Lecture Notes on Modelling Micro-Evolution[1]

Adam Prügel-Bennett
NORDITA,Blegdamsvej 17, DK-2100 Copenhagen Ø, Denmark.
adam@nordita.dk

May 29, 1996

[1]These notes are available on http://www.nordita.dk:/~adam/evol-course

Contents

Lecture 1

Applications of Micro-Evolution

These lectures deal with the problem of **micro-evolution**, that is evolution within a species. The aims of these lectures fall into two categories. In the first lecture I will try to motivate the study of micro-evolution by considering three applications. These are **directed molecular evolution**, **Genetic Algorithms** and models of biological systems. In the second and third lectures I will take up the second objective which is to show that at a very deep level micro-evolution is a statistical mechanics problem and can be solved using statistical mechanics techniques. We start this lecture with a short description of some of the common features in models of micro-evolution.

1.1 Micro-Evolution

We will consider the evolution of a population $\mathcal{P}(t)$, whose members are characterized by **chromosomes** or **gene sequences**. We will denote the size of the population by P and the length of gene sequence by N. The genes sequence we will denote by the vector

$$\vec{S}^{\alpha} = (S_1^{\alpha}, S_2^{\alpha}, \cdots, S_N^{\alpha}).$$

We will use Greek letters to label the different members of the population, $\alpha = 1, \ldots, P$ and Latin indices to label the sites of the gene sequence, $i = 1, \ldots, N$. The variables S_i^{α} denote the alleles at site i of member α. The genetic makeup of an individual is known as its **genotype**. The expression of this genetic information is known as the **phenotype**. In nature the phenotype is a result of an enormously elaborate self-organization mechanism. As a consequence small changes in the genotype may lead to significant changes in the phenotype. The **fitness** of an individual depends, of course, on its phenotype. In biology "fitness" is a precisely defined term: the fitness of an individual is proportional to its reproductive capacity. We will use the term less rigidly as any measure of reproductive capacity.

We will consider a variety of models defined by different **fitness function**. Examples of such fitness functions are:

- **Ones-counting.** In this case $S_i^{\alpha} \in \{0, 1\}$ and

$$F_{\alpha} = \sum_{i=1}^{N} S_i^{\alpha}.$$

This is everyone's favorite fitness function. It goes by many names: ones-max, simple additive fitness function, multiplicative fitness landscape, etc. It describes a gene sequence in which the genes are either good or bad and all contributed equally to the fitness.

Up to a (trivial) linear transform the ones-counting fitness function is equivalent to minus the energy of an ideal paramagnet in a constant field with Ising spins $S_i^\alpha \in \{-1, 1\}$. We can also consider the case $S_i^\alpha \in \{-1, 0, 1\}$ as a model for a variable length gene sequence.

- **Random-field paramagnet.** A simple generalization of the above fitness function is to consider the case where the genes have different efficacies. We can model this using $S_i^\alpha \in \{-1, 1\}$ and
$$F_\alpha = \sum_{i=1}^{N} h_i \, S_i^\alpha$$
and where the "fields" h_i are chosen at random from a distribution $P(h_i)$.

- **Spin-glass chain.** This can be considered as a model in which the genes depends also on neighbouring genes. We again have Ising variables but now
$$F_\alpha = \sum_{i=1}^{N} J_i \, S_i^\alpha \, S_{i+1}^\alpha + \sum_{i=1}^{N} h_i \, S_i^\alpha.$$
This adds a new twist because now there are many local-minima to single gene mutations. The spin-glass can be defined in higher dimensions where the barriers between the minima became more pronounced. The "infinite" dimensional spin-glass or SK-model is one of the most studied models of a disordered system. We will come back to the SK-model later in this lecture.

- **Random Energy Model.** Here the fitness for each sequence is assumed to be an independently chosen random number — i. e. the energy landscape is maximally rugged. It can be derived as a limiting of a spin classes in which many spins interact together [1]. Because of the distinction between phenotype and genotype this may not be such a bad model of evolution in biology.

The population
$$\mathcal{P}(t) = (\vec{S}^1(t), \vec{S}^2(t), \ldots, \vec{S}^P(t))$$
evolves due to the repeated application of a set of **evolution operators.** We will concern ourselves with just three

1. **Selection.** The fittest members of the population are reproduced preferentially at the expense of the less fit members of the population.
2. **Mutation.** The gene sequences are altered by small random mutations. For example, with Ising genes, $S_i^\alpha \in \{-1, 1\}$, each gene may be "flipped", $S_i^\alpha \longrightarrow -S_i^\alpha$, with some small probability.
3. **Sexual Recombination/Crossover.** The gene sequences are combined together to form new gene sequences. This is usually done in pairs; taking two parents and creating two adults.

The focus of this course will be to study the evolutionary dynamics
$$\mathcal{P}(t) \overset{selection}{\longrightarrow} \mathcal{P}^s(t) \overset{mutation}{\longrightarrow} \mathcal{P}^{s,m}(t) \overset{crossover}{\longrightarrow} \mathcal{P}^{s,m,c}(t) = \mathcal{P}(t+1)$$
starting from some initial population $\mathcal{P}(0)$, where the superscripts s, m and c denotes the populations after selection, mutation and crossover respectively. In the second and third lectures we will consider formal models of evolution and will develop a formalism for solving them. In the rest of this lecture we will give a motivation for why we should study such models.

Evolution is full of many other fascinating questions. These include the problems of **macro-evolution**: why did the dinosaurs die out?; how did mankind evolve? Other hot topics include the question of the how did life begin. These, however, I will not discuss at all.

1.2 Directed Molecular Evolution.

Directed molecular evolution (DME), sometimes simply called directed evolution, is a new experimental technique for developing enzymes, pharmaceuticals and other macro-molecules using "evolution in a test tube". The novel feature of DME is that evolution happens at the molecular level and not at the organism level — this requires considerable finesse on the part of the experimenter to achieve. There are two variants: evolution of naked DNA or RNA, involving very large numbers of molecules; and evolution of proteins using cell cultures as the reproductive machinery. Before discussing DME in more detail I will first give a brief overview of current techniques for developing macro-molecules (with apologies to biochemists).

1.2.1 Background

The active molecules of life are the biological **catalysts** which facilitates different chemical reactions, usually in a very specific way. The great majority of these are **protein enzymes** — chains of amino acids typically containing 100-1000 units. However, nature also uses RNA as a catalyst, known as **ribozymes** — this was discovered by Thomas Cech and Sidney Altman.

As we all now know proteins sequences are coded in DNA which is first copied to RNA and then decoded by **ribosomes** into proteins. These then curl up into complex three dimensional structures that perform the catalysis. We know a great deal about about protein **primary structure** (i.e. the sequence of amino acids — the first sequence to be decoded was insulin with less than 60 amino acids by Frederick Sanger and his collaborators in 1956; today deciphering primary sequences is a routine procedure as exemplified by the human genome project), however, we know relatively little about the higher order structure which determines the functionality of the enzyme. The precise three dimensional structure of only around 1000 proteins are known. It is a long, laborious task to obtain this information involving purification and crystallization of the protein, followed by complex crystallographic analysis. *Ab initio* calculation of the high order structure is known as the **protein folding problem** and remains a daunting and unsolved problem. The situation of RNA is somewhat simpler. To a first approximation RNA folds so that as many base pairs as possible match up — this determines its secondary structure (in proteins the secondary structure is largely determined by hydrogen bonds which are enormously more complex). Finding a good match of base pairs in RNA sequence is not a difficult problem so at least its secondary structure can be calculated to a reasonable good approximation. However, even if we could solve the folding problem fully we would also need to solve the **inverse folding problem** to design macro-molecular catalysts. That is given a desired three dimensional structure what sequence will fold into this structure.

At present the starting point for developing a new enzymes uses far more prosaic methodologies. One approach is to search large "libraries" of enzymes from naturally occurring organisms to see whether any exhibit a desired trait. An alternative approach is to study traditional medicines in the hope of uncovering a naturally occurring therapeutic drug (this approach is discussed in a fascinating *Scientific American* article by Balick and Cox [2]). Once an enzyme has been discovered with a desired trait then the mother organism must be cultured so that the enzyme can be purified. Once identified the DNA coding for the enzyme can be cut out and put into another organism. The enzyme can also be modified to improve its efficiency or expression. This is often done by substituting a different amino acid into the protein sequence and testing the new mutant strain. This whole process is, needless to say, very costly and time consuming.

1.2.2 RNA and DNA Dme

Both DNA and RNA can be readily copied to each other and reproduced in bulk using **polymerase chain reaction** (PCR) enzymes — these include **transciptase** and **reverse transcriptase** and **replicase**. In the replication of DNA or RNA the copying rate will depend on the numbers of copies there already are, therefore the size of the population will grows geometrically. As a consequence it is possible to produce a huge number of molecules. Typically workers use around 10^{13} molecules at a time. The PCR process can be controlled so that a desired number of mutations are introduced during replication. This step is often referred to as **mutagenesis**.

