted with a small subset of an imagined fitness criterion. For the effectiveness of this approach, we'll take a look at sorting algorithms. In particular, we will look at a paper by Hillis (Hillis, 1990). He tried to evolve good algorithms for parallel sorting. Sorting involves taking, for example, numbers, and sorting them correctly from high to low or low to high. He initialised random algorithms with a diploid genome. Good algorithms were allowed to reproduce, which introduced mutations and cross-over. He wanted a fast algorithm, but selected for the number of problems that an algorithm could correctly sort. Interestingly, this gave fast sorters as a side-effect (or epiphenomenon).

Interestingly, what he found was that this approach worked best if algorithms and sorting problems were in a CA field, such that there was spatial locality of algorithms and the problems to solve. In that way, algorithms are only presented a subset of problems per generation: *sparse fitness evaluation*. More interestingly, if the problems were static, the problem of local optima emerged: many algorithms evolved that could sort most problems easily, but they were still far from perfect. In this case, the fitness criterion gave little information about who was actually the best algorithm: with a static problem set that almost anyone can solve, there is little for selection to go by. If the problems were allowed to co-evolve with the sorters (where the fitness criterion for the problems was to not be solved, such that only problems that the current algorithms found difficult would thrive), however, an algorithm that was only marginally worse than the best known one emerged (61 computations versus 60). This is because the small, co-evolving subset of problems provided a more easily navigable fitness landscape. Thus, local red queen dynamics between algorithm and problem set, with *sparse fitness evaluation* proved a great optimisation procedure. Hillis concludes that 'It is ironic, but perhaps not surprising, that our attempts to improve simulated evolution as an optimization procedure continue to take us closer to real biological systems'. It is a testament to the effectiveness of evolution that evolving (parallel sorting) algorithms is still an important field today (Mora et al., 2015).

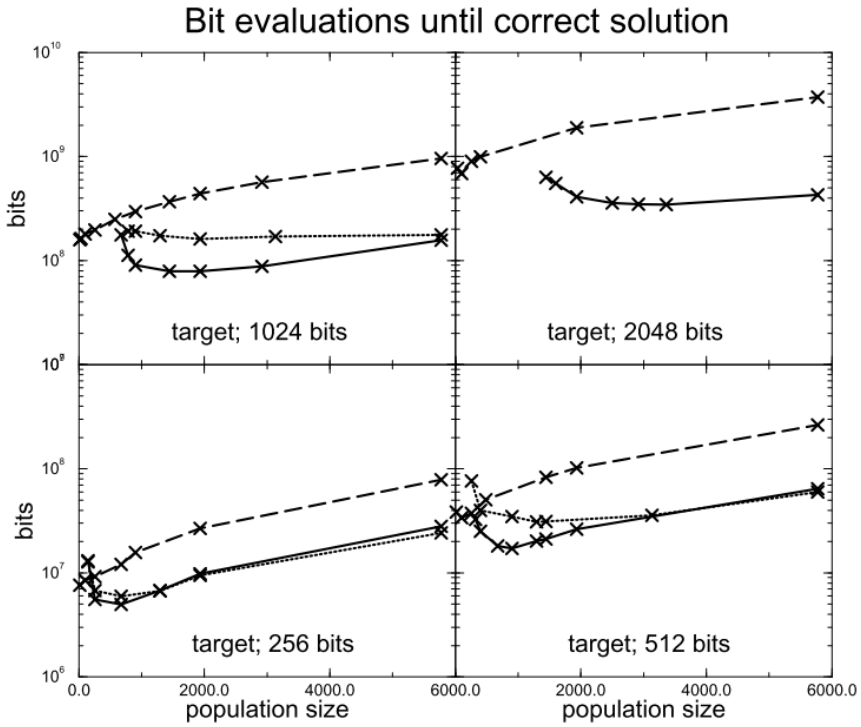
The effectiveness of *sparse fitness evaluation* was also shown by Pagie and Hogeweg (Pagie and Hogeweg, 1997). They looked at fitting of a mathematical function by algorithms, and specifically at how sparse and total fitness evaluation differed in results, generalisability of the solution, and mutational robustness. They did this by populating a CA with solutions and problems from a complete problem set, and co-evolving the problems from the problem set that were shown with the solutions. What they found was that even in the simplest case, where evolution was simply aimed at matching a certain preset bit string (string of 1 and 0, of variable lengths; this corresponds to a Mt. Fuji landscape), sparse fitness evaluation performed better (see image below).

For the actual mathematical function (the mathematical function to be approximated was $\frac{1}{1+X^{-4}} + \frac{1}{1+Y^{-4}}$), fitness evaluation on the full problem set did not achieve good solutions at all, whereas sparse evaluation quickly got to almost perfect solutions (see below).

These solutions were also more general, such that evaluation on *unseen problems* worked out well for the approximations evolved under sparse fitness evaluation, but were a disaster for those under static evaluation. Thus, we see here another example of *long-term information integration*: solutions (phenotypes) are only scored on certain problems in their lifetime (*sparse fitness evaluation*, but manage to integrate these separate challenges into a solution that is correct, quickly achieved, and generalisable (which you could equate with being more robust to changes in the environment).

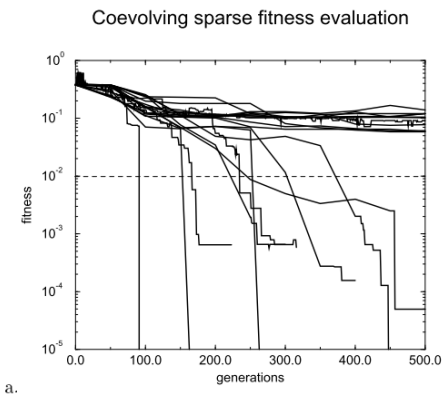
Giving up self-sufficiency, or exploiting opportunities?

In the view of The Major Transitions, complexity arose by giving up self-sufficiency. That sounds negative, or as something that one needs to be coaxed into. However, we could recast this as exploiting new

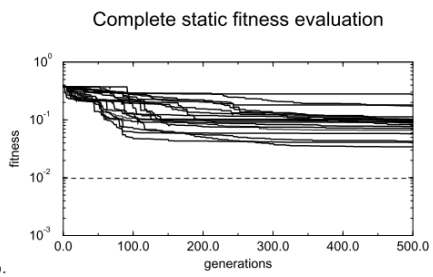


Coevolving (solid line), static (dashed line) and random (dotted line) fitness evaluation of a simple linear problem. Optimization time in terms of bit evaluations. Data are averaged over 5 runs.

Figure 6.2: Hillis sorter (coevolution vs. static) Hillis (1990)



a.



b.

Fitness curves of the best-of-generation solution for coevolving a) and complete static problem evaluation b). Fitness is based on the 'complete' problem set that consists of 26×26 problems. The fitness curves that drop below 10^{-5} go to values between10^{-15} and 10^{-17}. The horizontal dotted lines give the value of the hit criterium (see text).

Figure 6.3: Sparse fitness evaluation outperformes complete fitness evaluation.

opportunities. For example, there is an emergence of non-self-sustaining parasites or cheaters in systems with a single replicators. On the population level, the system is still self-sufficient, and the pressure exerted by the cheaters leads to more efficient self-replication. Local interactions lead to spatial patterns, with higher level entities that can enslave the lower-level entities, such that their evolutionary fate is not in their own hands. This creates evolutionary interdependence. For the spirals in the hypercycle model, how fast spirals rotate affect what is happening much more than what an individual replicator is doing. A replicator cannot decide what is good for it anymore. We have seen that RNA in protocells prevents evolutionary extinction with the stochastic corrector. This is an opportunity, and might have come about due to a concentration of nutrients within cells. We have seen that DNA can evolve in a RNA world by exploiting the weakness of having both catalysis and information storage in one molecule, offering a way out of that conundrum. Thus, exploitation of new opportunities is something that happens at many levels.

Division of labour

The same is true of division of labour: this is a phenomenon that we can find on many different levels, and in very different systems. We have seen non-heritable phenotypic differentiation: TODO-based behaviour and regulation in bumblebee colonies is regulated by dominance interactions and what is encountered. We have also seen a division of labour in quasispecies, where it lay in the mutational neighbourhood: at high mutation rates, what evolved was not the sequence that had optimal replication, but rather one that filled its immediate mutational neighbourhood with many helpers, which acted like worker bees, aiding its replication. We could call this mutation-decoded division of labour. In the host-parasitoid example, and in the hypercycle example, we have seen that there is a location-decoded division of labour: the spiral centre is similar to a germline, it generates all offspring in the system in the long-term. The other replicators might be seen as the soma. In symmetry breaking we find another division of labour: one strand becomes something of a proto-genome, the other a much better catalyst. This evolved division of labour has as side-effects that the system can survive at much higher volumes (protocell) or larger diffusion rates (spatial system).

We could also think of ecosystem-based division of labour or problem solving. In fact, you might have guessed that Paulien (and Folkert de Boer) did think of that, and we'll now dive in to a small example (de Boer and Hogeweg, 2012). In an ecosystem, organisms have to cope with their local environment (solving problems) and they evolve interactions via resources (problems). But how do these interactions in an ecosystem evolve? For that, they looked at an artificial ecosystem. The task to optimise is once again a mathematical function that needs to be approximated as well as possible. This is an extension of the co-evolution of problems and solutions we saw above. Now, however, there are predators and scavengers (two solutions), that together hunt prey (combined should approximate the target function for a given set of coordinates).

So how does that work? There are three planes. One with predators, one with prey, and one with scavengers. Importantly, predators and scavengers do not see each other. Instead, predators achieve fitness by eating as much of the prey as possible (approximating the true value of the mathematical function for a certain set of coordinates as well as possible). What they do not solve are leftovers, and the scavengers see only these leftovers, and try to best approximate these (see image below).

What happens? Well, solvers evolve that solve only one part of the problem (the Y part or the X part). Scavengers take care of the rest, and the predators and scavengers colocalise. Largely, the difficult problem can be completely solved by the predators and scavengers together (see image below). Thus, the difficult problem is automatically (or automagically) decomposed into easier sub-problems. A division of labour, if you will.

Thus, ecosystem-based problem solving happens. This might sound similar to what Eigen proposed for the information threshold problem, but his solution was unstable, and had a strange topology. This work shows, however, that it is possible. Other work on the topic by de Boer and Hogeweg has shown that finding a solution needs more information than maintaining a solution, and that ecosystem-based solutions are resilient to high mutation rates, whereas individual-based solutions are less so (De Boer and Hogeweg, 2010).

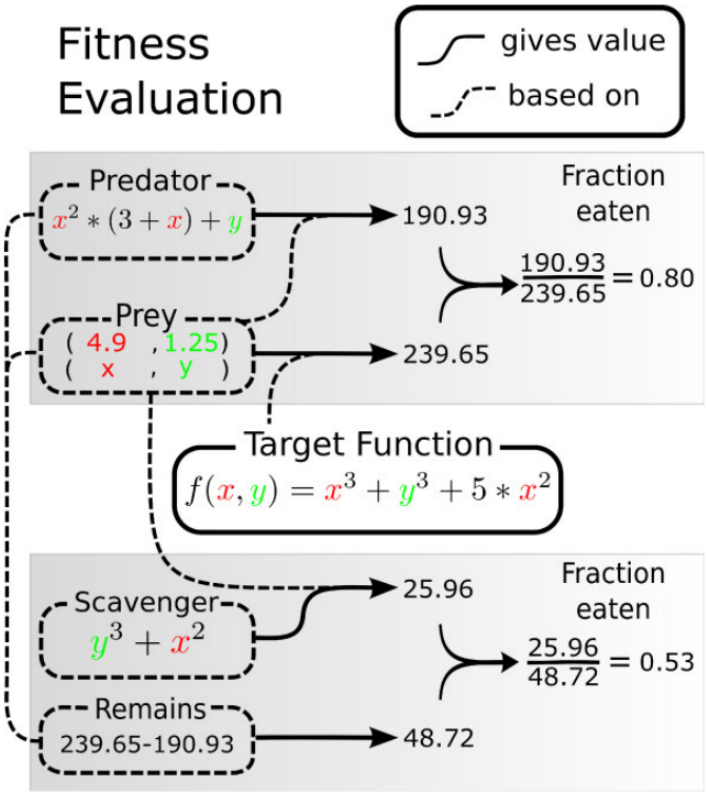


Figure 6.4: Evolution of problem solving in a predator-prey model.