Dme has been used both to make RNA catalysts and to make DNA catalysts. To perform dme it is necessary to interleave the mutagenesis steps with screening steps. The usual why screening is achieved is by preparing a substrate to which "fit" molecules will stick. The less fit molecules can then be washed away. The remaining molecules are then multiplied up using mutagenesis. A nice review article on this type of dme by one of its leading practitioners, Gerald Joyce, can be found in *Scientific American* [3].

1.2.3 Protein Dme

Directed molecular evolution with proteins is different from that for DNA and RNA because there are no reverse-ribosomes in nature, nor are there any enzymes that directly replicate proteins. However, dme has be accomplished by using bacteria as replicators and test vessels for the proteins. An example of this procedure was published very recently by Jeffrey Moore and Frances Arnold [4]. Here the authors used dme to increase the efficiency of an esterase enzyme 50–60 times. They did this by putting mutated copies of DNA encoding the enzyme into *E. coli* cells which they then cultured. They tested the cell cultures by allowing a test chemical to permeate through the cell membrane and react with the enzyme. The products of the reaction coloured the cell colonies allowing

the good mutations to be easily identified. Using this process it was possible to test thousands of cell colonies at a time. The best cell colonies could then be cultured and the DNA extracted. This could then be replicated using the same PCR mechanism as described above, again introducing mutations during the replications. The new DNA could then be put back into *E. coli* cells and the process repeated. In terms of the numbers involved this process is much more conservative than the RNA/DNA directed evolution but it represents a substantial improvement over the traditional techniques for developing protein enzymes

As well as mutagenesis it is also possible to perform **gene recombination** *in vitro*. This was also carried out by Moore and Arnold. To achieve this the DNA is cleaved using special enzymes that act at certain **restriction sites**. These can be reassembled using other enzymes. In the Moore and Arnold experiment two different mutants where chosen for recombination (i. e. pairwise recombination) so that the reassembled DNA were random combinations of the two parents.

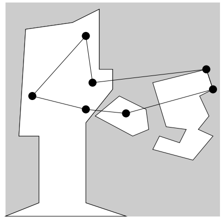
The paper by Moore and Arnold [4] gives many more details and a beautiful analysis as to what the evolution did in terms of amino acid substitution.

1.3 Genetic Algorithms.

A very different example of an application of micro-evolution is provided by **Genetic Algorithms** (GAs). These are search techniques for finding good solution to hard optimization problems. They work by evolving a population of solutions. Before describing GAs, I will first give an overview of optimization problems and other techniques for solving them.

1.3.1 Optimization Problems

The classic example of an optimization problem is the **travelling salesman problem** (TSP): "The salesman must make a **tour** of n-cites returning to the city he started in. The problem is to find the shortest tour."



	Kø	He	Od	Ve	Es	Ål	År
Kø		20	80	120	157	210	140
He	20		95	121	159	200	130
Od	80	80		31	65	100	40
Ve	120	120	31		40	95	25
Es	157	159	65	40		98	45
Ål	210	200	100	95	98		40
År	140	130	40	25	45	40	

The length of the tour can be regarded as a **cost function** which must be minimized.

There are a total of $n!/(2n)$ distinct tours. For 100 cities this is equal to 4.66631×10^{155} different possible tours. There is no known efficient (polynomial in n) algorithms for finding the shortest tour. (For Euclidean TSPs there are divide and conquer algorithms capable of doing a few hundred cities). Although the problem as stated is rather artificial it has applications to cable networks, pipelines, printed circuit boards layout, micro-chips, etc.

The TSP is in a class of problems known as **NP-hard**. For these problems there are no known polynomial time algorithms for solving all instances of the problem. It is thus necessary to resort to **heuristic** techniques, i. e. approximate techniques which find good solutions but not necessarily the best. Before, discussing heuristic search techniques we make a detour to discuss what NP-hard really means.

NP-Completeness

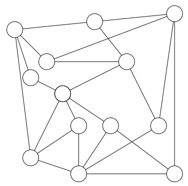
"A little knowledge is a dangerous thing;"

NP-hard is closely related to **NP-completeness**. This concept allows problems to be put into two classes; tractable problems (those for which there exists a polynomial time algorithm) and intractable problems (those for which there are believed to be no polynomial time algorithm). However, the concept of NP-completeness is rather subtle, requiring a number of definitions and concepts to be established before we can define what it means. It can also be a misleading concepts so it is worth understanding it properly.

NP-completeness is a category that distinguishes **decision problems**. It is a subset of a class of problems known as NP and is related to another class of problems known as P. We begin by define what all these terms mean. A decision problem consists of input which is a set of instances of some characteristic size n and an output which is a true/false decision depending of whether the instance fulfills a certain criteria. Examples of decision problems are

Compositeness
INPUTS: x are n-bit number
OUTPUT: True if x is factorizable, false otherwise
e. g. $x = 239812014798221$

Three-Colouring
INPUTS: G are graphs with n-nodes
OUTPUT: True if G can be coloured by three colours where all nodes (vertices) which share an edge are coloured differently, false otherwise.
e. g. G is the graph



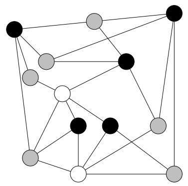
Satisfiability (SAT)
INPUT: F is a Boolean expression of n variables $(x_1, \cdots, x_n)$
OUTPUT: True if the variables of F can be chosen so that F is true, false otherwise.
e. g. $F = (x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge x_4 \wedge x_5) \vee x_3$

The Subset-Sum Problem
INPUTS: A set $\mathcal{S}$ of n integers and another integer c.

OUTPUT: True if there exists a subset of $\mathcal{S}$ whose sum equals c, false otherwise.
e. g. $\mathcal{S} = \{5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610\}$ and $c = 1001$.

The Travelling Salesman (Decision) Problem
INPUT: A set of n cities with the matrix of distances connecting the cities and a distance l.
OUTPUT: True if there is a tour shorter than or equal to l, false otherwise.

We are now in a position to define the complexity class NP (non-deterministic polynomial time algorithms). This is the class of decision problems for which there exists a **witness** which can be used to verify an instance of the problem in polynomial time. All the problems defined above are in NP. Example of witnesses would be the number 15485863 for the composite problem which can be shown to be a factor of 239812014798221 by simple division. The coloured graph



acts as a witness for the three colouring problem as it can be checked in polynomial time. Similarly for SAT and the other problems any solution to the problem can be checked in polynomial time and therefore acts as a suitable witness. A subset of complexity class NP is the set of problems P for which a witness can be found in polynomial time. Examples of problems in this class are squareness (is the n-digit number a square) or matrix singularity (is the $n \times n$ matrix singular).

There is another class of problems in NP which are called NP-complete and have the striking property:

if an NP-complete problem can be solved in polynomial time then all problems in the class NP can also be solved in polynomial time.

In 1971 Steven Cook showed that SAT was an NP-complete problem. That is, if there exists a polynomial time algorithm for determining whether an arbitrary Boolean function F of n variables can be satisfied then all decision problems in the class NP can be solved in polynomial time — this is known as Cook's Theorem. The proof of this is not hard but it is certainly not an obvious result — we will not give the proof of this result.

A consequence of the definition of NP-completeness is that if any problem can be used to solve an NP-complete problem then that too is an NP-complete problem. More accurately, suppose we have a known NP-complete problem P_a and we have a second decision problem P_b, then if there is a polynomial algorithm which can **transform** every instance of P_a, into an instance of P_b in such a way that the output of P_b determines P_a then P_b must also be NP-complete. It turns out that this transform from one problem onto another is relatively easy and there are now known to be over 10 000 problems which are NP-complete. These include a great many of the classic optimization problems: 3-colourablity, the subset-sum problem, the travelling salesman (decision) problem, Hamiltonian cycles, the knapsack problem, etc. Interestingly, compositionality or primality has not been shown to be in either P or NP although it is widely believed to be in P (there are "random algorithms" which can determine primality to any degree of certainty in polynomial time).

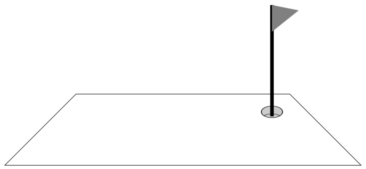
NP-hard problem are not decision problems but they are problems which if they could be solved in polynomial time would mean that an NP-complete problem was in P. The TSP is an obvious example of this, other examples include minimal graph colouring, graph partitioning, time-tabling, etc. Because NP-complete problems and NP-hard problems comprise such a large set of problems, for none of which there is known to exist an efficient (i. e. polynomial time) algorithm it is strongly believed that no such algorithms exist. Of course, if an algorithm of degree 100 was found to solve the TSP then it would mean P=NP, but it might not significantly change the world.

WARNING: NP-completeness or NP-hardness does not imply that the problem is typically hard! (What we mean by typical depends, of course, on how we select the instances.) It just means there exists some instances which are hard. For many problems there are **pseudo-polynomial time** algorithms which for many instances will solve the problem in polynomial time. For example, in the subset-sum problem there are pseudo-polynomial time algorithms which solves the problem in polynomial time provided the members of $\mathcal{S}$ are chosen form a compact distribution; only if the members of $\mathcal{S}$ grow exponentially will the algorithm fail to converge in polynomial time. Thus knowing a problem is NP-hard does not mean it is necessarily difficult.

1.3.2 Why are Problems Hard?

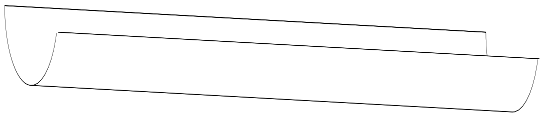
Problems become hard in high dimensions for a number of reasons.

There can be little gradient information to direct the search: this is often referred to as a **golf course potential**

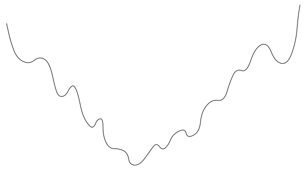


This is a common problem in training multi-layer perceptrons.

In high dimension there are typically *different length scales* to contend with. This makes it difficult to chose an appropriate step sizes for the search.



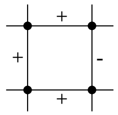
There maybe **many local minima**.



The archetypal model with many local minima is the spin-glass defined by the energy function

$$E = -\sum_{ij} J_{ij}\, S_i\, S_j$$

where S_i are Ising variables and the couplings $J_{ij} = J_{ji}$ are random variables. The couplings remain fixed and are known as **quenched variables**. In finite dimensions J_{ij} is non zero only for neighbouring bonds. There are an exponential number of local minima due to **frustration**



(around any closed plaquette with an odd number of negative bonds it is not possible to choose the spins so as to satisfy all the bonds). The finite dimensional spin-glass is unsolved (except in one-dimension where it is trivial). Note that the dimensionality of the search space in 2^N and not related to the dimensionality of spin-glass model. The "infinite dimensional" spin-glass where all spins are coupled to each other is solved it is known as the Sherrington-Kirkpatrick (SK) model. Much of our intuition about systems with quenched disorder and many local minima comes from the SK model.

Finding the grounds state of a spin-glass in more than two dimensions is an NP-hard problem. (In one and two dimensions there are still an exponential number of local minima but there are efficient algorithms for finding the ground state energy nevertheless). Many local minima are typical of very many NP-hard problems such as the TSP, which also have quenched disorder and frustration.

Optimization problems can be regarded as (disordered) statistical mechanics problems where the cost function is treated as the energy of the system. In some cases this allows the statistical properties of the problem to be calculated exactly. A good discussion of this approach is given in part 2 of *Spin-Glass Theory and Beyond* by Mézard and Parisi and Virasoro [5].

1.3.3 Other Heuristic Search Techniques

Hill Climbing or Greedy Algorithms

The simplest strategy to search for an optimal solution is to do **hill climbing** (on the fitness landscape or hill descent on the cost/energy landscape[1]). This is achieved by

[1]Due to a bad education I have acquired the annoying habit of talking about both hill climbing and local minima — the reader is required to make the appropriate global gauge transformations where necessary.

trying an **elementary move**, i. e. altering the current solution in some small way, and accepting the move if it reduces the energy (increases the fitness) — for this reason such algorithms are often referred to as **greedy algorithms**. The algorithm will find a **local optima** of the fitness but not the **global optima** — the algorithm is typically very fast. One can restart the search from a new starting position to try to find a better solution. This strategy can be quite effective particularly for small problems, however, for large systems it is usually doomed to failure. This is illustrated by the SK model where the average ground state energy is known to be $-0.76N$ (where N is the number of spins). The typical local minima has an energy $-0.70N$ with fluctuations of order 1. Thus for large N one cannot hope to find a good solution by repeated hill climbing.

Stochastic Hill Climbing or Simulated Annealing.

To escape from local minima one strategy is to allow uphill moves with a small probability. A sophisticated way to do this is to treat the cost function as an energy function and choose a dynamics (acceptance probability) which satisfies **detailed balance** – this is called **Simulated Annealing**. This ensures that (asymptotically) the system visits all the states with a Boltzmann probability $\exp(-\beta E)$. This dynamics is exactly what you would usually do in a **Monte Carlo simulation**. As the temperature is reduced (β increases) the time spent in the low energy states increases exponentially thus compensating for the rarity of low lying states. The catch is that the system must be equilibrated. For disordered systems there is typically a **freezing temperature** beyond which it takes an exponentially long time to equilibrate the system. To achieve a good search one must choose the **annealing schedule** so that most of the simulation time is spent around the critical temperature. This temperature is usually characterized by a peak in the specific heat, which can be used to guide the annealing schedule.

Simulated annealing has received considerable attention. It is very well understood in a large range of problems and it is relatively easy to implement (see, for example, the second edition of Numerical Recipes [6]). In almost all practical situations simulated annealing has been found to out perform Genetic Algorithms, although this might just be a coincidence.

Designer Heuristics.

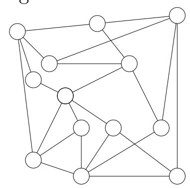
For the well known optimization problems there are often hand-crafted search strategies that are often the method of choice for these problems. These may involve some clever extension of the move set. Sometimes these can be incorporated into either simulated annealing or a Genetic Algorithm.

1.3.4 Genetic Algorithms

Genetic Algorithms are another heuristic technique for solving optimization problems, which work by evolving populations of solutions. They where introduce and popularized by John Holland [7] over twenty years ago. The classic book on the subject is by David Goldberg [8], although now there are a huge number of books on the subject which may well be better. A newer book which discussed GAs in a broad context is by Melanie Mitchell [9]. There is a huge research literature on Genetic Algorithms, a lot of this is published in conference proceedings, e. g. *Foundations of Genetic Algorithms* (Foga)

and the *International Conference on Genetic Algorithms*, although there is now also a journal, *Evolutionary Computation*, on GAs. However, in this course I will largely ignore this literature.

We illustrate how Genetic Algorithms work by considering three colouring. The problem is to colour a graph G consisting of N nodes with three colours.



We wish to minimize the cost function, which is the number of nodes which shares a common vertex and are coloured the same. If we represent the colour at node i by $S_i \in \{1, 2, 3\}$ then a solution to the problem can be represented by a vector $\vec{S}^\alpha$ where α labels the different solutions. A suitable fitness function to describe this problem is

$$F_\alpha = -\sum_{i,j} J_{ij}\, \delta(S_i^\alpha, S_j^\alpha)$$

where

$$\delta(S_i^\alpha, S_j^\alpha) = \begin{cases} 1 & \text{if } S_i^\alpha = S_j^\alpha \\ 0 & \text{otherwise} \end{cases}$$

and

$$J_{ij} = \begin{cases} 1 & \text{if there is an edge from node } i \text{ to node } j \\ 0 & \text{otherwise} \end{cases}$$

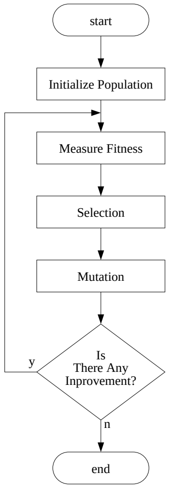
(This fitness function is just minus the energy for the 3-state Potts anti-ferromagnet on a random lattice.)

In the classic GA a population is generated at random and acted upon iteratively by the three evolution operators: selection, mutation and crossover. The algorithms is sketched in figure 1.1. To perform selection a weight is assigned to each members of the population depending on its fitness and then a new population is created by selecting members of the old population in proportion to their weight. There are a number of ways of doing this and we will discuss the details in the next lecture. Selection has an important deleterious effect: it correlates the population. Since the fitter members of the population are chosen more than once their genes will begin to dominate the population. This effect is often missed in theoretical treatments because it is a finite population effect. For large populations the correlation will be very small at each generation, but it is accumulative. We will therefore need to learn how to treat finite populations.

Mutation is performed by considering each node in the population and with a small probability changing its colour. If the mutation rate is too high then mutation will almost always be deleterious. As a rule of thumb, the optimal mutation rate tends to be around one mutation per sequence.

Crossover is the most important operator in a GA — it is what makes Genetic Algorithms unique. In the graph colouring problem crossover can be accomplished by taking two sequences, cutting them at a random point and swap the two ends.

$$\left.\begin{array}{c} (102120|2112121) \\ (002012|0102212) \end{array}\right\} \longrightarrow \left\{\begin{array}{c} (102120|0102212) \\ (002012|2112121) \end{array}\right.$$



Initialize population: generate randomly coloured graphs.

Measure Fitness: count number of unsatisfied edges.