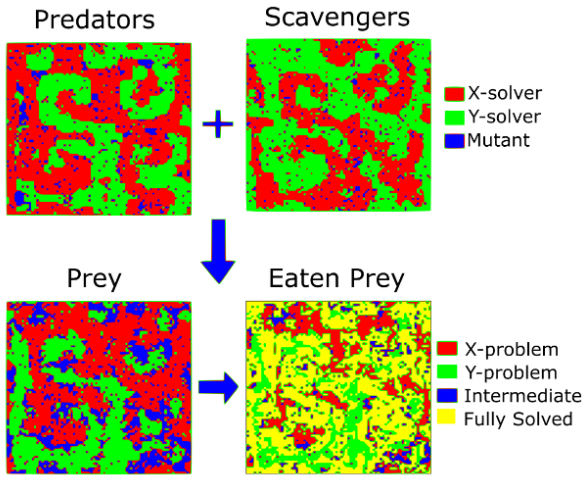


Figure 6.5: Ecosystem based solutions for solving problems works at high mutation rates. The problem is decomposed by multiple species on both "trophic" layers (predators and scavengers)

Chapter 7

Supervised and non-supervised modeling strategies to understand specific phenomena

Up until now, we have looked at non-supervised multi-level evolution modeling. That is to say: we have taken basic (evolutionary) assumptions (mutations and selection occur) and looked at what happens if we add space, (evolvable) **genotype-phenotype mapping** (using RNA, Pearl on a String genomes, **Gene Regulatory Networks**, and environmental feedback), and different **mutational operators**. Understanding what happens in such models often proves difficult, because there are many **observables** (just as in real life). Finding out what the quasispecies with the functional neighbourhood was doing, for instance, took at least a year, because we need to look at such systems as if we are studying a real life system: slowly gleaning understanding from myriad observations of what is going on (Colizzi and Hogeweg, 2014).

From now on, we will instead focus on some (experimentally) observed phenomena and look at various supervised modeling strategies to understand these specific phenomena. Some models will not involve evolution at all, other models will incorporate it as a tool, rather than as a phenomenon whose dynamics are to be studied in its own right. We will first turn towards the transcription/replication conflict.

Genome organisation and the conflict between transcription and translation

The yeast genome has long tandem repeats of ribosomal RNA genes, often way more than 100! It has been experimentally shown that up to 20% of them can be knocked out without fitness effects (Colizzi and Hogeweg, 2016). Nevertheless, in a short time period, the repeats recover their length. Why should this happen? After modelling was brought to bear upon the issue, the answer is: this is regulation of mutation.

The step-by-step explanation: transcriptional loads induce mutations due to a replication/transcription conflict (García-Muse and Aguilera, 2016). When the translational and transcriptional machinery of the cell bump into each other, this causes increased chance of mutations. Within the tandem repeats of rRNA genes, there are so-called replication fork barriers (RFB). These are areas on the genome where proteins can bind that stop the unbinding of base pairs necessary for DNA replication (Brewer and Fangman, 1988). These barriers cause replication to wait until transcription is done, and diverts mutations that might become SNPs to instead become duplications or deletions. The TOR pathway in yeast, which is involved in activating rRNA transcription, also biases mutations towards duplications.

The model

How did we learn the above explanation? The model worked with a population whose size was kept constant. Cells can pump in a resource, which costs them energy. Cells need an enzyme (the pump) to

do this. When amino acid concentrations in the cells rise, this results in more transcription. There are a few types of genes on the genome (enzymes, rRNA, ribosomal proteins, household genes). All genes are transcribed to mRNA. Mutation at the gene level (a SNP) causes an mRNA to become inactive. Housekeeping genes need to be produced in certain concentrations (this is thus an **external fitness criterion**). There is a background mutation chance, and a conflict between transcription and translation imposes an additional chance of mutation (note: if this is left out, the model only holds at very high mutational rates). The additional mutation chance on regions with conflict between transcription and translation is evolvable: the chance of mutation is constant, but the type of mutation (**mutational operator**) can be changed (SNPs, deletions or duplications) by evolution.

Model results

We first look at the case where there is only the background mutation rate (i.e. transcription/translation conflict is left out). In such a situation, the genome inflates, and interdivision time is high (Figure 7.1top left and bottom left, respectively). There are many inactive genes (Figure 7.1top right). If the extra evolvable mutation rate is added, the problem of inactive genes is alleviated: genome size shrinks, inactive genes are few and far between, and the interdivision time is markedly reduced. The inactivating mutations (SNPs) are almost nonexistant, whereas duplications and deletions do happen, which is selected for by the system (Figure 7.1bottom right). Here, adding mutations of a certain type alleviates genome deterioration: allowing more mutations resolves the problem of increasing interdivision time and a large proportion of inactive genes.

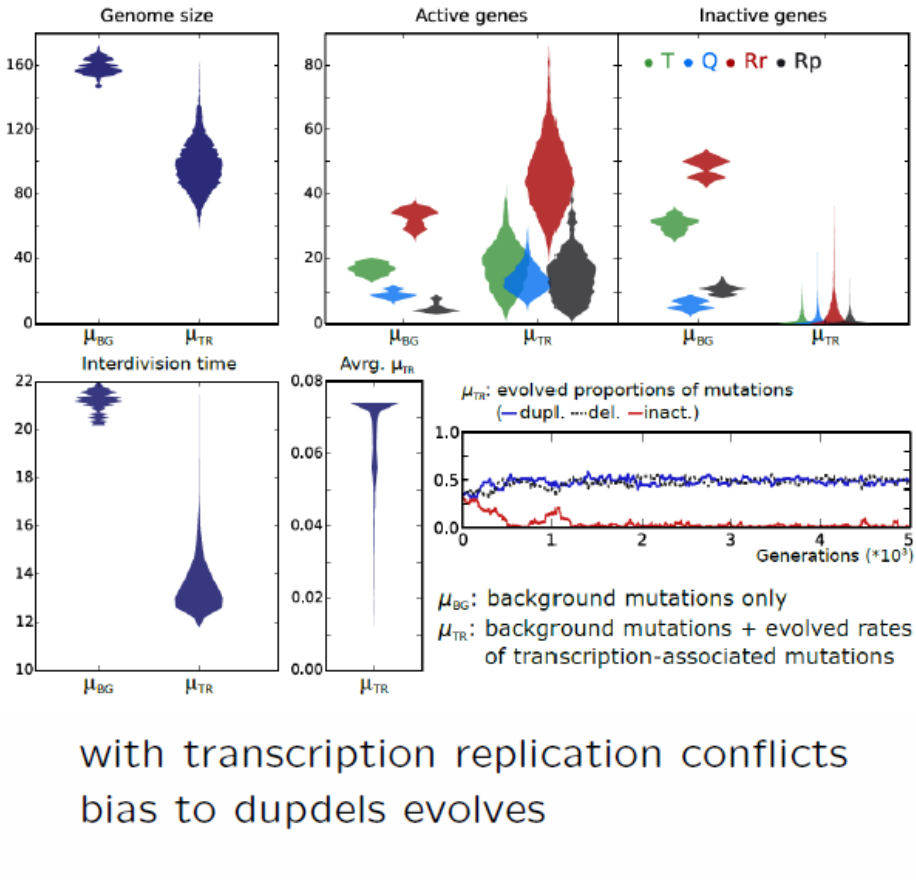


Figure 7.1: Yeast transcription and translation conflict. Situations with only background mutations (μ_{BG}) or additional evolved mutational load because of replication/transcription conflict (μ_{TR}). Left: genome size and interdivision time. Top right: amount of active and inactive genes (genes become inactive due to SNPs). Bottom right: evolution of the **mutational operator** that is used for the additional mutation rate due to transcription/replication conflict. The system clearly prefers duplications and deletions, and this ameliorates the problem of inactive genes. Taken from work that is partially unpublished and (Colizzi and Hogeweg, 2016).

Intermezzo: comparison with mutator strains *E. coli*

We have seen a similar type of behaviour in the mutator screen of *E. coli*. There, if an individual was turned into a mutator, it expanded the non-coding parts of its genome. This made the **mutational neighbourhood** into a **U-shape**. First, there was a loss of fitness, however, following that, there was less deterioration because of slightly deleterious mutations. By increasing the non-coding parts of the genome and decreasing the coding part of the genome, the chance for duplications, deletions and LCRs increased (remember, SNPs in non-coding parts of the genome didn't matter). Because selection on deleterious mutations was stronger (the compressed coding part causes any mutations therein to be (almost) lethal), a strong **U-shape** emerges. [In a similar vein, the system here chooses duplication and deletions over inactivating SNPs, increasing lethality and neutrality (duplication is neutral, deletion is more often lethal), and decreasing the amount of inactive genes, similar to what yeast does with TOR and replication fork barriers (Colizzi and Hogeweg, 2016).

There are, however, important differences. In the *E. coli* case, duplication and deletion probability could only be increased by evolving a larger genome. Here, there is a parameter that governs what type of mutations the added mutation rate will inflict. However, similar increase of dupdels over SNPs is seen, so the two findings are qualitatively linked. If given the chance, a system will choose more lethality and more **neutrality**.

Deeper look into model results

Why does the transcription-translation problem arise at all? There is a short-term pressure to obtain active genes (you want housekeeping genes at certain levels) by duplication of these genes. However, this short-term pressure leads to an increase in genome size. This, in turn, increases the mutational load. If this mutational load becomes too large, inactive genes are likely to result (due to SNPs), leading to **mutational breakdown** (i.e. decrease in fitness due to rampant mutation). We see that the ratio of active genes over inactive genes is often lower in the current population (dark red) than for the ancestors (blue) (Figure 7.2). Thus, mutational breakdown is frequent. On the right, we can see that there is no fitness effect of removing rRNA genes, just like the finding in yeast that prompted this model.

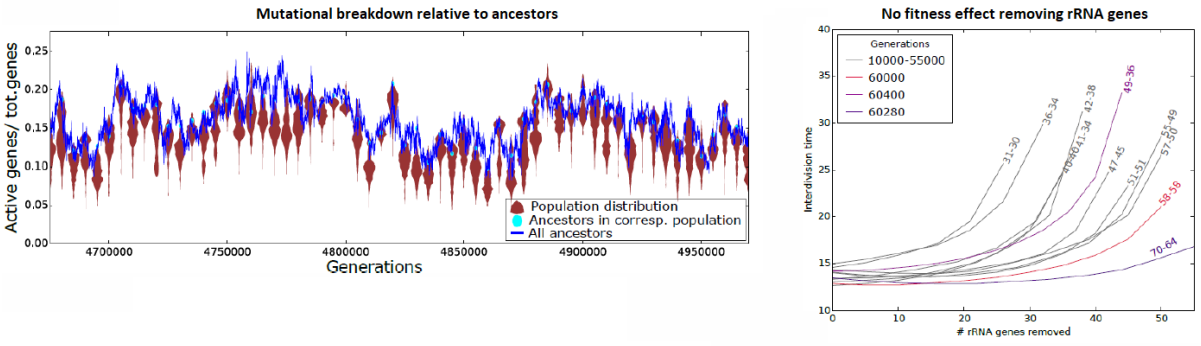


Figure 7.2: Transcription and translation conflict hinges on mutational breakdown and loss of genes gives no loss of fitness. Left: proportion of active genes over total genes over evolutionary time. Plotted are the distribution in the current population (dark red) and the ancestral population (blue). There is often a loss of active genes due to **mutational breakdown**. Right: removing up to 20% rRNA genes in the model has no fitness effect, just like in yeast. Taken from work that is partially unpublished and (Colizzi and Hogeweg, 2016).

Bottom line

An elaborate system evolved for long-term evolutionary purposes. This is thus a nice example of evo-evo, the **evolution of evolvability**. By choosing duplications and deletions over SNPs, the system has less of a burden of inactive genes. Real yeast might have many rRNA genes with replication fork barriers for the same reason, to alleviate mutational pressures caused by the transcription/replication conflict. Different kinds of mutations can have many effects on a system, and evolution might prefer one over the other. Dobzhansky was thus right: nothing makes sense except in the light of evolution (why have many rRNA genes if losing up to 20% shows no fitness effect? Long-term evolutionary optimisation). It was not predefined in this model what good was. The only **fitness criterion** was that housekeeping genes were needed in certain concentrations, and that growth was needed. These are very minor assumptions: we never explicitly forced a criterion that more rRNA genes was better for the system. That is what a **mini-model** might do (more on that in the next section), but we allowed the system freedom to evolve a solution. In this case, we see that the system wishes more duplications and deletions, increasing the amount of rRNA genes, and alleviating the conflict.

Intermezzo: experimental and modeling strategies

We will now look at how multi-level evolutionary modeling can correct errors in thinking that might arise due to other modeling and even experimental approaches. Experiments use controlled conditions. It is assumed that, in this way, the internal state of different replicates is the same, or as similar as possible (due to few fluctuations in environment). This limits the degrees of freedom of a system (organism). **Mini-models** allow one to study the parameter space and choose good parameters based on the outcome of the modeling. Basically, a **mini-model** is heavily based on experimental outcomes and knowledge (such as measured kinetic rates of reactions) and you vary parameters to see what this does to the system. **Detailed models** use many measured or estimated (reasonable, is often claimed) parameters to very stringently model an experimental system. Whereas a **mini-model** would take core reactions,

a **detailed model** would take all known reactions, estimate all parameters that are unknown, and thus try to model all or most intricacies of a system. Note that such systems suffer from the **parameter curse**: they need huge amounts of parameters to work, and these are often very hard to find. Hence the reasonable parameters. However, a cautionary tale exists of ecologists who, with *reasonable parameters*, made an ecological model that predicted the need for a population of several monsters in Loch Ness (about 1-156, the researchers reckoned, afterwards thanking their colleague for drawing their attention to this problem, because they had been unaware that monsters were a problem) (Sheldon and Kerr, 1972). We do not really believe, currently, that such a predator exists. Thus, reasonable parameters can be made to predict almost anything, and one should always be cautious in using and trusting such parameters. Finally, **minimal evolutionary optimisation models** ask what certain properties could be good for in a given system, but often include only a very minimal representation of the system. **Bet hedging** models are an example.