Selection: multiply up the fittest members of the population keeping the total size of the population fixed.

Mutation: with some small probability change the colour of the nodes.

Crossover: take two strings $\vec{S}^{\alpha}$ and $\vec{S}^{\beta}$ and cross them at some randomly chosen site.

Figure 1.1: Sketch of a GA for the graph colouring problem.

This is known as **single-point crossover**. The effects on the graphs are illustrated in figure 1.2. We note that there is a **interface cost or energy** associated with this process because it break links in the graphs — this may not always be the case, for example, there is no interface energy for ones-counting. In graph colouring the interface energy can be reduced by ordering the string so that cutting the string brakes a relatively small number of bonds. To a first approximation the interface energy will be proportional to the fitness of graphs, the number of broken bonds, and to $1 - q$, where q is the correlation between the graphs. The population will correlate over time due to selection, this will reduce the interface energy, but also reduce the mixing effect of crossover. The benefit of crossover is that it adds a new move to the move-set which radically changes the solutions but without a great loss in energy (provided the interface cost is not too high). Consequently the population can explore the search space much more globally and is less likely to get stuck in a local minima.

A Genetic Algorithm will run out of steam when the population becomes too correlated. There are strategies to try to prevent this from happening to quickly. One popular approach is evolve populations in spatially distributed islands. The islands only interbreed weakly so that the populations remain relatively diverse. This has additional nice feature in that it is very easily to put on a parallel computer.

1.4 Modelling Biological Evolution

Evolution in biology is, of course, much more complicated than the previous examples we have seen. Consequently models of biological evolution are at best caricatures of reality

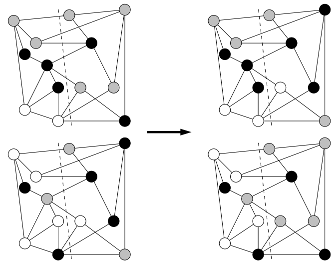


Figure 1.2: Illustration of crossover for the graph colouring problem.

and can be misleading. In this section we first discuss a few of the factors which makes biology so complicated and then give a very brief and very patchy discussion of some popular models of biological micro-evolution.

1.5 Problems of Biology

Some of the things which makes biology so complicated are:

- **Genotype/Phenotype Dichotomy.** The genetic information or genotype of an organism expresses itself as the phenotype through a hugely complex self-organization mechanism. Selection occurs on the phenotype while mutation and selection occurs at the level of the genotype. These leads to very non-trivial phenomena (Richard Dawkins has explored this dichotomy in some detail [10, 11]).

- **Fitness Landscapes.** In biology mutation occurs on DNA bases; this can result in one amino acid being substituted for another which may radically or subtly change the configuration of a protein. This in turn may change how a chemical reaction is catalysed, which may cause a small change in the organism or drastically alter its development. The fitness landscape that describes such a highly interacting system is clearly very complex. To make matters more complicated selection can be a noisy process so that the theoretically fittest individual may not be one which produces the most offspring.

- **Co-Evolution.** A population rarely evolves in a static environment, it rather evolves in unison with other evolving organisms. Thus the fitness function may change over time. Co-evolution is believed to increase the rate of evolution. In particular mutualistic co-evolution (e. g. flowers and pollinating insects) and antagonistic co-evolution (e. g. predators and prey) are believed to be examples of accelerated evolution. This ideal was used by W. Dan Hillis in evolving a sorting network using a GA. To reduce the evaluation time a finite test set was used to measure the fitness, however, this lead to a relatively poor performance. Hillis then allowed the test set to evolve in time to be as difficult as possible. Doing so the GA found a much improved solution to the problem.

- **Biology is messy.** The simple picture of selection, mutation and sexual recombination or crossover is far from describing what really goes on. For example, organisms usually carry two sets of genes (**diploidy**) only one of which is expressed. This is not a strict rule and there some organisms which only carry around one set of genes **haploidy**. Yet nature is truly diverse and there are a large number of variants **Haplodiploidy, Polyploidy, Cyclical Parthenogenesis**, etc. In addition organisms have evolved an incredible mechanism for rearranging their genetic material — these clever tricks our only now beginning to be understood.

1.5.1 Models of Micro-evolution in Biology

Here we give only a flavour of the sort of models that have been proposed as models of biology.

- **Neutral Selection.** Some mutations cause no change in the fitness of an individual. These are examples of neutral selection. They are very popular, in part because they are easy to solve, but also because they can act as markers for helping to establish genealogies. Their popularity increased because of the discovery of "junk genes", sections of DNA which are not expressed. Some of this DNA may now be accounted for as part of the machinery for a higher level of organization in the mechanics of genes.

- **Kauffman Model.** Stuart Kauffman has pioneered a lot of research in understanding evolution in rugged fitness landscape. He introduced the Kauffman model (of which there are a number of variants) as a simple model which captures some of the features of a highly interacting system. There have been many computational and theoretical investigations of these models (see, for example, references [12–14]).

- **Multiplicative Fitness Landscapes.** The multiplicative fitness landscape is ones-counting with different name. The genes are taken to be binary $S_i^\alpha \in \{0, 1\}$ and the fitness function is given by

$$F_\alpha = \sum_{i=1}^{N} S_i^\alpha .$$

The members of the next generation are chosen with a probability $p_\alpha = w_\alpha / \sum_\beta w_\beta$, where

$$w_\alpha = (1-s)^{F_\alpha} = \prod_i (1-s)^{S_i^\alpha} .$$

In biology w_α is by definition the fitness hence this is known as a **multiplicative fitness function**. In addition to selection the population also suffers mutation. This is a popular model for the evolution of an asexual population [15]. There is no reason for not considering crossover as well.

- **Muller's Ratchet.** This is a special case of evolution on a multiplicative fitness landscape where $N \to \infty$ (infinite number of genes) and all the genes are initial good. As a consequence all mutations will be bad (the probability of a back mutation is zero). In this model the fitness will decrease in spite of mutation (hence its name). This model is supposed to provide one explanation of why sex is useful — in asexual population the fitness always decreases (see, for example, Higgs and Woodcock [16]). This is clearly rubbish and a consequence of choosing an out of equilibrium starting condition. Figure 1.3 shows the rate of change of fitness as a function of the average fitness per gene for one set of parameters (taken from [17]). Muller's ratchet corresponds to the situation at $k_1 = 1$. In practice the population will sit at the equilibrium point where the rate of change of fitness is zero.

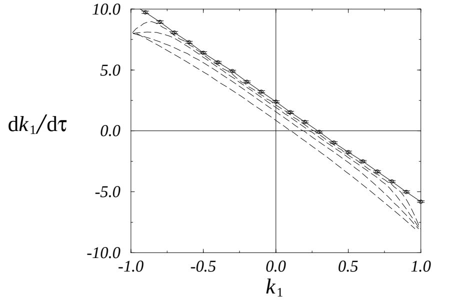


Figure 1.3: Example of evolution in a multiplicative fitness landscape for an infinitely long gene sequence. The rate of change of fitness is shown as a function of the average fitness per gene (solid curve). The dotted curves shows the evolution for sequences of length $N = 100$, 250, 500 and 1000.

- **Models of Sexual Recombination.** There are currently many models to understand why there is sex. The problem is not so much why is there sex but why are there males. If in a sexual population a female could mutate so that she could fertilize herself and produced only daughters and this trait was inherited, then this mutant strain would breed at twice the rate of a male-female pair. It would quickly take over the population. Thus two gendered populations are unstable to this kind of mutation. Indeed there are examples of species where this has been known to happen. Of course this is not a problem for hermaphrodites and the really question may be why do hermaphrodites specialize into males and females?

Lecture 2

Micro-Evolution as a Statistical Mechanics Problem

2.1 The Statistical Mechanics Formalism

In the next two lectures we will develop a formalism for quantitatively predicting the dynamics of evolving populations. This approach has been developed in a series of papers by Jon Shapiro, Magnus Rattray and myself [17–22].

2.1.1 Phase Space for Models of Micro-Evolution

We will confine our attention to models of micro-evolution with a well defined dynamics. The evolution operators are stochastic so that the natural framework for describing these models is statistical mechanics. Of course, there is no detailed balance so we do not expect the population to approach a Boltzmann distribution. In many models the system will reach a steady state equilibrium, but we will be interested in the full dynamics not just the final equilibrium state.

In the rest of this course we will make the following simplifying assumptions

- The population size, P, remains fixed.
- The gene length, N, remains fixed.
- The fitness, F_α, of member α depends only on the gene sequence, $\vec{S}^\alpha$, (all nature and no nurture) and does not change in time.
- The alleles take on Ising values, $S_i^\alpha \in \{-1, 1\}$.

These assumptions clearly do not hold for biological evolution, although in Genetic Algorithms they are often satisfied by construction. They may be relaxed, but for the sake of keeping the models simple they will not be. We will consider models whose dynamics is given by the, now familiar, steps

1. Generate a population at random.
2. Measure the fitness of each member in the population.
3. Select the fittest members of the population preferentially.