Is the lac-operon a bistable switch?

The lac operon is the prototype of gene regulation. It is a bacterial set of genes that regulates the uptake of lactose. LacY makes a lactose pump and LacZ makes beta-galactosidase, which breaks down lactose. Glucose reduces the expression of the operon, whereas a shortage of energy and sensing of allolactose stimulate the expression of the genes in the operon (Ozbudak et al., 2004) (Figure 7.3). Classical **mini-models**, based on experimental data, showed that there was bistability and hysteresis in this system: for a certain concentration of allolactose, the operon could be either active or inactive, depending on whether allolactose was previously high and was now decreasing, or whether it was previously low and was now increasing. Thus, for a certain range of allolactose, there were two possible states of the system. This finding from models based on measured parameters was later experimentally verified (Ozbudak et al., 2004). Models and experiments agree, so that is that. Well, not really, as we shall see.

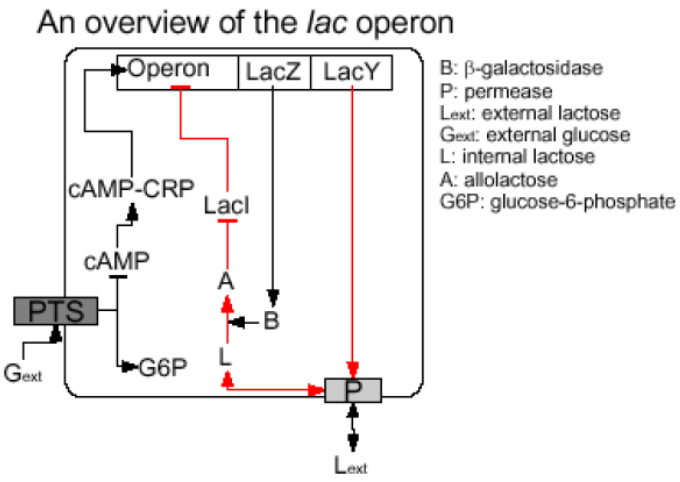


Figure 7.3: Workings of the *E. coli* Lac operon. Taken from: [REF NEEDED].

Very complex models of lac operon dynamics had been made. Paulien and von Hoek adapted such a complex model (Wong et al., 1997), and allowed the parameters governing promoter function to evolve (Van Hoek and Hogeweg, 2006). Note that there was again a huge **parameter curse**. Worse, a literature search of experimental parameters (which are often treated as representing a ground truth) showed that they could vary by three orders of magnitude. What parameter is one then to pick? Of course, this variation is not wholly unexpected: *the parameter* for *E. coli* kinetics does not exist, it is a metaphysical construct. There will always be differences between strains and conditions. Thus, even experimental parameters are uncertain.

Nevertheless, the model was parameterised using the best of our knowledge. This was the most-studied regulatory system, and it was often considered an AND-gate: it was ON if lactose AND NOT glucose,

otherwise it was off. However, recent promoter measurements seemed to indicate a more graded response of the promoter to metabolites (Figure 7.4). Still, it was maintained that in real *E. coli*, bistability is the norm.

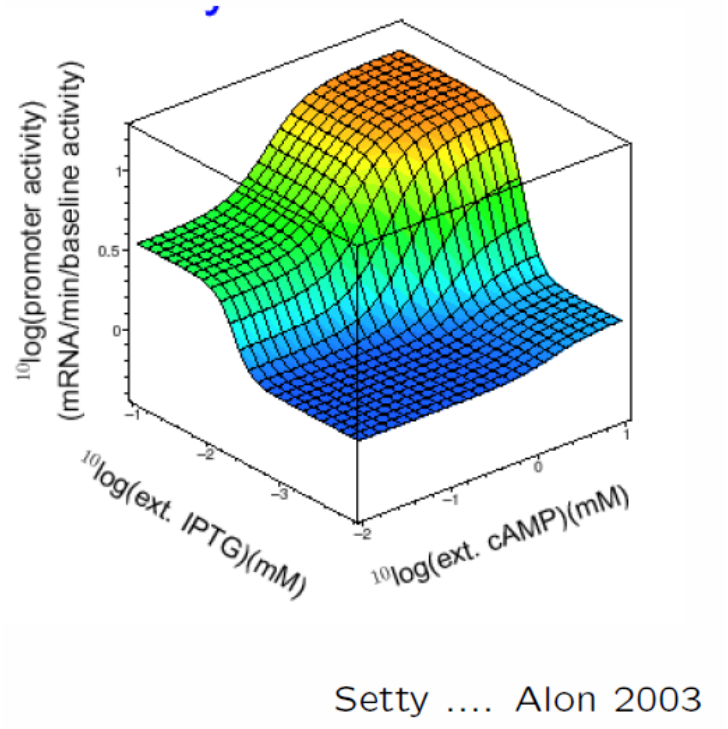


Figure 7.4: Lac operon promoter activity for varying levels of IPTG (an artificial inducer of the operon) and cAMP (signals energy state). Adapted from: (Van Hoek and Hogeweg, 2006), experimental data for figure provided by (Setty et al., 2003).

The adapted model

Measured values were used for all parameters (as far as possible) except those governing lac operon activation: those were set to some arbitrary value and then allowed to evolve, 11 in total. Paulien and von Hoek found out that reducing the dimensionality caused an evolutionary lock-in (if some parameters were bundled to ease computation, no good solution evolved). Specifically, there were five parameters that explicitly impacted promoter activity, but these five depended on seven other parameters in the model. If only the 5 supraparameters were allowed to evolve, evolution was too constrained and there was no result. This is a reminder that evolution needs enough degrees of freedom to find good solutions.

An environment was designed with fluctuating levels of glucose and lactose. This was actually one of the most difficult parts to model. Why? Because cells immediately take up glucose or lactose they encounter, and so many don't experience the same levels of glucose and lactose, because cells at sites of inflow immediately change the levels in the environment. Eventually, the environment was made by creating global, aperiodic influx of glucose and lactose in the medium, diffusion, and scaling. Growth, division, and decay of cells was implemented. There are many different time scales in the model: proteins outlast the cells (in bacteria, proteins have (much) longer lives than the cell that makes them (Koch and Levy, 1955)), there is metabolism, cell division, there are environmental switches, and there is evolution.

The operon is then initialised as a bistable switch and the aforementioned 11 parameters are allowed to evolve. The bottom line is that the model uses evolution as a trick to cope with parameter uncertainty: what evolves by itself, if allowed the freedom to do so?

Adapted model results and observables

Comparing the outcomes is also a difficult ordeal. **Ancestor traces** were used, as well as competing last common ancestors an n amount of times, and competing the last populations an n amount of times to determine what the best evolved promoter function is. If one looks at the changes in individual parameters over time, the picture that emerges is quite uninformative (Figure 7.5A). There just seem to be variations all the time, without any clear goal or optimum emerging. However, if one looks at phenotypes in the four extreme conditions (4 combinations of low/high glucose/lactose) a trend emerges: while there are still large fluctuations, especially for the bottom two, the phenotype seems to be evolving towards something (Figure 7.5B).

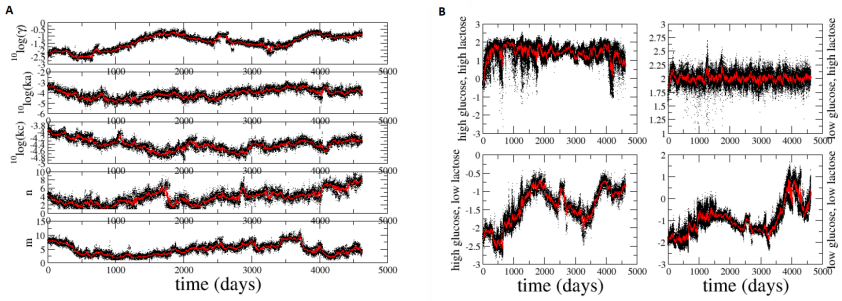


Figure 7.5: Evolved parameters and phenotypes in different nutrient conditions. A: evolutionary trajectory of the 5 main parameters allowed to evolve. There is no clear pattern here. B: evolutionary trajectory of the phenotype in four extreme conditions. Some directionality is apparent, though there are large fluctuations. Based on: (Van Hoek and Hogeweg, 2006).

If one compares the best last common ancestor with the published promoter function, they match very well (see figure below). However, there was *no bistability* in this sytem (see figure below). In fact, the model evolved away from bistability. How could this be?

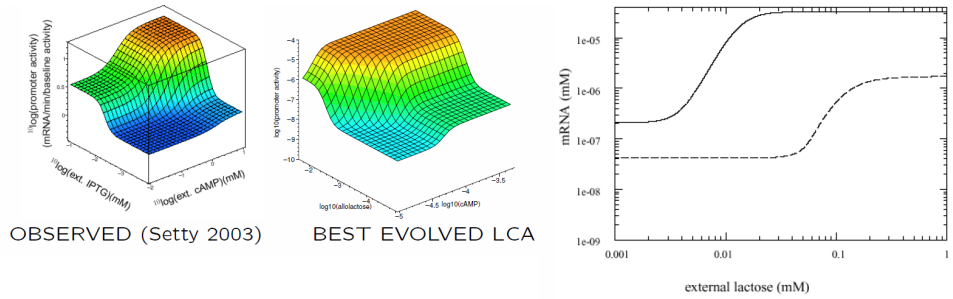


Figure 7.6: Experimentally measured promoter function and promoter function evolved in the model. Though these functions look alike, there is no bistability in the model (right). Based on: (Van Hoek and Hogeweg, 2006), experimental data for figure provided by (Setty et al., 2003).

The devil is in the missed details

The evolved promoter from the work by Paulien and von Hoek *did not have bistability*. When this work was due to be published, a disheartening 2004 experimental paper came out that claimed to have found bistability (Ozbudak et al., 2004). However, stowed far away in its tome of supplemental data and methods was a liberating sentence that said that if lactose was used *bistability did not happen*. As it turns out, experiments had always used artificial inducers of the lac operon instead of actual lactose. However, these artificial inducers cause different behaviours than actual lactose, because they cannot be metabolised. If you look at the figures of the observed behaviour by Setty, you see IPTG on one axis: this is the artificial inducer. Thus, the experimentalists had found the result as well, but failed to report it properly or reckon with what it implies.

This was the story of how models and experiments combined can still be wrong. Experimentalists are hooked to doing things under controlled conditions. For good reason, but it can go too far. In this case, the artifical inducer IPTG made sure that metabolic conditions did not change (i.e., metabolism of actual lactose would affect cellular energy levels and hence the activation of the lac operon), but this led to wrong results precisely because the artificial inducer could not be metabolised. If modelers then fit models to these wrong results, 50 years of false thinking can follow.

The new results showing that there is no bistability were beautifully illustrated in a 2012 experimental paper (Quan et al., 2012), and a 2017 modeling and experimental paper (Zander et al., 2017), but it took a long time. Only by **debugging assumptions** of both experiment and models could this finding be made.

Conclusion evolutionary modeling to test regular systems biology and experiments

Here, evolution was used as a trick to overcome parameter uncertainty (the **parameter curse** that plagues very specific models). This helped to **debug** long-held misconceptions. Evolutionary change in the parameters themselves was non-informative. Instead, the change in the phenotype (promoter behaviour) shed light on what was happening with the promoter function. Experiments are not stupid but, like models, have limitations: things that can be done, and things that cannot easily be done in a controlled fashion. Thus, experimental results should not be trusted blindly. The parameters of a biological system are a metaphorical construct. If you consider evolution, standing variation, and (slight) differences in conditions, it is easy to see that no true parameters for a system exist.

Chapter 8

Final words: on bananas, elephants, and how to model biocomplexity

This section tries to take everything we have looked at and provide some sort of final message. A daring goal, so let's get to it. After all that we have learned, the question is: how to model biocomplexity? Biology is filled to the brim with interactions on multiple scales of space and time. Many models use reductions of this complexity that can be misplaced: adaptive dynamics, for example, separates evolutionary and ecological time scales, and while some of its results may be interesting, the conclusions one reaches through such modelling are possibly only marginally applicable to the real world. Similarly, simple ODEs might make one conclude that cheaters cannot possibly coexist with altruists, while we've seen from the exercises that this not true for spatial systems.

Let's recap some defining properties of biocomplexity:

1. There is a *locality of interactions*, which often goes hand in hand with pattern formation in space. There are often many different entities in small numbers. This is an important difference with physics (modeling): particles in biology are not equal. We can capture this part of complexity by introducing locality (CA framework), and using individual-based models.

2. In biology, there are multiple levels of organisation. These levels are not separate from each other, but actively influence each other in both directions: from micro- to macro-level, and from macro- to micro-level. Early death of replicators does not make sense from the perspective of the micro-level, but it evolves due to feedback from the higher level (propagating waves, which are Darwinian entities) on the lower level.

3. There are multiple time scales: ecology, evolution, regulation, etc. Though these processes might seem as entirely different in scope and timing, we have seen that separating time scales is a dangerous business.

4. Organisms are evolved systems, and bear signatures of evolution. Neutrality is extremely important, and evolution can work to change evolution

Given these properties, how can we work with this complexity? As a first, models should be simple ENOUGH but not more so. That is a conundrum: what is the right level of complexity for tackling a certain problem? That is difficult to know, and implies that we often have to look at different modeling approaches (caricatures). Additionally, because we know that biological systems can have micro-level properties such that higher-order structures arise, so you need to formulate models that can do these things.