4. Perform mutation on each member of the population.
5. Perform crossover on the population.
6. Go back to 2.

For the model defined above the population can be represented as a point on an $N \times P$-dimensional hyper-cube (since the population can be represented as a binary string of length $N \times P$). This hyper-cube can be regarded as the **phase space** of the system. The full evolution corresponds to a trajectory in the phase space. The trajectory will be stochastic because of the random nature of the evolution operators. To provide a complete description of the evolution we would need to known the probability $P(x, t)$ for being at a point x in the phase space at generation t. The evolution of this probability density can be described in terms of a **master equation**, which we can schematically write as

$$P(x', t+1) = \hat{C}\hat{M}\hat{S}P(x, t)$$

where $\hat{S}$, $\hat{M}$ and $\hat{C}$ are linear operators ($2^{NP} \times 2^{NP}$ unitary matrices) giving the transition probability for going from one configuration x to another x' under the operations of selection, mutation and crossover, respectively. Despite the illusory simplicity of the master equation this approach is most probably intractable. Master equations are rarely if ever the easiest approach to solving a statistical mechanics problem. However this approach does have its advocates most notably Michael Vose and his co-workers [23–25]. The master equation approach does, however, show that evolution is a statistical mechanics problem and we should be able to use statistical mechanics to solve it.

2.1.2 Macroscopic Variables

The usual approach to simplifying statistical mechanics problems is to consider **macroscopic variables** in the belief that the microscopic details are washed out by the randomness. The macroscopic variables are usually usually the quantities of interested. In models of micro-evolution the natural macroscopic variables are those that describe the distribution of fitnesses, such as the average fitness and the variance in the distribution of fitnesses, as well as quantities measuring the correlation between gene sequences.

The different evolution operators couple to the microscopic variables in different ways. Thus selection cares only about the distribution of fitnesses and not on any of the microscopic details. Mutation depends on the fitness of a gene sequence, but may also depend on which alleles are present in the string. This is illustrated by the random field paramagnet

$$F_\alpha = \sum_{i=1}^{N} h_i \, S_i^\alpha \tag{2.1.1}$$

where the genes S_i^α are Ising variables and h_i are random variables drawn from some distribution $P(h_i)$. In this case the effect of mutation will depend on whether many small fields are satisfied or a few large fields are satisfied (where a gene is satisfied if $h_i S_i^\alpha > 0$). This information is not available from the distribution of fitnesses. For this type of problem we will try to infer this information using a maximum entropy ansatz (see section 3.2). For the special case of ones-counting mutation depends only on the distribution of fitnesses. Crossover will depend on the correlation between the genes as well as on their fitnesses. Thus to do crossover properly we should keep information about the correlation.

To facilitate a general discussion we well denote the macroscopic variables describing a population $\mathcal{P}(t)$ at time t by $\Omega(t) = \{\omega_1, \omega_2, \ldots, \omega_M\}$. For a finite population the evolution will be stochastic, even in the space of macroscopic variables. We must again use a probability density, $P(\Omega, t)$, to describe the evolution deterministically.

$$P(\Omega, t)\,\mathrm{d}\Omega \equiv \text{Probability of being in a volume } \mathrm{d}\Omega \text{ around the point } \Omega \text{ at time } t$$

where $\mathrm{d}\Omega$ denotes a measure element in the space of macroscopic variables. By definition

$$\int P(\Omega, t)\,\mathrm{d}\Omega \equiv \langle\!\langle 1 \rangle\!\rangle = 1.$$

Because we are in the space of macroscopic variables the probability density $P(\Omega, t)$ will be compact (this certainly was not the case in the full phase space). We can think of this probability density as defining an (Gibbsian) **ensemble**. To describe the ensemble we can use a set of **ensemble variables**, $\Xi = \{\xi_1, \xi_2, \ldots\}$ (these are like the thermodynamic variables and order parameters we usually meet in statistical mechanics). Examples of ensemble variables would be the average value of a macroscopic variable

$$\bar{\omega}_n = \langle\!\langle \omega_n \rangle\!\rangle \equiv \int \omega_n P(\Omega, t)\,\mathrm{d}\Omega \tag{2.1.2}$$

or the covariances between macroscopic variables

$$C_{m,n} = \langle\!\langle \omega_m \omega_n \rangle\!\rangle - \langle\!\langle \omega_m \rangle\!\rangle \langle\!\langle \omega_n \rangle\!\rangle\,. \tag{2.1.3}$$

A simple model for the ensemble is to assume it is described by a multi-variant Gaussian distribution[1]

$$P(\{\omega_n\}) = \frac{1}{\sqrt{(2\,\pi)^M\,\det[C]}}\,\mathrm{e}^{-\frac{1}{2}\sum_{m,n} \delta\omega_m\,[C^{-1}]_{m,n}\,\delta\omega_n} \tag{2.1.4}$$

where $\delta\omega_n = \omega_n - \bar{\omega}_n$. In doing so we are clearly ignoring the higher order statistics of the ensemble.

To model the evolution including stochastic fluctuations we should consider the evolution of the full ensemble. This is schematically illustrated in figure 2.1. The problem

[1] To calculate averages over a multi-variant Gaussian distribution a useful trick is to introduce a source term $\exp(\sum_n h_n\,\delta\omega_n)$. Then

$$\begin{aligned}
\langle\!\langle f(\omega_n) \rangle\!\rangle &= \langle\!\langle f(\bar{\omega}_n + \delta\omega_n) \rangle\!\rangle \\
&= \left\langle\!\!\left\langle f\left(\bar{\omega}_n + \frac{\partial}{\partial h_n}\right) \exp\left(\sum_n h_n\,\delta\omega_n\right) \right\rangle\!\!\right\rangle \Bigg|_{h_n=0} \\
&= f\left(\bar{\omega}_n + \frac{\partial}{\partial h_n}\right) \left\langle\!\!\left\langle \exp\left(\sum_n h_n\,\delta\omega_n\right) \right\rangle\!\!\right\rangle \Bigg|_{h_n=0} \\
&= f\left(\bar{\omega}_n + \frac{\partial}{\partial h_n}\right) \exp\left(\frac{1}{2}\sum_{m,n} h_m\,C_{m,n} h_n\right) \Bigg|_{h_n=0}
\end{aligned}$$

Thus, for example, $\langle\!\langle w_n \rangle\!\rangle = \bar{\omega}_n$, $\langle\!\langle w_m\,w_n \rangle\!\rangle = \bar{\omega}_m\,\bar{\omega}_n + C_{m,n}$, etc.

of solving the dynamics of an evolving population is thus to calculate the evolution of the ensemble variables

$$\Xi(t) \overset{selection}{\longrightarrow} \Xi^{s}(t) \overset{mutation}{\longrightarrow} \Xi^{s,m}(t) \overset{crossover}{\longrightarrow} \Xi^{s,m,c}(t) = \Xi(t+1).$$

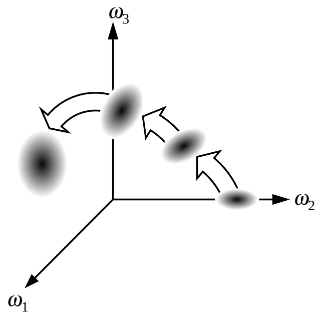


Figure 2.1: Schematic representation of the evolution of an ensemble of populations in the space of macroscopic variables.

2.1.3 Truncation of the Set of Ensemble Variables

There are infinitely many ensemble variables which could be used to describe the ensemble of populations. To make the problem tractable we must choose a finite set of them. In doing this we lose some information about the ensemble. If this information is necessary in determining the evolution then we must make some assumption about this missing information. This is a common problem in statistical mechanics where microscopic information about the phase space is discarded. The information is usually inferred from a maximum entropy ansatz. *A priori* there is no way to know whether this ansatz is correct and it must check *a posteriori*. In the case of evolving populations the problem is more acute because we will not take the thermodynamic limit (i. e. take the limit of an infinite population) since the asymptotic behaviour in this limit is highly non-trivial. (It is the trivial asymptotic behaviour of physical systems that lies at the heart of thermodynamics — i. e. the thermodynamic functions are all extensive quantities.) Without taking the thermodynamic limit we have little hope of describing the dynamics exactly with a finite number of ensemble variables. Nevertheless, we would hope that by taking a sufficient number of ensemble variables we might obtain a reasonable approximation.

To add some substance to this hope figure 2.2 shows a comparison between simulation and theory for ones-counting with only selection and mutation (asexual reproduction in a multiplicative fitness landscape). The curves show the first four cumulants, κ_n, (see below) together with the covariances $v_{n,n}$ for the first two cumulants. The meaning of the parameters values will become clear during the rest of the course. This is a particularly simple problem because the evolution depends only on the distribution of fitnesses. As ensemble variables the first six cumulants were used as were the covariance $v_{m,n}$ with $m + n \leq 6$.