We observe a lot of complexity in life as well. The mechanism of evolution is random mutations and selection. Fitness is, in principle,how many descendants you create in a certain time frame (though

note that it is a time-dependent function). Complexity in theory means that you reproduce less fast (we cannot compete with a bacterium). Classical models often cannot account for the inexorable rise in complexity. If they do, it is often because of a very narrow pre-defined fitness criterion. If you take the highest growth as a criterion, complexity would never arise. Why are there bats and cats, and not just microbes (Koonin, 2011)? Bats and cats have arisen, so perhaps taking growth-rate as a fitness criterion is not informative enough to what life actually *does*. Additionally, there is the *information threshold*: given a certain mutation rate, only so much information can be retained evolutionarily.

About that elephant specifically

In a session with John Maynard Smith that Paulien attended circa 1975, he said that as biologists, we should all go to the zoo once a year and look at an elephant. Then, we should solemnly intone: 'Elephant, I believe you got about by random mutations'. Then, we should spend yet another year trying to find out *how*, because population genetic models do not explain an elephant, neither can many other models. This belief is real and founded, and it is so that elephants and other complex systems arose through random mutations and selection (Figure 8.1), but we do not, in reality, know how all this complexity arose.

Figure 8.1: Do you really need a caption for this picture? I'd say it doesn't take genius level wetware computations to figure out what you might be looking at here. Still reading? Have it your way then. It might surprise you to learn that this is, in fact, an elephant. Satisfied? (image taken from: (Manuel).)

Maynard Smith and Sathmary versus bioinformatics

Maynard Smith and Sathmary wrote an influential book called The Major Transitions in Evolution, in which they mapped the important points that increased the complexity of organisms.They followed a course that is quite different from what we have tried to do with bioinformatics. They focussed on what did happen in evolution, noting major events, and trying to find explanations for them. In fact, they reconstructed these intermediate steps such that the evolution of higher levels of complexity could be fitted to the idea of small cumulative changes with positive effects. Specifically, they wrote 'The transitions must be explained in terms of immediate selective advantage to individual replicators...'. I presume you can already see where this is going: we have seen that one of the defining marks of biology is the emergence of multiple levels and their feedbacks on each other, and we have seen that properties can emerge that are *not* to the immediate selective advantage of individual replicators!

So, what have we done? We took Darwinian selection, added local interactions (as a sort of truism, i.e. we assume most interactions in biology are best modeled by local processes) and asked: what does

happen? In other words: we did not set out to push empirical knowledge into this theory, but instead went from the minimal requirements of the theory, then let it unfold and looked at what happens.

So what are these major transitions that Maynard Smith and Sathmary identify? Let us refer to the table below ~~that we have shamelessly ripped~~ from their book:

Table 1.2 The major transitions

Replicating molecules	→	Populations of molecules in compartments
Independent replicators	→	Chromosomes
RNA as gene and enzyme	→	DNA + protein (genetic code)
Prokaryotes	→	Eukaryotes
Asexual clones	→	Sexual populations
Protists	→	Animals, plants, fungi (cell differentiation)
Solitary individuals	→	Colonies (non-reproductive castes)
Primate societies	→	Human societies (language)

Table 1.3 Conflict between selection at different levels

Form of cooperation	Exceptions
A fair meiosis	Meiotic drive, transposition
Sexual reproduction	Parthenogenesis
Differentiation of somatic cells	Escape from growth control
Non-reproductive castes of social insects	Egg-laying worker bees

What is common to all these steps? A form of self-sufficient entities which become part of a whole, which inevitably leads to conflicts. Here, this is illustrated with conflicts that are known to exist (meiotic drive, parthenogenesis). Often, these things involve a division of labour: RNA going to DNA and RNA, direct cellular reproduction changing into germline and soma, and social insects, where reproduction and maintenance are spread over distinct individuals. There is also a change from limited to unlimited inheritability, or attractor-based versus storage-based inheritability.

In the bioinformatic view, we instead confront complexity with different levels of multilevel evolution:

1. Replicators and self-organised spatial patterns arose as an automatic consequence of local interactions.

2. Replicators within protocells, or, in other words, replicators within replicators, where we choose for an explicit and pre-imposed coupling of dynamics.

3. We have looked at (virtual) cells, with duplications, deletions, large chromosomal rearrangements, plasmids, and transposons. The genome is, itself, a replicator, but because of duplications, deletions, rearrangements, etc., genes themselves are also in some form a replicator.

4. We then have multilevel genotype-phenotype mapping. We looked at the evolution of coding structures, primarily in RNA. All the stages between the genotype and the phenotype are *evolved interactions*, every molecule in a cell needs to do its job and somehow survive. Again, multiple levels are present here, though it is less clear-cut than replicators within replicators for the vesicle system.

To reiterate: Maynard Smith and Sathmary took *immediate selective advantage* for an individual replicator as a constraint. How does that square with what we have seen? Well, we never imposed that as a criterion. We didn't constrain what might happen by stating that the individual replicator should, always, benefit. Given that biology is this huge multilevel process, from strands of DNA all the way to the scale of the organism, part of this terminology of immediate benefit is very doubtful. Additionally,

evolution can select things that provide benefit in the long term as we have seen. We have also seen that often, things that do not provide immediate benefits can later lead to fitness (think of genome expansion, that is neutral or detrimental at first, but leads to high fitness later in evolution). Neutrality, too, is an example: lots of neutrality is not of immediate selective advantage, but it does make a population fitter by reducing non-neutral mutations. So is the "immediate"-part in this sentence, really true?

If we are to go by immediate benefit, we might ask: immediate benefit, *to whom*? In the spatial patterns with replicators that formed, early death could evolve. This was not of immediate benefit to the replicator, but it was of immediate benefit to the wave. Note that these sort of reasoning mistakes riddle the field: can we truly say whether something is beneficial or not out of hand? What we have seen indicates that no, we cannot.

Why models? What are they for?

Besides the different approaches to modeling complexity, we might ask what models are for, exactly. Well, models have many uses. One is that they can be a proof of principle: you can prove that it is possible for phenomena or behaviours to occur given some circumstance. Think of D.S. Wilson's group selection model.

We can also use modeling to get a baseline expectation. For example, given models of networks, we came to assume that they would have multiple attractors. By studying the basic properties of networks, we figured out that this was to be expected. In spatial systems, we found that pattern formation was the norm, rather than a special exception. In the same vein, we can use them for expectation exploration, where we can ask: what happens if we assume...?

Modeling also allows us to plant our flags in hitherto unexplored regions. For example, the paradigm RNA genotype-phenotype mapping system that revealed many interesting properties of the evolutionary process provided a search image, a first flag planted firmly in the soil of discovery. From this beacon, similar properties were soon found in many more aspects of biology. Thus, models can provide **search images**: often extreme examples that nevertheless show what might be possible. Another example is the Lotka-Volterra model: it is an absolutely preposterous model (prey never die, for one), but it taught us a lot about population dynamics and more.

Models can help us *debug* assumptions or ideas. We saw this explicitly in the case of the Lac operon, where 50 years of experimental and model knowledge was dispelled with a model that evolved parameters controlling operon expression to optimal values. You can draw an analogy with programming: if you just look at the code, it is hard to find bugs. But if you instead let the program run, it is much easier to find where the logic goes wrong. The same holds for scientific theories and findings: models force you to make things explicit, allowing efficient debugging.

Models can also be predictive. However, in this case, not only do you need a correct model, you also need parameters and initial conditions. If we compare aerodynamics and weather prediction, we can say that models of aerodynamics work pretty well, but weather prediction is still not too good, simply because the initial conditions are almost impossible to know precisely enough. You also need to ask yourself what they are predictive of exactly, and what the constraints are. For example, Newton's apple was a model for how forces work, which eventually led to much more than you might at first think. This is another example of Results ++.

There are many flavours and purposes of models, but the simplest answer is that they are for *understanding*.

Modes of explanation

Let us start with a great Dutch saying that showcases the very best of our nation's creative genius: 'Waarom, waarom, waarom zijn de bananen krom?'. Why, oh why, are bananas bent? Well, the first mode of explanation is that a yellow fruit that is bent is a banana, so if it is not bent is is not a banana. This is a tautology.

This particular tautology is uninformative. Such a tautology is often invoked when it comes to survival of the fittest, as if that marginalises how important it is. On the other hand, survival of the fittest for any

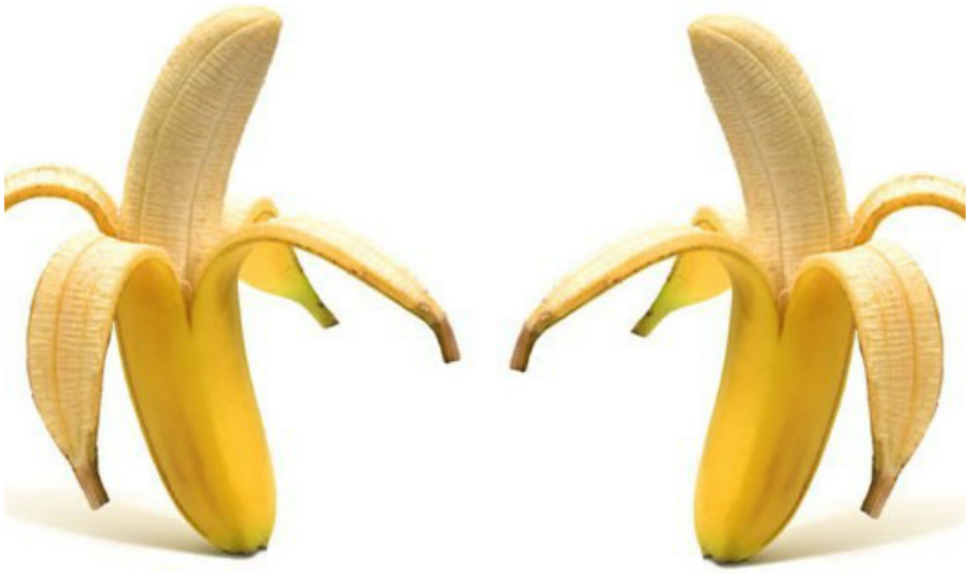


Figure 8.2: "Waarom, waarom, waarom zijn de bananen krom?"

explicit fitness is not necessarily true: the host-parasitoid example showed that low-fitness parasitoids in the spiral centre nevertheless ruled the system in the long term. At the same time, some tautologies can be very deep insights. For example, during the exercises, you will have discussed that survival of the fittest is equal to competitive exclusion, they are thus a tautology. However, understanding where these different terms came from can nevertheless be insightful.

Another mode of explanation could be that in almost all cases, bananas are bent. Hence, a straight banana is a pathological case. However, we often think a straight banana would be the simplest case. Oftentimes, what has been researched are specific cases, such as model organisms. It is very interesting to know what generic properties are. You can also think from the angle of why you would not want to be bent? Are we even asking the right questions? In the course we have discussed many examples where the question 'why' just turned out to be the wrong question to ask! For example: why are lymph nodes structured as they are? Because structure is the default, suboptimal case, while randomness is in fact very hard to achieve!

A third mode of explanation is imposing a certain value or benefit. A bent banana is optimal, one could say. However, if I do not define what optimal means *a priori*, I can say a phenomenon is optimal for something. For example, if there is a robot that continuously walks in circles, you can say that is optimal for guarding an area. However, in truth, one of the motors could just be blocked so only the left wheel moves, turning the robot in circles. Thus, optimality by itself is an empty statement. Note also that a lot of behaviour that is hailed as (evolutionaryily) optimised could just be TODO-based behaviour: simple rules whose resultant behaviours are determined by the environment. We have also learnt to ask the question 'optimal for whom?'. Early death is...well, a death sentence for the individual replicator, but not for the higher-level wave.

Yet another mode of explanation is one of side-effects. Perhaps being bent is a side-effect of growing in bunches, or a side effect of growing in the presence of gravity. We could also explain away banana bentness as a side-effect of growth. Not evolutionary, but rather TODO (i.e. an individual banana should grow, if the forces during growth make it bent, that is simply a side-effect of the needed behaviour). We can possibly identify the gene, or the genes, that cause bentness in bananas. If we knock out enough genes, we might be able to pinpoint, on a genetic level, what causes a banana to be bent. We could then reiterate, and ask the question why this gene is the way it is, why plants are the way they are, why the earth is the way it is, etc. This infinite regress will eventually make us end up at the big bang. Surely, it is not useful to say that bananas are bent because of the big bang (?).

To wrap up, we can (attempt to) make a detailed model of real bananas and see whether we can

explain why they grow in a bent shape if we input all known data. The point of this exercise is that one question, no matter how nonsensical, can have many different answers, given from a plethora of viewpoints and levels. It is best to view a question, problem, or behaviour, from many different angles to get the 'complete' answer. We have advocated this during the course, by using similar rules in different systems, and seeing whether similar behaviours show up, showing that they are robust to changes in model formalism (our viewpoint) and general.

One last time, Mr. Elephant

Elephant, I believe you have come about by random mutations, local interactions, multi-level selection, genome structuring, mutational priming (non-random mutations), and who knows what else, and I can only understand you by simplification. But not to one level.