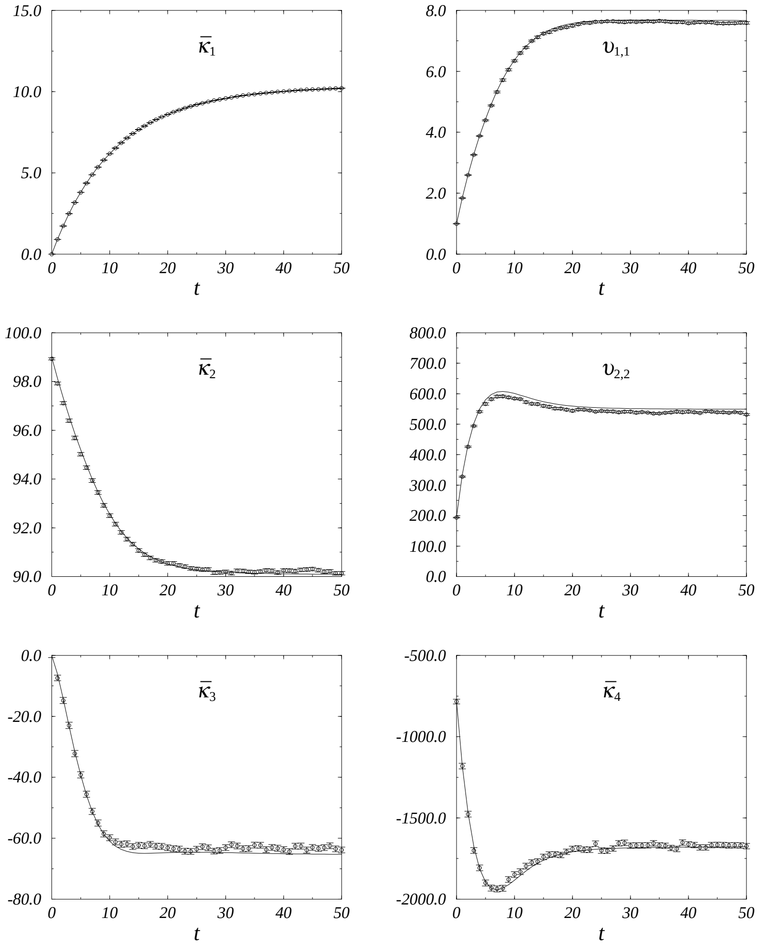


Figure 2.2: Examples of the predicted evolution of the ensemble variables $\bar{\kappa}_1$, $\bar{\kappa}_2$, $\bar{\kappa}_3$, $\bar{\kappa}_4$, $v_{1,1}$ and $v_{2,2}$ (solid lines) together with simulations (data points). In the example $P = 100$, $N = 100$, $\beta = 0.01$ and $\gamma = 4/N$. The simulations were averaged over 100 000 runs. The error bars show the predicted errors in the mean.

2.1.4 Cumulants

The distribution of fitnesses is an important factor in determining the evolution of a population since it determines selection. We will therefore spend some time to describe a distribution. A natural way to parameterize a (compact) distribution, $\rho(x)$, is in terms of its cumulants. (Selection ensures that the distribution remains fairly compact).

The first four cumulants, K_n, are defined to be

$$K_1 \equiv \langle x \rangle \tag{2.1.5}$$

$$K_2 \equiv \left\langle x^2 \right\rangle - \left\langle x \right\rangle^2 = \left\langle \left(x - \langle x \rangle\right)^2 \right\rangle \tag{2.1.6}$$

$$K_3 \equiv \left\langle \left(x - \langle x \rangle\right)^3 \right\rangle \tag{2.1.7}$$

$$K_4 \equiv \left\langle \left(x - \langle x \rangle\right)^4 \right\rangle - 3K_2^2 \tag{2.1.8}$$

where

$$\left\langle f(x) \right\rangle = \int \rho(x)\, f(x)\, \mathrm{d}x.$$

The first cumulant, K_1, is the **mean** of $\rho(x)$, the second cumulant, K_2, is its **variance**, while $K_3/K_2^{3/2}$ is its **skewness** and K_4/K_2^2 is its **kurtosis**. The cumulants can be obtained from the **cumulant generating function**

$$G(z) = \log \left\langle \mathrm{e}^{\,z\,x} \right\rangle \tag{2.1.9}$$

Taking derivatives of the generating function we obtain

$$G'(z) = \frac{\langle x\, \mathrm{e}^{\,z\,x} \rangle}{\langle \mathrm{e}^{\,z\,x} \rangle}$$

so that $G'(0) = \langle x \rangle = K_1$. Similarly

$$G''(z) = \frac{\langle x^2\, \mathrm{e}^{\,z\,x} \rangle}{\langle \mathrm{e}^{\,z\,x} \rangle} - \frac{\langle x\, \mathrm{e}^{\,z\,x} \rangle^2}{\langle \mathrm{e}^{\,z\,x} \rangle^2}$$

and $G'(0) = \langle x^2 \rangle - \langle x \rangle^2 = K_2$. In general

$$K_n = \left. \frac{\partial^n G(z)}{\partial z^n} \right|_{z=0} \equiv G^{(n)}(0). \tag{2.1.10}$$

Note that for a Gaussian distribution

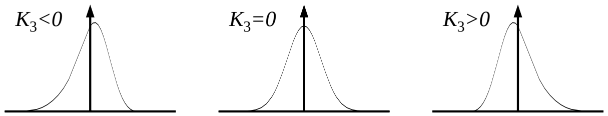
$$\rho(x) = \frac{1}{\sqrt{2\pi}\,\sigma}\, \mathrm{e}^{-(x-\mu)^2/(2\sigma^2)}$$

$$\left\langle \mathrm{e}^{\,z\,x} \right\rangle = \int \frac{\mathrm{d}x}{\sqrt{2\pi}\,\sigma} \mathrm{e}^{-(x-\mu)^2/(2\sigma^2) + z\,x} = \mathrm{e}^{\mu\, z + \sigma^2 z^2/2}$$

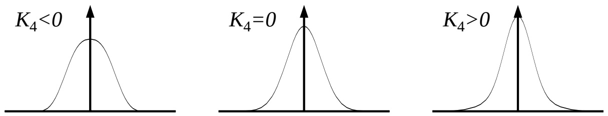
so that

$$G(z) = \mu\, z + \frac{\sigma^2 z^2}{2}$$

and $K_1 = \mu$, $K_2 = \sigma^2$ and $K_n = 0$ for $n > 2$. Thus the higher cumulants of a distribution can be regarded as a measure of how much it differs from a Gaussian. A skewed distribution will typically have one long tail.



The kurtosis typically signifies is the tails of the distribution are more or less pronounced than a Gaussian distribution.



Finite Sample Effects

A set of sample points drawn from a distribution $\rho(x)$ will have a different set of average cumulants to those of the distribution. To show this consider a finite sample $\{x_1, x_2, \cdots, x_P\}$ drawn from a distribution $\rho(x)$. We will denote the cumulants for the finite sample by κ_n. We wish to calculate the average cumulants $\bar{\kappa}_n$ for the ensemble of finite samples generated by drawing samples from $\rho(x)$ in all possible ways. The average first cumulant for the ensemble is

$$\bar{\kappa}_1 = \left\langle \frac{1}{P} \sum_{\alpha=1}^{P} x_\alpha \right\rangle = \frac{1}{P} \sum_{\alpha=1}^{P} \langle x_\alpha \rangle$$

but $\langle x_\alpha \rangle = K_1$ so $\bar{\kappa}_1 = K_1$. The second cumulant can be calculated in a similar fashion

$$\begin{aligned}
\bar{\kappa}_2 &= \left\langle \frac{1}{P} \sum_{\alpha=1}^{P} x_\alpha^2 - \left(\frac{1}{P} \sum_{\alpha=1}^{P} x_\alpha \right)^2 \right\rangle \\
&= \left(\frac{1}{P} - \frac{1}{P^2} \right) \sum_{\alpha=1}^{P} \left\langle x_\alpha^2 \right\rangle - \frac{1}{P^2} \sum_{\alpha \neq \beta} \langle x_\alpha \, x_\beta \rangle
\end{aligned}$$

where we have separated out the the term $\alpha = \beta$. Since the terms are drawn independently $\langle x_\alpha \, x_\beta \rangle = \langle x_\alpha \rangle^2$ and

$$\begin{aligned}
\bar{\kappa}_2 &= \left(\frac{1}{P} - \frac{1}{P^2} \right) P \left\langle x_\alpha^2 \right\rangle - \frac{P(P-1)}{P^2} \langle x_\alpha \rangle^2 \\
&= \left(1 - \frac{1}{P} \right) \left(\left\langle x_\alpha^2 \right\rangle - \langle x_\alpha \rangle^2 \right) = \mathcal{P}_2 K_2
\end{aligned}$$

where $\mathcal{P}_2 = (1 - 1/P)$. K_2 is sometimes called the estimator for the variance of a distribution. Following a similar procedure we find the finite sample corrections for the higher cumulants are[2]

$$\begin{aligned}
\kappa_3 &= \mathcal{P}_3 \, K_3, & \mathcal{P}_3 &= \mathcal{P}_2 \, (1 - 2/P) \\
\kappa_4 &= \mathcal{P}_4 \, K_4 - 6 \mathcal{P}_2 \, K_2^2 / P, & \mathcal{P}_4 &= \mathcal{P}_2 \, (1 - 6/P + 6/P^2).
\end{aligned}$$

[2]Finite sample effects in higher cumulants are not widely known. Thus for example in Numerical Recipes [26, section 14.1] they correct the variance but not the higher cumulants.