Thanks for reading, folks!

Chapter 9

Glossary of bold terms

This section attempts to summarize all the jargon you have come to learn to appreciate by reading this document. Is something missing? Please let us know!

Bibliography

Patrick Abbot, Jun Abe, John Alcock, Samuel Alizon, Joao A.C. Alpedrinha, Malte Andersson, Jean Baptiste Andre, Minus Van Baalen, Francois Balloux, Sigal Balshine, Nick Barton, Leo W. Beukeboom, Jay M. Biernaskie, Trine Bilde, Gerald Borgia, Michael Breed, Sam Brown, Redouan Bshary, Angus Buckling, Nancy T. Burley, Max N. Burton-Chellew, Michael A. Cant, Michel Chapuisat, Eric L. Charnov, Tim Clutton-Brock, Andrew Cockburn, Blaine J. Cole, Nick Colegrave, Leda Cosmides, Iain D. Couzin, Jerry A. Coyne, Scott Creel, Bernard Crespi, Robert L. Curry, Sasha R.X. Dall, Troy Day, Janis L. Dickinson, Lee Alan Dugatkin, Claire El Mouden, Stephen T. Emlen, Jay Evans, Regis Ferriere, Jeremy Field, Susanne Foitzik, Kevin Foster, William A. Foster, Charles W. Fox, Juergen Gadau, Sylvain Gandon, Andy Gardner, Michael G. Gardner, Thomas Getty, Michael A.D. Goodisman, Alan Grafen, Rick Grosberg, Christina M. Grozinger, Pierre Henri Gouyon, Darryl Gwynne, Paul H. Harvey, Ben J. Hatchwell, Jürgen Heinze, Heikki Helantera, Ken R. Helms, Kim Hill, Natalie Jiricny, Rufus A. Johnstone, Alex Kacelnik, E. Toby Kiers, Hanna Kokko, Jan Komdeur, Judith Korb, Daniel Kronauer, Rolf Kümmerli, Laurent Lehmann, Timothy A. Linksvayer, Sébastien Lion, Bruce Lyon, James A.R. Marshall, Richard McElreath, Yannis Michalakis, Richard E. Michod, Douglas Mock, Thibaud Monnin, Robert Montgomerie, Allen J. Moore, Ulrich G. Mueller, Ronald Noë, Samir Okasha, Pekka Pamilo, Geoff A. Parker, Jes S. Pedersen, Ido Pen, David Pfennig, David C. Queller, Daniel J. Rankin, Sarah E. Reece, Hudson K. Reeve, Max Reuter, Gilbert Roberts, Simon K.A. Robson, Denis Roze, Francois Rousset, Olav Rueppell, Joel L. Sachs, Lorenzo Santorelli, Paul Schmid-Hempel, Michael P. Schwarz, Tom Scott-Phillips, Janet Shellmann-Sherman, Paul W. Sherman, David M. Shuker, Jeff Smith, Joseph C. Spagna, Beverly Strassmann, Andrew V. Suarez, Liselotte Sundström, Michael Taborsky, Peter Taylor, Graham Thompson, John Tooby, Neil D. Tsutsui, Kazuki Tsuji, Stefano Turillazzi, Francisco Úbeda, Edward L. Vargo, Bernard Voelkl, Tom Wenseleers, Stuart A. West, Mary Jane West-Eberhard, David F. Westneat, Diane C. Wiernasz, Geoff Wild, Richard Wrangham, Andrew J. Young, David W. Zeh, Jeanne A. Zeh, and Andrew Zink. Inclusive fitness theory and eusociality. *Nature*, 471(7339), 2011. ISSN 00280836. doi: 10.1038/nature09831.

Katarzyna Adamala and Jack W. Szostak. Nonenzymatic template-directed RNA synthesis inside model protocells. *Science*, 342(6162):1098–1100, 2013. ISSN 10959203. doi: 10.1126/science.1241888.

Christoph Adami, Charles Ofria, and Travis C Collier. Evolution of biological complexity. *Proceedings of the National Academy of Sciences*, 97(9):4463–4468, 2000.

Richard D. Alexander and Gerald Borgia. Group Selection, Altruism, and the Levels of Organization of Life. *Annual Review of Ecology and Systematics*, 1978.

Uri Alon. Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461, 2007. ISSN 1471-0056. doi: 10.1038/nrg2102. URL http://www.nature.com/doifinder/10.1038/nrg2102.

W Ross Ashby. Design for a Brain an Introduction To. *Director*, 80(4):295, 1999. ISSN 13665278. doi: 10.2307/3006723. URL http://pcp.vub.ac.be/books/IntroCyb.pdf.

Claudia Bank, Ryan T. Hietpas, Jeffrey D. Jensen, and Daniel N.A. Bolon. A Systematic Survey of an Intragenic Epistatic Landscape. *Molecular Biology and Evolution*, 32(1):229–238, 2015. ISSN 15371719. doi: 10.1093/molbev/msu301.

Jeffrey E. Barrick, Dong Su Yu, Sung Ho Yoon, Haeyoung Jeong, Tae Kwang Oh, Dominique Schneider, Richard E. Lenski, and Jihyun F. Kim. Genome evolution and adaptation in a long-term experiment with Escherichia coli. *Nature*, 461(7268):1243–1247, 2009. ISSN 00280836. doi: 10.1038/nature08480. URL http://dx.doi.org/10.1038/nature08480.

Bérénice Batut, David P. Parsons, Stephan Fischer, Guillaume Beslon, and Carole Knibbe. In silico experimental evolution: a tool to test evolutionary scenarios. *BMC bioinformatics*, 14 Suppl 1(Suppl 15), 2013. ISSN 14712105. doi: 10.1186/1471-2105-14-S15-S11.

Hubertus J.E. Beaumont, Jenna Gallie, Christian Kost, Gayle C. Ferguson, and Paul B. Rainey. Experimental evolution of bet hedging. *Nature*, 462(7269):90–93, 2009. ISSN 00280836. doi: 10.1038/nature08504. URL http://dx.doi.org/10.1038/nature08504.

Joost B. Beltman, Athanasius F.M. Marée, Jennifer N. Lynch, Mark J. Miller, and Rob J. de Boer. Lymph node topology dictates T cell migration behavior. *The Journal of Experimental Medicine*, 204 (4):771–780, 2007. ISSN 0022-1007. doi: 10.1084/jem.20061278. URL http://www.jem.org/lookup/doi/10.1084/jem.20061278.

Guillaume Beslon, Jonas Abernot, Santiago F. Elena, Dominique Schneider, Paulien Hogeweg, Susan Stepney, and Christophe Rigotti. EvoEvo Deliverable 6.8: Final report. Technical report, 2017.

William A. Beyer, Peter H. Sellers, and Michael S. Waterman. Stanislaw M. Ulam's contributions to theoretical theory. *Letters in Mathematical Physics*, 10(2-3):231–242, 1985. ISSN 03779017. doi: 10.1007/BF00398163.

Christof K. Biebricher and William C. Gardiner. Molecular evolution of RNA in vitro. *Biophysical Chemistry*, 66(2-3):179–192, 1997. ISSN 03014622. doi: 10.1016/S0301-4622(97)00059-8.

Priscila Biller, Carole Knibbe, Guillaume Beslon, and Eric Tannier. Comparative genomics on artificial life. In *Pursuit of the Universal: 12th Conference on Computability in Europe, CiE 2016, Paris, France, June 27-July 1, 2016, Proceedings 12*, pages 35–44. Springer, 2016.

I. Boekhorst and P. Hogeweg. Self-structuring in artifical "chimps" offers new hypothesis for male grouping in chimpanzees. *Behaviour*, 130(3-4):229–252, 1994.

M C Boerlijst, M E Lamers, and P Hogeweg. Evolutionary Consequences of Spiral Waves in a Host Parasitoid System. *Proceedings of the Royal Society of London Series B-Biological Sciences*, 253 (1336):15–18, 1993.

Maarten C. Boerlijst and Pauline Hogeweg. Spatial gradients enhance persistence of hypercycles. *Physica D: Nonlinear Phenomena*, 88(1):29–39, 1995. ISSN 01672789. doi: 10.1016/0167-2789(95)00178-7.

M.C. Boerlijst and P. Hogeweg. SPIRAL WAVE STRUCTURE IN PRE-BIOTIC EVOLUTION: HYPERCYCLES STABLE AGAINST PARASITES*. *Physica D*, 48:17–28, 1991.

V Braitenberg and C Langton. Artificial life: SFI studies in the sciences of complexity. 1988.

Bonita J Brewer and Walton L Fangman. A Replication Fork Barrier at the 3 ' End of Yeast Ribosomal RNA Genes. 55:637–643, 1988.

Rodney A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47(1-3):139–159, 1991. ISSN 00043702. doi: 10.1016/0004-3702(91)90053-M.

Arthur Walter Burks. *Essays on cellular automata*. University of Illinois Press, 1970.

E S Colizzi and P Hogeweg. Mutational load is ameliorated by increased transcriptional load-associated mutations, if these are biased towards duplications and deletions. In *2nd EvoEvo Workshop, Amsterdam (NL), September 2016*, 2016.

Enrico Sandro Colizzi and Paulien Hogeweg. Evolution of functional diversification within quasispecies. *Genome Biology and Evolution*, 6(8):1990–2007, 2014. ISSN 17596653. doi: 10.1093/gbe/evu150.

Tom M. Conrad, Nathan E. Lewis, and Bernhard O. Palsson. Microbial laboratory evolution in the era of genome-scale science. *Molecular Systems Biology*, 7(509), 2011. ISSN 17444292. doi: 10.1038/msb.2011.42.

Otto X. Cordero and Paulien Hogeweg. Feed-forward loop circuits as a side effect of genome evolution. *Molecular Biology and Evolution*, 23(10):1931–1936, 2006. ISSN 07374038. doi: 10.1093/molbev/msl060.

Iain D. Couzin, Jens Krause, Nigel R Franks, and Simon A Levin. Effective leadership and decision-making in animal groups on themove. *Nature*, 433(February):513–516, 2005. ISSN 1476-4687. doi: 10.1038/nature03236.

Anton Crombach and Paulien Hogeweg. Chromosome rearrangements and the evolution of genome structuring and adaptability. *Molecular Biology and Evolution*, 24(5):1130–1139, 2007. ISSN 07374038. doi: 10.1093/molbev/msm033.

Anton Crombach and Paulien Hogeweg. Evolution of evolvability in gene regulatory networks. *PLoS Computational Biology*, 4(7), 2008. ISSN 1553734X. doi: 10.1371/journal.pcbi.1000112.

J P Crutchfield and M Mitchell. The evolution of emergent computation. *Proceedings of the National Academy of Sciences of the United States of America*, 92(23):10742–10746, 1995. ISSN 0027-8424. doi: 10.1073/pnas.92.23.10742.

Thomas D. Cuypers and Paulien Hogeweg. Virtual genomes in flux: An interplay of neutrality and adaptability explains genome expansion and streamlining. *Genome Biology and Evolution*, 4(3):212–229, 2012. ISSN 17596653. doi: 10.1093/gbe/evr141.

Thomas D. Cuypers and Paulien Hogeweg. A Synergism between Adaptive Effects and Evolvability Drives Whole Genome Duplication to Fixation. *PLoS Computational Biology*, 10(4), 2014. ISSN 15537358. doi: 10.1371/journal.pcbi.1003547.

Thomas D. Cuypers, Jacob P. Rutten, and Paulien Hogeweg. Evolution of evolvability and phenotypic plasticity in virtual cells. *BMC evolutionary biology*, 17(1):60, 2017. ISSN 14712148. doi: 10.1186/s12862-017-0918-y.

Charles Darwin. On the origins of species by means of natural selection. *London: Murray*, 247:1859, 1859.

Rajarshi Das, James P. Crutchfield, Melanie Mitchell, and James E. Hanson. Evolving Globally Synchronized Cellular Automata. *Proceedings of the Sixth International Conference on Genetic Algorithms*, 005:1–12, 1995. doi: 10.1002/3527601945.ch11.

Richard Dawkins. The Extended Phenotype-The Gene as the Unit of Selection, 1982.

Richard Dawkins. The extended phenotype: The gene as the unit of selection. *Ethology and Sociobiology*, 5(3):215–217, 1984. ISSN 01623095. doi: 10.1016/0162-3095(84)90028-1.

Folkert K. De Boer and Paulien Hogeweg. Eco-evolutionary dynamics, coding structure and the information threshold. *BMC Evolutionary Biology*, 10(1), 2010. ISSN 14712148. doi: 10.1186/1471-2148-10-361.

Folkert K. de Boer and Paulien Hogeweg. Less can be more: RNA-adapters may enhance coding capacity of replicators. *PLoS ONE*, 7(1):1–11, 2012. ISSN 19326203. doi: 10.1371/journal.pone.0029952.

Daniel P Depledge, Kalanghad Puthankalam Srinivas, Tomohiko Sadaoka, Devin Bready, Yasuko Mori, Dimitris G Placantonakis, Ian Mohr, and Angus C Wilson. Direct rna sequencing on nanopore arrays redefines the transcriptional complexity of a viral pathogen. *Nature communications*, 10(1):754, 2019.

Theodosius Dobzhansky. Nothing in biology makes sense except in the light of evolution. *The American Biology Teacher*, 75(2):3–10, 1973. ISSN 00027685. doi: 10.2307/4444260.