The covariance, $v_{1,1}$, measuring the fluctuation in the mean from sample to sample, is given by

$$\begin{aligned}
v_{1,1} &= \left\langle \kappa_1^2 \right\rangle - \left\langle \kappa_1 \right\rangle^2 \\
&= \left\langle \left(\frac{1}{P} \sum_{\alpha=1}^{P} x_\alpha \right)^2 \right\rangle - \left(\frac{1}{P} \sum_{\alpha=1}^{P} \langle x_\alpha \rangle \right)^2 \\
&= \frac{1}{P^2} \sum_{\alpha=1}^{P} \left(\left\langle x_\alpha^2 \right\rangle - \langle x_\alpha \rangle^2 \right) = \frac{K_2}{P}
\end{aligned}$$

where the off diagonal terms cancelled. Similar calculations give

$$v_{1,2} = \frac{\mathcal{P}_2 K_3}{P}, \qquad v_{2,2} = \frac{1}{P} (2 \mathcal{P}_2 K_2^2 + \mathcal{P}_2^2 K_4).$$

Using Cumulants as Ensemble Variables

To describe the fluctuations in an evolving population we consider an ensemble of populations. To parameterize the distribution of these populations we will use the cumulants, K_n, for the **infinite population distribution**, $\rho(F)$, rather than those for the finite population, κ_n. The two descriptions are equivalent as we can go backwards and forwards between them. An ensemble of infinite population distributions with average cumulants $\bar{K}_n$ and and covariance $V_{n,m}$ is equivalent to an ensemble of finite populations with average cumulants and covariances given by

$$\bar{\kappa}_1 = \bar{K}_1 \tag{2.1.11}$$

$$\bar{\kappa}_2 = \mathcal{P}_2 \bar{K}_2 \tag{2.1.12}$$

$$\bar{\kappa}_3 = \mathcal{P}_3 \bar{K}_3 \tag{2.1.13}$$

$$\bar{\kappa}_3 = \mathcal{P}_4 \bar{K}_4 - \frac{6 \mathcal{P}_2}{P} \left(\bar{K}_2^2 + V_{2,2} \right) \tag{2.1.14}$$

$$v_{1,1} = V_{1,1} + \frac{1}{P} \bar{K}_2 \tag{2.1.15}$$

$$v_{1,2} = \mathcal{P}_2 V_{1,2} + \frac{\mathcal{P}_2}{P} \bar{K}_3 \tag{2.1.16}$$

$$v_{2,2} = \left(1 - \frac{1}{P^2} \right) V_{2,2} + \frac{1}{P} \left(2 \, \mathcal{P}_2 \, \bar{K}_2^2 + \mathcal{P}_2^2 \bar{K}_4 \right). \tag{2.1.17}$$

(These are obtained by averaging the results of the previous section over the ensemble.) There are some constraints of the type of finite population ensembles we can model using ensembles of infinite population distribution. For example,

$$v_{1,1} \geq \frac{\kappa_2}{P - 1}$$

otherwise $V_{1,1} < 0$ in which case the infinite population ensemble is not well defined. Because we are considering a stochastic system these constraints never seem to be violated. It is an illustration of the importance of microscopic randomness in formulating statistical mechanics.

The great advantage of using the ensemble of infinite population distributions is that for any finite population drawn from this distribution the members of the populations are independent! This is closely analogous to using a **canonical ensemble** rather than a **micro-canonical** ensemble in traditional statistical mechanics. We have managed to relax a strong constraint by using the fluctuations inherent in the ensemble.

2.2 Selection.

Having outline the formalism in general terms we now start to go into a detailed description of the calculations involved.

2.2.1 Overview

Selection can be performed in many different ways. Since the Genetic Algorithm community has explored many of these routes I will use their jargon. A popular selection method is to assign a weight $w_\alpha = w(F_\alpha)$ to each member of the population depending on its fitness. The new population is drawn (with replacement) from the old population with a probability

$$p_\alpha = \frac{w_\alpha}{Z}, \qquad Z = \sum_{\alpha=1}^{P} w_\alpha. \tag{2.2.1}$$

(Note that in biology jargon the weight w_α would be called the fitness.)

One convenient choice of weights is the Boltzmann factor

$$w_\alpha = \mathrm{e}^{\beta F_\alpha} \tag{2.2.2}$$

where β controls the strength of selection: if $\beta = 0$ then each member of the population is drawn with the same probability; if β is infinite then only the fittest member of the population is selected. Using a Boltzmann factor does not, of course, have anything to do with achieving a thermodynamic equilibrium; it is however a convenient choice for many reasons. It ensures that the weight are positive. More significantly it makes the probability, p_α, invariant under a transformation $F_\alpha \to F_\alpha + c$. Furthermore if we rescale β so that $\beta = \beta_s/\sqrt{\kappa_2}$ then p_α remains invariant under a general linear transformation $F_\alpha \to aF_\alpha + c$. This is particularly useful in Genetic Algorithms because it means the average fitness remains more or less constant whatever the average fitness and width of the population are. The typical behaviour of a GA is illustrated in figure 2.3. This shows the distributions of fitnesses after 0, 10, 20, 30, 40, 50 and 200 generations. Initially the population is random. As it evolves the fitness increases but the width decreases — in GA lingo this is called **convergence**.

Although the Boltzmann weight is a convenient choice and probably as good as any it is not a common choice amongst GA people. It is, however, entirely equivalent to the selection procedure used by biologists in the multiplicative fitness landscape model. We will discuss over popular choices of selection mechanism later.

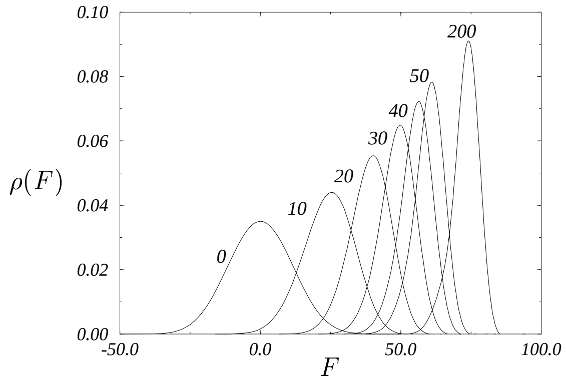
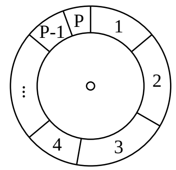


Figure 2.3: The distribution of fitness is shown after 0, 10, 20, 30, 40, 50 and 200 steps of the GA applied to the spin-glass chain. The curves were created by averaging over 10 000 samples for $N = 127$, $P = 50$, $\beta_s = 0.1$, $\gamma = 1/2N$ and $f = 1$. The average ground state energy is at $F_{max} \approx 101$. The figure was taken from reference [21].

Having weighted the members of the population we still have a choice as to how we draw the members of the new population. There is the classic dumb method known as **roulette wheel selection** and the method of choice commonly known as **Baker selection** [27]. In roulette wheel selection the members of the population are drawn independently. This can be visualizes as playing roulette where the size of the pockets are proportional to the weights w_α.



Roulette wheel selection is dumb because it allows good members to be lost through chance (more accurately it reduces the diversity of the population unnecessarily). A better approach is to select the new population so that the number of times, n_α, an individual α is selected is equal to (round up or down) $P\,p_\alpha$. A procedure to achieved this can be visualized as spinning a roulette wheel only once and the placing P balls at equal intervals around the wheel. For GAs there is no excuse for not using Baker selection.

In some application the fitness is prohibitively expensive to evaluate exactly, but may be estimated much more readily. In biology external factors might prevent the fittest members from achieving the reproductive potential. We can model this kind of noisy fitness evaluation by using as a weighting function

$$w_\alpha = \mathrm{e}^{\beta(F_\alpha + \eta_\alpha)} \tag{2.2.3}$$

where η_α is a randomly distributed variable. This problem has been studied by Magnus Rattray and Jonathan Shapiro [22]. Although this problem is very interesting and has many applications we will not pursue it any further in these lectures.