Jennifer A. Doudna and Thomas R. Cech. The chemical repertoire of natural ribozymes. *Nature*, 418 (6894):222–228, 2002. ISSN 00280836. doi: 10.1038/418222a.

M. J. Dunham, H. Badrane, T. Ferea, J. Adams, P. O. Brown, F. Rosenzweig, and D. Botstein. Characteristic genome rearrangements in experimental evolution of Saccharomyces cerevisiae. *Proceedings of the National Academy of Sciences*, 99(25):16144–16149, 2002a. ISSN 0027-8424. doi: 10.1073/pnas.242624799. URL http://www.pnas.org/cgi/doi/10.1073/pnas.242624799.

Maitreya J Dunham, Hassan Badrane, Tracy Ferea, Julian Adams, Patrick O Brown, Frank Rosenzweig, and David Botstein. Characteristic genome rearrangements in experimental evolution of saccharomyces cerevisiae. *Proceedings of the National Academy of Sciences*, 99(25):16144–16149, 2002b.

Patrick P. Edger and J. Chris Pires. Gene and genome duplications: The impact of dosage-sensitivity on the fate of nuclear genes. *Chromosome Research*, 17(5):699–717, 2009. ISSN 09673849. doi: 10.1007/s10577-009-9055-9.

Manfred Eigen. Selforganization of matter and the evolution of biological macromolecules. *Die Naturwissenschaften*, 58(10):465–523, 1971. ISSN 00281042. doi: 10.1007/BF00623322.

Manfred Eigen and Peter Schuster. The Hypercyde. *Die Naturwissenschaften*, 64(2):541, 1977. ISSN 0028-1042, 1432-1904. doi: 10.1007/BF00420631.

Manfred Eigen and Peter Schuster. The Hypercycle. *Naturwissenschaften*, 65(1):7–41, 1978. ISSN 0028-1042, 1432-1904. doi: 10.1007/BF00420631. URL http://link.springer.com/article/10.1007/BF00420631{%}5Cnhttp://link.springer.com/content/pdf/10.1007{%}2FBF00420631.pdf.

Manfred Eigen, John McCaskill, and Peter Schuster. Molecular quasi-species. *Journal of Physical Chemistry*, 92(24):6881–6891, 1988. ISSN 00223654. doi: 10.1021/j100335a010.

Eric H Ekland and David P Bartel. The secondary structure and sequence optimization of an RNA ligase ribozyme. *Nucleic acids research*, 23(16):3231–3238, 1995.

Eric H Ekland, Jack W Szostak, and David P Bartel. Structurally complex and highly active RNA ligases derived from random RNA sequences. *Science*, 269(5222):364–370, 1995.

Niles Eldredge and Stephen Jay Gould. Punctuated equilibria: an alternative to phyletic gradualism, 1972. ISSN 0003066X. URL http://www.somosbacteriasyvirus.com/phyletic.pdf{%}5Cnpapers3://publication/uuid/E4D0D86A-52A5-4E3C-882E-85B5984334E7.

Santiago F Elena, Vaughn S Cooper, Richard E Lenski, Santiago F Elena, Vaughn S Cooper, and Richard E Lenski. Punctuated Evolution Caused by Selection of Rare Beneficial Mutations Published by : American Association for the Advancement of Science Stable URL : http://www.jstor.org/stable/2889330 JSTOR is a not-for-profit service that helps scholars , researchers ,. 272(5269):1802–1804, 1996.

M. Eric Schranz, Setareh Mohammadin, and Patrick P. Edger. Ancient whole genome duplications, novelty and diversification: The WGD Radiation Lag-Time Model. *Current Opinion in Plant Biology*, 15(2):147–153, 2012. ISSN 13695266. doi: 10.1016/j.pbi.2012.03.011. URL http://dx.doi.org/10.1016/j.pbi.2012.03.011.

P. G. Falkowski, T. Fenchel, and E. F. Delong. The Microbial Engines That Drive Earth 's Biogeochemical Cycles. *Science*, 320(5879):1034–1039, 2008. ISSN 1095-9203. doi: 10.1126/science.1153213. URL http://www.ncbi.nlm.nih.gov/pubmed/18497287.

T L Ferea, D Botstein, P O Brown, and R F Rosenzweig. Systematic changes in gene expression patterns following adaptive evolution in yeast. *Proceedings of the National Academy of Sciences of the United States of America*, 96(17):9721–6, 1999. ISSN 0027-8424. doi: 10.1073/pnas.96.17.9721. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=22277{&}tool=pmcentrez{&}rendertype=abstract.

Regis Ferriere and Richard E Michod. Inclusive fitness in evolution. *Nature*, 471(7339):E6, 2011.

P.J Fleming and R.C Purshouse. Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice*, 10(11):1223–1241, 2002. ISSN 09670661. doi: 10.1016/S0967-0661(02)00081-3.

Haoyue Fu, Lianping Yang, and Xiangde Zhang. An RNA secondary structure prediction method based on minimum and suboptimal free energy structures. *Journal of Theoretical Biology*, 380:473–479, 2015. ISSN 10958541. doi: 10.1016/j.jtbi.2015.05.007. URL http://dx.doi.org/10.1016/j.jtbi.2015.05.007.

Laiyi Fu, Yingxin Cao, Jie Wu, Qinke Peng, Qing Nie, and Xiaohui Xie. Ufold: fast and accurate rna secondary structure prediction with deep learning. *Nucleic acids research*, 50(3):e14–e14, 2022.

Tatiana García-Muse and Andrés Aguilera. Transcription-replication conflicts: How they occur and how they are resolved. *Nature Reviews Molecular Cell Biology*, 17(9):553–563, 2016. ISSN 14710080. doi: 10.1038/nrm.2016.88. URL http://dx.doi.org/10.1038/nrm.2016.88.

D T Gillespie. DT Gillespie, J. Phys. Chem. 81, 2340 (1977). *J. Phys. Chem.*, 81:2340, 1977.

Antoine Giraud, Miroslav Radman, Ivan Matic, and François Taddei. The rise and fall of mutator bacteria. *Current Opinion in Microbiology*, 4(5):582–585, 2001. ISSN 13695274. doi: 10.1016/S1369-5274(00)00254-X.

Jane Goodall. The chimpanzees of Gombe: Patterns of behavior. *Cambridge Mass*, 1986.

S. A. Grady, M. Y. Hussaini, and M. M. Abdullah. Placement of wind turbines using genetic algorithms. *Renewable Energy*, 30(2):259–270, 2005. ISSN 09601481. doi: 10.1016/j.renene.2004.05.007.

François Graner and James Glazier. Simulation of biological cell sorting using a two-dimensional extended Potts model, 1992. ISSN 0031-9007. URL http://link.aps.org/doi/10.1103/PhysRevLett.69.2013.

Michael D. Gross and Eli C. Siegel. Incidence of mutator strains in Escherichia coli and coliforms in nature. *Mutation Research Letters*, 91(2):107–110, 1981. ISSN 01657992. doi: 10.1016/0165-7992(81)90081-6.

R K Guy and John Horton Conway. Winning ways for your mathematical plays, 1982.

W D Hamilton. The Evolution of Altruistic Behavior. *The American Naturalist*, 97(896):354–356, 1963.

W. D. Hamilton. The genetical evolution of social behaviour. II. *Journal of Theoretical Biology*, 7(1):17–52, 1964. ISSN 10958541. doi: 10.1016/0022-5193(64)90039-6.

James E Hanson and James P. Crutchfield. Computational Mechanics of Cellular Automata. 1997.

Alexander H Harcourt, Frans B M de Waal, and Others. *Coalitions and alliances in humans and other animals*. Oxford University Press Oxford, 1992.

Yuuki Hayashi, Takuyo Aita, Hitoshi Toyota, Yuzuru Husimi, Itaru Urabe, and Tetsuya Yomo. Experimental rugged fitness landscape in protein sequence space. *PLoS ONE*, 1(1), 2006. ISSN 19326203. doi: 10.1371/journal.pone.0000096.

Charlotte K. Hemelrijk and Hanno Hildenbrandt. Schools of fish and flocks of birds: Their shape and internal structure by self-organization. *Interface Focus*, 2(6):726–737, 2012. ISSN 20428901. doi: 10.1098/rsfs.2012.0025.

Edward Allen Herre and William T Wcislo. In defence of inclusive fitness theory. *Nature*, 471(7339):E8, 2011.

B Hesper and P Hogeweg. Bioinformatica: een werkconcept. *Kameleon*, 1(6):28–29, 1970.

P G Higgs and B Derrida. Genetic-Distance and Species Formation in Evolving Populations. *Journal of Molecular Evolution*, 35:454–465, 1992.

W. Daniel Hillis. Co-evolving parasites improve simulated evolution as an optimization procedure. *Physica D: Nonlinear Phenomena*, 42(1-3):228–234, 1990. ISSN 01672789. doi: 10.1016/0167-2789(90)90076-2.

I L Hofacker, M Fekete, and P Stadler. Secondary Structure Prediction for Aligned RNA Sequences. *J. Mol. Biol.*, 319(5):1059–1066, 2002. ISSN 0022-2836. doi: 10.1016/s0022-2836(02)00308-x. URL papers3://publication/doi/10.1016/S0022-2836(02)00308-X.

P. Hogeweg. Cellular automata as a paradigm for ecological modeling. *Applied Mathematics and Computation*, 27(1):81–100, 1988. ISSN 00963003. doi: 10.1016/0096-3003(88)90100-2.

P. Hogeweg. Local T-T cell and T-B cell interactions: a cellular automaton approach. *Immunology Letters*, 22(2):113–122, 1989. ISSN 01652478. doi: 10.1016/0165-2478(89)90177-6.

P. Hogeweg. Shapes in the shadow: Evolutionary dynamics of morphogenesis. *Artificial Life*, 6(1): 85–101, 2000. ISSN 10645462. doi: 10.1162/106454600568339.

P. Hogeweg and B. Hesper. Socioinformatic processes: MIRROR modelling methodology, 1985. ISSN 10958541.

P Hogeweg and B Hesper. Evolution as pattern processing: TODO as substrate for evolution. *Meyer and Wilson [MW91]*, pages 492–497, 1991.

Paulien Hogeweg and B. Hesper. The ontogeny of the Interaction Structure in Bumble Bee Colonies: A MIRROR Model. *Behavioral Ecology and Sociobiology*, 1983.

John Henry Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

Sui Huang and Donald E. Ingber. Shape-dependent control of cell growth, differentiation, and apoptosis: Switching between attractors in cell regulatory networks. *Experimental Cell Research*, 261(1):91–103, 2000. ISSN 00144827. doi: 10.1006/excr.2000.5044.

Sui Huang, Gabriel Eichler, Yaneer Bar-Yam, and Donald E. Ingber. Cell fates as high-dimensional attractor states of a complex gene regulatory network. *Physical Review Letters*, 94(12):1–4, 2005. ISSN 00319007. doi: 10.1103/PhysRevLett.94.128701.

M. A. Huynen, P. F. Stadler, and W. Fontana. Smoothness within ruggedness: the role of neutrality in adaptation. *Proceedings of the National Academy of Sciences*, 93(1):397–401, 1996. ISSN 0027-8424. doi: 10.1073/pnas.93.1.397. URL http://www.pnas.org/cgi/doi/10.1073/pnas.93.1.397.

Martijn A. Huynen. Exploring phenotype space through neutral evolution. *Journal of Molecular Evolution*, 43(3):165–169, 1996. ISSN 00222844. doi: 10.1007/BF02338823.

Martijn A Huynen and Paulien Hogeweg. Pattern generation in molecular evolution: exploitation of the variation in rna landscapes. *Journal of Molecular Evolution*, 39:71–79, 1994.

Martijn A. Huynen, Danielle A. Konings, and Paulien Hogeweg. Multiple Coding and the Evolutionary Properties of RNA Secondary Structure, 1993.

L. Jaeger, M. C. Wright, and G. F. Joyce. A complex ligase ribozyme evolved in vitro from a group I ribozyme domain. *Proceedings of the National Academy of Sciences*, 96(26):14712–14717, 1999. ISSN 0027-8424. doi: 10.1073/pnas.96.26.14712. URL http://www.pnas.org/cgi/doi/10.1073/pnas.96.26.14712.

Srdja Janković and Milan M. Ćirković. Evolvability Is an Evolved Ability: The Coding Concept as the Arch-Unit of Natural Selection. *Origins of Life and Evolution of Biospheres*, 46(1):67–79, 2016. ISSN 15730875. doi: 10.1007/s11084-015-9464-z.

Gerald F. Joyce. Forty years of in vitro evolution. *Angewandte Chemie - International Edition*, 46(34): 6420–6436, 2007. ISSN 14337851. doi: 10.1002/anie.200701369.

Gerald F. Joyce. Bit by bit: The darwinian basis of life. *PLoS Biology*, 10(5), 2012. ISSN 15449173. doi: 10.1371/journal.pbio.1001323.

Jos Käfer, Paulien Hogeweg, and Athanasius F.M. Marée. Moving forward moving backward: Directional sorting of chemotactic cells due to size and adhesion differences. *PLoS Computational Biology*, 2(6): 0518–0529, 2006. ISSN 1553734X. doi: 10.1371/journal.pcbi.0020056.

S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467, 1969. ISSN 10958541. doi: 10.1016/0022-5193(69)90015-0.

Stuart Kauffman. The large scale structure and dynamics of gene control circuits: an ensemble approach. *Journal of Theoretical Biology*, 44(1):167–190, 1974.

Stuart Kauffman and Simon Levin. Towards a general theory of adaptive walks on rugged landscapes. *Journal of theoretical Biology*, 128(1):11–45, 1987.

Motoo Kimura. *The neutral theory of molecular evolution*. Cambridge University Press, 1983.

Carole Knibbe, Antoine Coulon, Olivier Mazet, Jean Michel Fayard, and Guillaume Beslon. A long-term evolutionary pressure on the amount of noncoding DNA. *Molecular Biology and Evolution*, 24(10): 2344–2353, 2007. ISSN 07374038. doi: 10.1093/molbev/msm165.

Arthur L. Koch and Richard H. Levy. Protein turnover in growing cultures of Escherichia coli. (8), 1955. doi: 10.4324/9780203105658. URL http://www.tandfebooks.com/isbn/9780203105658.

D. A.M. Konings, L. P. van Duijn, H. O. Voorma, and P. Hogeweg. Minimal energy foldings of eukaryotic mRNAs form a separate leader domain. *Journal of Theoretical Biology*, 127(1):63–78, 1987. ISSN 10958541. doi: 10.1016/S0022-5193(87)80161-3.

Eugene V Koonin. *The logic of chance: the nature and origin of biological evolution*. FT press, 2011.

V. Lila Koumandou, Bill Wickstead, Michael L. Ginger, Mark Van Der Giezen, Joel B. Dacks, and Mark C. Field. Molecular paleontology and complexity in the last eukaryotic common ancestor. *Critical Reviews in Biochemistry and Molecular Biology*, 48(4):373–396, 2013. ISSN 10409238. doi: 10.3109/10409238.2013.821444.

V.I. Krinsky and K. I. Agladaze. INTERACTION OF ROTATING WAVES IN AN ACTIVE CHEMICAL MEDIUM. *Physica D: Nonlinear Phenomena*, 1983.

Nick Lane and William F. Martin. The Origin of Membrane Bioenergetics. *Cell*, 151(7):1406–1416, 2012. ISSN 00928674. doi: 10.1016/j.cell.2012.11.050.

Susan Q. Lang, David A. Butterfield, Mitch Schulte, Deborah S. Kelley, and Marvin D. Lilley. Elevated concentrations of formate, acetate and dissolved organic carbon found at the Lost City hydrothermal field. *Geochimica et Cosmochimica Acta*, 74(3):941–952, 2010. ISSN 00167037. doi: 10.1016/j.gca.2009.10.045. URL http://dx.doi.org/10.1016/j.gca.2009.10.045.

Christopher G Langton. Artificial Life 1989. *ARTIFICIAL LIFE-II*, 1991.

T.I. Lee, N.J. Rinaldi, F. Robert, D.T. Odom, Z. Bar-Joseph, G.K. Gerber, N.M. Hannett, C.T. Harbison, C.M. Thompson, I. Simon, J. Zeitlinger, E.G. Jennings, H.L. Murray, D.B. Gordon, B. Ren, J.J. Wyrick, J.B. Tagne, T.L. Volkert, E. Fraenkel, D.K. Gifford, and R.A. Young. Transcriptional regulatory networks in Saccharomyces cerevisiae. 298(5594):799–804, 2002. URL https://www.science.org/doi/10.1126/science.1075090.

Egbert G Leigh. How does selection reconcile individual advantage with the good of the group? *Evolution*, 74(10):4542–4546, 1977. ISSN 0027-8424. doi: 10.1073/pnas.74.10.4542.

R. E. Lenski and M. Travisano. Dynamics of adaptation and diversification: a 10,000-generation experiment with bacterial populations. *Proceedings of the National Academy of Sciences*, 91(15):6808–6814, 1994. ISSN 0027-8424. doi: 10.1073/pnas.91.15.6808. URL http://www.pnas.org/cgi/doi/10.1073/pnas.91.15.6808.

R E Lenski, M R Rose, S C Simpson, and S C Tadler. Long-term experimental evolution in Escherichia coli. *The American Naturalist*, 138(6):1315–1341, 1991.

F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences*, 101(14):4781–4786, 2004. ISSN 0027-8424. doi: 10.1073/pnas.0305937101. URL http://www.pnas.org/cgi/doi/10.1073/pnas.0305937101.

Pier Luigi Luisi, Peter Walde, and Thomas Oberholzer. Lipid vesicles as possible intermediates in the origin of life. *Current Opinion in Colloid and Interface Science*, 4(1):33–39, 1999. ISSN 13590294. doi: 10.1016/S1359-0294(99)00012-6.

Michael Lynch. Evolution of the mutation rate. *Trends in Genetics*, 26(8):345–352, 2010. ISSN 01689525. doi: 10.1016/j.tig.2010.05.003. URL http://dx.doi.org/10.1016/j.tig.2010.05.003.

Francois Major and Philippe Thibault. *Bioinformatics: from genomes to therapies*, volume 3. 2008. ISBN 9783527312788. doi: 10.1002/9783527619368.ch36.

Thomas Robert Malthus. *An essay on the principle of population: or, A view of its past and present effects on human happiness*. Reeves & Turner, 1888.

Alejandra Manjarrez. " Thinking in the Most Interesting Simplification ". pages 26–29, 2011.

S. S. Mansy and J. W. Szostak. Reconstructing the emergence of cellular life through the synthesis of model protocells. *Cold Spring Harbor Symposia on Quantitative Biology*, 74:47–54, 2009. ISSN 00917451. doi: 10.1101/sqb.2009.74.014.

Franklin Manuel. different types elephant high resolution hd images free. URL http://finehdwallpapers.blogspot.com/2013/01/different-types-elephant-high.html.

Norman Margolus and Tommaso Toffoli. Cellular Automata Machines. *Complex Systems*, 1(May 2014): 967–993, 1987.

David H. Mathews and Douglas H. Turner. Prediction of RNA secondary structure by free energy minimization. *Current Opinion in Structural Biology*, 16(3):270–278, 2006. ISSN 0959440X. doi: 10.1016/j.sbi.2006.05.010.

Mark J Miller, Sindy H Wei, Michael D Cahalan, and Ian Parker. Autonomous T cell trafficking examined in vivo with intravital two-photon microscopy. *Proceedings of the National Academy of Sciences of the United States of America*, 100(5):2604–2609, 2003. ISSN 00278424. doi: 10.1073/pnas.2628040100.

Dusan Misevic, Antoine Frénoy, David P Parsons, and François Taddei. Effects of public good properties on the evolution of cooperation. *Artificial life*, 13:218–225, 2012. doi: 10.7551/978-0-262-31050-5-ch030.

Denis Mollison. Dependence of Epidemic and Population Velocities on Basic Parameters. *Mathematical Biosciences*, 107:255–287, 1991.

Antonio M. Mora, Giovanni Squillero, Alexandros Agapitos, Paolo Burelli, William S. Bush, Stefano Cagnoni, Carlos Cotta, Ivanoe De Falco, Antonio Della Cioppa, Federico Divina, A. E. Eiben, Anna I. Esparcia-Alcázar, Francisco Fernández de Vega, Kyrre Glette, Evert Haasdijk, J. Ignacio Hidalgo, Michael Kampouridis, Paul Kaufmann, Michalis Mavrovouniotis, Trung Thanh Nguyen, Robert Schaefer, Kevin Sim, Ernesto Tarantino, Neil Urquhart, and Mengjie Zhang. Automatic Evolution of Parallel Sorting Programs on Multi-cores. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9028(April), 2015. ISSN 16113349. doi: 10.1007/978-3-319-16549-3.

Alexander J Nicholson and Victor A Bailey. The Balance of Animal Populations. Part I. In *Proceedings of the Zoological Society of London*, volume 105, pages 551–598. Wiley Online Library, 1935.

Martin A. Nowak, Corina E. Tarnita, and Edward O. Wilson. Nowak et al. reply. *Nature*, 471(7339), 2011. ISSN 00280836. doi: 10.1038/nature09836.

E. O. Nowak, Martin; Corina, Tarnita; Wilson. The Evolution of Eusociality. *Nature*, 466(7310):1057–1062, 2010. doi: 10.1038/nature09205.THE.

Laura Nuño de la Rosa. Computing the Extended Synthesis: Mapping the Dynamics and Conceptual Structure of the Evolvability Research Front. *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution*, 328(5):395–411, 2017. ISSN 15525015. doi: 10.1002/jez.b.22741.

Stephen L. Nutt, Philip D. Hodgkin, David M. Tarlinton, and Lynn M. Corcoran. The generation of antibody-secreting plasma cells. *Nature Reviews Immunology*, 15(3):160–171, 2015. ISSN 14741741. doi: 10.1038/nri3795. URL http://dx.doi.org/10.1038/nri3795.

Leslie E Orgel. Prebiotic chemistry and the origin of the RNA world. *Critical Reviews in Biochemistry and Molecular Biology*, 39(2):99–123, 2004. ISSN 1040-9238. doi: 10.1080/10409230490460765. URL http://www.ncbi.nlm.nih.gov/pubmed/15217990{%}5Cnhttp://www.crbmb.com/cgi/content/full/39/2/99.

Ertugrul M. Ozbudak, Mukund Thattal, Han H. Lim, Boris I. Shraiman, and Alexander Van Oudenaarden. Multistability in the lactose utilization network of Escherichia coli. *Nature*, 427(6976):737–740, 2004. ISSN 00280836. doi: 10.1038/nature02298.

Ludo Pagie and Paulien Hogeweg. Evolutionary consequences of coevolving targets. *Evolutionary Computation*, 5(4):401–418, 1997. ISSN 10636560. doi: 10.1162/evco.1997.5.4.401.

Ludo Pagie and Paulien Hogeweg. Individual- and population-based diversity in restriction-modification systems. *Bulletin of Mathematical Biology*, 62(4):759–774, 2000. ISSN 00928240. doi: 10.1006/bulm.2000.0177.

Robin Pearce, Gilbert S Omenn, and Yang Zhang. De novo rna tertiary structure prediction at atomic resolution using geometric potentials from deep learning. *bioRxiv*, pages 2022–05, 2022.

Joel R. Peck and David Waxman. Is life impossible? Information, sex, and the origin of complex organisms. *Evolution*, 64(11):3300–3309, 2010. ISSN 00143820. doi: 10.1111/j.1558-5646.2010.01074.x.

Massimo Pigliucci. Is evolvability evolvable? *Nature Reviews Genetics*, 9(1):75–82, 2008. ISSN 14710056. doi: 10.1038/nrg2278.

Selwyn Quan, J. Christian J. Ray, Zakari Kwota, Trang Duong, Gábor Balázsi, Tim F. Cooper, and Russell D. Monds. Adaptive evolution of the lactose utilization network in experimentally evolved populations of Escherichia coli. *PLoS Genetics*, 8(1):17–22, 2012. ISSN 15537390. doi: 10.1371/journal.pgen.1002444.

Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21(4):25–34, 1987. ISSN 00978930. doi: 10.1145/37402.37406. URL http://portal.acm.org/citation.cfm?doid=37402.37406.

Thomas Riggs, Adrienne Walts, Nicolas Perry, Laura Bickle, Jennifer N. Lynch, Amy Myers, Joanne Flynn, Jennifer J. Linderman, Mark J. Miller, and Denise E. Kirschner. A comparison of random vs. chemotaxis-driven contacts of T cells with dendritic cells during repertoire scanning. *Journal of Theoretical Biology*, 250(4):732–751, 2008. ISSN 00225193. doi: 10.1016/j.jtbi.2007.10.015.

Kepa Ruiz-Mirazo, Carlos Briones, and Andrés De La Escosura. Prebiotic systems chemistry: New perspectives for the origins of life. *Chemical Reviews*, 114(1):285–366, 2014. ISSN 00092665. doi: 10.1021/cr2004844.

Jacob Rutten, Paulien Hogeweg, Guillaume Beslon, Jacob Rutten, Paulien Hogeweg, and Guillaume Beslon. Evolution of mutator populations in constant environments. In *2nd EvoEvo Workshop, satellite workshop of CCS2016*, 2016.

Biswajit Samanta and Gerald F Joyce. SHORT REPORT A reverse transcriptase ribozyme. pages 1–10, 2017. ISSN 2050084X. doi: 10.7554/eLife.31153.

Nicholas J. Savill, Pejman Rohani, and Paulien Hogeweg. Self-reinforcing spatial patterns enslave evolution in a host-parasitoid system. *Journal of Theoretical Biology*, 188(1):11–20, 1997. ISSN 00225193. doi: 10.1006/jtbi.1997.0448.

Willem Scharloo. CANALIZATION : GENETIC AND DEVELOPMENTAL ASPECTS. pages 65–93, 1991.

J. P. Schrum, T. F. Zhu, and J. W. Szostak. The Origins of Cellular Life. *Cold Spring Harbor Perspectives in Biology*, 2(9):a002212–a002212, 2010. ISSN 1943-0264. doi: 10.1101/cshperspect.a002212. URL http://cshperspectives.cshlp.org/lookup/doi/10.1101/cshperspect.a002212.

Erik a Schultes, David P Bartel, Erik a Schultes, and David P Bartel. One Sequence, Two Implications of New for the Ribozyme Ribozymes: Emergence of new ribozyme folds. 289(5478):448–452, 2000. ISSN 00368075. doi: 10.1126/science.289.5478.448.

P. Schuster, W. Fontana, P. F. Stadler, and I. L. Hofacker. From sequences to shapes and back: A case study in RNA secondary structures. *Proceedings of the Royal Society B: Biological Sciences*, 255 (1344):279–284, 1994. ISSN 14712970. doi: 10.1098/rspb.1994.0040.

Peter Schuster. How to search for RNA structures Theoretical concepts in evolutionary biotechnology. *Journal of Biotechnology*, 41(2-3):239–257, 1995. ISSN 01681656. doi: 10.1016/0168-1656(94)00085-Q.

Jared Seifter and James A. Reggia. Lambda and the Edge of Chaos in Recurrent Neural Networks. 71: 55–71, 2014. ISSN 10645462. doi: 10.1162/ARTL. URL http://arxiv.org/abs/1411.7267.

Y. Setty, A. E. Mayo, M. G. Surette, and U. Alon. Detailed map of a cis-regulatory input function. *Proceedings of the National Academy of Sciences*, 100(13):7702–7707, 2003. ISSN 0027-8424. doi: 10.1073/pnas.1230759100. URL http://www.pnas.org/cgi/doi/10.1073/pnas.1230759100.

Premal Shah, Yang Ding, Malwina Niemczyk, Grzegorz Kudla, and Joshua B. Plotkin. Rate-limiting steps in yeast protein translation. *Cell*, 153(7):1589–1601, 2013. ISSN 00928674. doi: 10.1016/j.cell.2013.05.049. URL http://dx.doi.org/10.1016/j.cell.2013.05.049.

Bruce A. Shapiro. An algorithm for comparing multiple RNA secondary structures. *Bioinformatics*, 4 (3):387–393, 1988. ISSN 13674803. doi: 10.1093/bioinformatics/4.3.387.

R. W. Sheldon and S. R. Kerr. THE POPULATION DENSITY OF MONSTERS IN LOCH NESS. *Limnology & Oceanography*, pages 796–798, 1972.

Herbert A Simon. The sciences of the artificial. *Cambridge, MA*, 1969.

Robert A. Skipper, Jr. The Heuristic Role of Sewall Wright's 1932 Adaptive Landscape Diagram. *Philosophy of Science*, 71(5):1176–1188, 2004. ISSN 0031-8248. doi: 10.1086/425240.

M Slatkin and D S Wilson. Coevolution in structured demes. *Proceedings of the National Academy of Sciences of the United States of America*, 76(4):2084–7, 1979. ISSN 0027-8424. doi: DOI10.1073/pnas.76.4.2084. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=383539{&}tool=pmcentrez{&}rendertype=abstract.

Joan E Strassmann, Robert E Page Jr, Gene E Robinson, and Thomas D Seeley. Kin selection and eusociality. *Nature*, 471(7339):E5, 2011.

Eörs Szathmary and Laszlo Demeter. Group Selection of Early Replicators and the Origin of Life. *Journal of Theoretical Biology*, 128:463–486, 1987.

Natalia Szostak, Szymon Wasik, and Jacek Blazewicz. Hypercycle. *PLoS Computational Biology*, 12(4): 1–13, 2016. ISSN 15537358. doi: 10.1371/journal.pcbi.1004853.

Nobuto Takeuchi and Paulien Hogeweg. Error-threshold exists in fitness landscapes with lethal mutants. *BMC Evolutionary Biology*, 7:1–12, 2007. ISSN 14712148. doi: 10.1186/1471-2148-7-15.

Nobuto Takeuchi and Paulien Hogeweg. Evolution of complexity in RNA-like replicator systems. *Biology Direct*, 3:1–20, 2008. ISSN 17456150. doi: 10.1186/1745-6150-3-11.

Nobuto Takeuchi and Paulien Hogeweg. Multilevel selection in models of prebiotic evolution II: A direct comparison of compartmentalization and spatial self-organization. *PLoS Computational Biology*, 5 (10), 2009a. ISSN 1553734X. doi: 10.1371/journal.pcbi.1000542.

Nobuto Takeuchi and Paulien Hogeweg. Multilevel selection in models of prebiotic evolution II: A direct comparison of compartmentalization and spatial self-organization. *PLoS Computational Biology*, 5 (10), 2009b. ISSN 1553734X. doi: 10.1371/journal.pcbi.1000542.

Nobuto Takeuchi, Paulien Hogeweg, and Eugene V. Koonin. On the origin of DNA genomes: Evolution of the division of labor between template and catalyst in model replicator systems. *PLoS Computational Biology*, 7(3), 2011. ISSN 1553734X. doi: 10.1371/journal.pcbi.1002024.

Nobuto Takeuchi, Kunihiko Kaneko, and Paulien Hogeweg. Evolutionarily stable disequilibrium: Endless dynamics of evolution in a stationary population. *Proceedings of the Royal Society B: Biological Sciences*, 283(1830), 2016. ISSN 14712954. doi: 10.1098/rspb.2015.3109.

Nobuto Takeuchi, Paulien Hogeweg, and Kunihiko Kaneko. The origin of a primordial genome through spontaneous symmetry breaking. *Nature Communications*, 8(1), 2017. ISSN 20411723. doi: 10.1038/s41467-017-00243-x. URL http://dx.doi.org/10.1038/s41467-017-00243-x.

Kirsten ten Tusscher. *Mathematics for biologists*. 2016. ISBN 0121982505. URL http://theory.bio.uu.nl/sb/wiskunde/WiskundeDictaatSysBio2016.pdf.

David Tilman. Competition and Biodiversity in Spatially Structured Habitats. *Ecology*, 75(1):2–16, 1994.

Herbert H Tsang and Kay C Wiese. SARNA-Predict: Accuracy Improvement of RNA Secondary Structure Prediction Using Permutation-Based Simulated Annealing. 7(4):727–740, 2010. doi: 10.1109/TCBB.2008.97.

Mark Twain. *King Leopold's soliloquy: A defense of his Congo rule*. LeftWord Books, 1905.

J. D. Van der Laan and P. Hogeweg. Predator-prey coevolution: Interactions across different timescales, 1995. ISSN 14712970.

D J van der Post and P Hogeweg. Learning what to eat: Studying inter-relations between learning, grouping, and environmental conditions in an artificial world. *Cellular Automata, Proceedings*, 3305: 492–501, 2004. ISSN 03029743. doi: 10.1007/978-3-540-30479-1_51.

Daniel J. Van Der Post and Paulien Hogeweg. Resource distributions and diet development by trial-and-error learning. *Behavioral Ecology and Sociobiology*, 61(1):65–80, 2006. ISSN 03405443. doi: 10.1007/s00265-006-0237-6.

Daniel J. van der Post and Paulien Hogeweg. Diet traditions and cumulative cultural processes as side-effects of grouping. *Animal Behaviour*, 75(1):133–144, 2008. ISSN 00033472. doi: 10.1016/j.anbehav.2007.04.021.

A. van Doorn and J. Heringa. The ontogeny of a dominance hierarchy in colonies of the BumblebeeBombus terrestris (Hymenoptera, Apidae). *Insectes Sociaux*, 33(1):3–25, 1986. ISSN 0020-1812, 1420-9098. doi: 10.1007/BF02224031. URL http://link.springer.com/article/10.1007/BF02224031{%}5Cnhttp://link.springer.com/article/10.1007/BF02224031{#}page-1.

M. J.A. Van Hoek and P. Hogeweg. In silico evolved lac operons exhibit bistability for artificial inducers, but not for lactose. *Biophysical Journal*, 91(8):2833–2843, 2006. ISSN 00063495. doi: 10.1529/biophysj.105.077420. URL http://dx.doi.org/10.1529/biophysj.105.077420.

Cor van Honk and Pauline Hogeweg. The ontogeny of the social structure in a captive Bombus terrestris colony. *Behavioral Ecology and Sociobiology*, 9(2):111–119, 1981. ISSN 03405443. doi: 10.1007/BF00293582.

Erik Van Nimwegen. *The Statistical Dynamics of Epochal Evolution*. Number september. 1999. ISBN 9039321620.

Peter M. Visscher, Naomi R. Wray, Qian Zhang, Pamela Sklar, Mark I. McCarthy, Matthew A. Brown, and Jian Yang. 10 Years of GWAS Discovery: Biology, Function, and Translation. *American Journal of Human Genetics*, 101(1):5–22, 2017. ISSN 15376605. doi: 10.1016/j.ajhg.2017.06.005. URL http://dx.doi.org/10.1016/j.ajhg.2017.06.005.

Samuel von der Dunk, Enrico Colizzi, and Paulien Hogeweg. Evolutionary Conflict Leads to Innovation: Symmetry Breaking in a Spatial Model of RNA-Like Replicators. *Life*, 7(4):43, 2017. ISSN 2075-1729. doi: 10.3390/life7040043. URL http://www.mdpi.com/2075-1729/7/4/43.

John Von Neumann, Arthur W Burks, and Others. Theory of self-reproducing automata. *IEEE Transactions on Neural Networks*, 5(1):3–14, 1966.

Renske M A Vroomans, Athanasius F M Marée, Rob J. de Boer, and Joost B. Beltman. Chemotactic Migration of T Cells towards Dendritic Cells Promotes the Detection of Rare Antigens. *PLoS Computational Biology*, 8(11), 2012. ISSN 1553734X. doi: 10.1371/journal.pcbi.1002763.

Sake Van Wageningen, Patrick Kemmeren, Philip Lijnzaad, Thanasis Margaritis, Joris J Benschop, Inês J De Castro, Dik Van Leenen, Marian J A Groot Koerkamp, Cheuk W Ko, Antony J Miles, Nathalie Brabers, Mariel O Brok, Tineke L Lenstra, Dorothea Fiedler, Like Fokkens, Rodrigo Aldecoa, Eva Apweiler, Katrin Sameith, Loes A L Van De Pasch, Sander R Van Hooff, V Linda, Nevan J Krogan, Berend Snel, and Frank C P Holstege. Functional Overlap and Regulatory Links Shape Genetic Interactions between Signaling Pathways. 143(6):991–1004, 2011. doi: 10.1016/j.cell.2010.11.021.van.

Andreas Wagner. Causal Drift, Robust Signaling, and complex disease. *PLoS ONE*, 10(3):8–13, 2015. ISSN 19326203. doi: 10.1371/journal.pone.0118413.

Godfrey A. Walters and David K. Smith. Evolutionary Design Algorithm for Optimal Layout of Tree Networks. *Engineering Optimization*, 24(4):261–281, 1995. ISSN 0305-215X. doi: 10.1080/03052159508941193. URL http://www.tandfonline.com/doi/abs/10.1080/03052159508941193.

Charles W Whitfield, Susanta K Behura, Stewart H Berlocher, Andrew G Clark, J Spencer Johnston, Walter S Sheppard, Deborah R Smith, and Andrew V Suarez. Thrice Out of Africa : Ancient and Recent Expansions of the Honey Bee, Apis mellifera. *Science*, 314(October):642–645, 2006. ISSN 10959203. doi: 10.1126/science.1132772.

D S Wilson. A theory of group selection. *Proceedings of the National Academy of Sciences of the United States of America*, 72(1):143–146, 1975. ISSN 0027-8424. doi: 10.1073/pnas.72.1.143. URL http://www.pnas.org/content/72/1/143.abstract{%}255Cnhttp://www.pnas.org/content/72/1/143.full.pdf{%}255Cnhttp://www.pnas.org/cgi/content/abstract/72/1/143.

Stephen Wolfram. *A new kind of science*, volume 5. Wolfram media Champaign, IL, 2002.

Patrick Wong, Stephanie Gladney, and J. D. Keasling. Mathematical model of the lac operon: Inducer exclusion, catabolite repression, and diauxic growth on glucose and lactose. *Biotechnology Progress*, 13(2):132–143, 1997. ISSN 87567938. doi: 10.1021/bp970003o.

Sewall Wright. *Evolution: selected papers*. University of Chicago Press, 1986.

Vero C Wynne-Edwards. Self-regulating systems in populations of animals. *Science*, 147(3665):1543–1548, 1965.

A. H. Yona, Y. S. Manor, R. H. Herbst, G. H. Romano, A. Mitchell, M. Kupiec, Y. Pilpel, and O. Dahan. Chromosomal duplication is a transient evolutionary solution to stress. *Proceedings of the National Academy of Sciences*, 109(51):21010–21015, 2012. ISSN 0027-8424. doi: 10.1073/pnas.1211150109. URL http://www.pnas.org/cgi/doi/10.1073/pnas.1211150109.

Dominique Zander, Daniel Samaga, Ronny Straube, and Katja Bettenbrock. Bistability and Nonmonotonic Induction of the lac Operon in the Natural Lactose Uptake System. *Biophysical Journal*, 112(9):1984–1996, 2017. ISSN 15420086. doi: 10.1016/j.bpj.2017.03.038. URL http://dx.doi.org/10.1016/j.bpj.2017.03.038.

Michael Zuker. On finding all suboptimal foldings of an RNA molecule. *Science*, 244(4900):48–52, 1989.