2.2.2 Calculating the Effect of Selection

Selection depends only on the distribution of fitnesses (the phenotype) and not on the details of the model being studied. In this respect it is problem independent and our discussion can be applied to all models of micro-evolution.

We will consider roulette wheel selection as this is the simplest mechanism to model since the new members of the population are drawn independently. We will also use Boltzmann weights, although the generalization to arbitrary weights is straightforward. We indicate how these generalization can be made at the end of this lecture.

To understand selection we will first consider what happens to the first couple of cumulants. Suppose we have a population consisting of P members with fitness, F_α, and we select a new population also consisting of P members. If we denote the number of times an individual α is selected by n_α then the first two cumulants after selection will be given by

$$\begin{aligned} \kappa_1' &= \frac{1}{P}\sum_{\alpha=1}^{P} n_\alpha F_\alpha \\ \kappa_2' &= \frac{1}{P}\sum_{\alpha=1}^{P}\left(n_\alpha - \frac{n_\alpha^2}{P}\right) F_\alpha^2 - \frac{1}{P^2}\sum_{\alpha\neq\beta} n_\alpha n_\beta F_\alpha F_\beta. \end{aligned} \tag{2.2.4}$$

where in κ_2' the term $\alpha=\beta$ has been separated out. The probability of a particular draw, $\vec{n} = (n_1, n_2, \ldots, n_P)$, is given by the multinomial probability distribution

$$P(\vec{n}) = P! \prod_{\alpha=1}^{P} \frac{p_\alpha^{n_\alpha}}{n_\alpha!}\; \delta\left(\sum_{\beta=1}^{P} n_\beta - P\right). \tag{2.2.5}$$

Averaging equations (2.2.4) over all possible outcomes we find

$$\begin{aligned} \kappa_1^s &= \sum_{\alpha=1}^{P} p_\alpha F_\alpha \\ \kappa_2^s &= \left(1-\frac{1}{P}\right)\left(\sum_{\alpha=1}^{P}\left(p_\alpha - p_\alpha^2\right) F_\alpha^2 - \sum_{\alpha\neq\beta} p_\alpha p_\beta F_\alpha F_\beta\right) \end{aligned} \tag{2.2.6}$$

where we have used[3]

$$\langle n_\alpha\rangle = P\,p_\alpha, \qquad \langle n_\alpha n_\beta\rangle = P\,(P-1)\,p_\alpha p_\beta, \qquad \left\langle n_\alpha^2\right\rangle = P\,(P-1)\,p_\alpha^2 + P\,p_\alpha.$$

[3] A simple way to calculate these is to note that

$$\langle f(n_\alpha)\rangle = f\left(p_\alpha \frac{\partial}{\partial p_\alpha}\right)\langle 1\rangle$$

which follows from the definition of $P(\vec{n})$ in equation (2.2.5). Using the fact that the multinomial coefficients are the terms in the expansion of a multinomial raised to some power.

$$\langle f(n_\alpha)\rangle = f\left(p_\alpha \frac{\partial}{\partial p_\alpha}\right)\left(\sum_{\beta=1}^{P} p_\beta\right)^P$$

To calculate the average we first take the derivatives with respect to p_α and then use the fact that $\sum_\alpha p_\alpha = 1$.

The factor $\mathcal{P}_2 = (1 - 1/P)$ in the expression for κ_2^s arises because there are P members in the **new** population. To find the cumulants for the infinite population distribution we use the relationship $K_1 = \kappa_1$ and $K_2 = \kappa_2/\mathcal{P}_2$, see appendix 2.1.4. Assuming that the F_α's are drawn from an infinite population ensemble we can treat them as independent. This allows us to average them independently over a distribution $\rho(F_\alpha)$ with cumulants K_n. Finally we can average over an ensemble of distributions with average cumulants $\bar{K}_n$ and covariance $V_{n,m}$. Thus

$$\begin{aligned} \bar{K}_1^s &= \left\langle\!\!\left\langle \left\langle \sum_{\alpha=1}^{P} p_\alpha F_\alpha \right\rangle_\rho \right\rangle\!\!\right\rangle \\ \bar{K}_2^s &= \left\langle\!\!\left\langle \left\langle \left(\sum_{\alpha=1}^{P}\left(p_\alpha - p_\alpha^2\right) F_\alpha^2 - \sum_{\alpha\neq\beta} p_\alpha p_\beta F_\alpha F_\beta\right) \right\rangle_\rho \right\rangle\!\!\right\rangle \end{aligned} \tag{2.2.7}$$

where $\langle\cdots\rangle_\rho$ denotes the average

$$\left\langle f(F_\alpha)\right\rangle_\rho = \left(\prod_{\beta=1}^{P}\int_{-\infty}^{\infty}\rho(F_\beta)\,d\,F_\beta\right) f(F_\alpha) \tag{2.2.8}$$

and $\langle\!\langle\cdots\rangle\!\rangle$ denotes the average over the ensemble.

To calculate all the cumulants after selection we can use the generating function

$$G_1(z) = \left\langle\!\!\left\langle \left\langle \operatorname*{Tr}_{\{\vec{n}\}} P(\vec{n}) \log\left(\sum_{\alpha=1}^{P} n_\alpha \mathrm{e}^{z F_\alpha}\right)\right\rangle_\rho \right\rangle\!\!\right\rangle \tag{2.2.9}$$

where the trace denotes the sum over all possible ways of doing the selection. The average cumulants after selection are given by the derivatives of the generating function, $\bar{\kappa}_n = G_1^{(n)}(0)$. This formula simplifies if we wish to find the cumulants for the infinite population after selection since, when we select an infinite population, $n_\mu = P\,p_\mu = P\,w_\mu \sum_\nu w_\nu$. Expanding the logarithm

$$\log\left(\frac{\sum_\mu w_\mu \mathrm{e}^{z F_\mu}}{P\sum_\nu w_\nu}\right) = \log\left(\sum_\mu w_\mu \mathrm{e}^{z F_\mu}\right) - \log\left(\frac{1}{P}\sum_\nu w_\nu\right) \tag{2.2.10}$$

and noting that the second term does not depend on z so can be dropped, we obtain the generating function for the average cumulants of the infinite population

$$G_1^\infty(z) = \left\langle\!\!\left\langle \left\langle \log\left(\sum_{\alpha=1}^{P} w_\alpha \mathrm{e}^{z F_\alpha}\right)\right\rangle_\rho \right\rangle\!\!\right\rangle \tag{2.2.11}$$

where $\bar{K}_n = G_1^{\infty\,(n)}(0)$. We are effectively breaking up the selection operation into two pieces; we first select an infinite population with the weighting factor $w(F_\alpha)$, and then we select P members at random from the infinite population. It is easy to check the first two derivatives of $G_1^\infty(z)$ give the same results as equation (2.2.7).

When we select in proportion to the Boltzmann factor, equation (2.2.2), we can use the somewhat simpler generating function

$$G_1^\infty(\beta) = \left\langle\!\!\left\langle \left\langle \log\left(\sum_{\alpha=1}^{P} \mathrm{e}^{\beta F_\alpha}\right)\right\rangle_\rho \right\rangle\!\!\right\rangle \tag{2.2.12}$$

where the average cumulants after selection are now given by $\bar{K}_n = G_1^{\infty\,(n)}(\beta)$.

We have still to evaluate equation (2.2.11) or (2.2.12). To do this we exploit a trick used by Derrida to solve the random energy model [1]. We use the integral representation of the logarithm [4]

$$\log(Z) \equiv \int_0^\infty \frac{\mathrm{e}^{-t} - \mathrm{e}^{-tZ}}{t}\,\mathrm{d}t. \tag{2.2.13}$$

The generating function then becomes

$$G_1^\infty(\beta) = \left\langle\!\!\left\langle \int_0^\infty \frac{\mathrm{e}^{-t} - g^P(t,\beta)}{t}\,\mathrm{d}t \right\rangle\!\!\right\rangle \tag{2.2.14}$$

where

$$g(t,\beta) = \int \rho(F)\,\mathrm{e}^{-t\mathrm{e}^{\beta F}}\,\mathrm{d}\,F. \tag{2.2.15}$$

These two integrals can be performed numerically to obtain the cumulants after selection. Details of this approach are given in the appendices of reference [17].

Weak selection expansion

We can also find an expansion in $1/P$ for cumulants after selection provided $\beta\sqrt{K_2}$ remains sufficiently small. For biological system this expansion is usually valid. To perform this expansion we first expand the exponential in equation (2.2.15) in a power

[4] This is not immediately obvious. Naïvely we might think of splitting the integral into two and making a change of variables $s = tZ$ in the second integral, but then the integral vanishes! This is clearly the wrong answer; the problem is that the integrals do not converge when the integrand is split into two. Thus we must be more cautious and define the integral in terms of a limit

$$I(Z) = \lim_{\epsilon\to 0} \int_\epsilon^\infty \frac{\mathrm{e}^{-t} - \mathrm{e}^{-tZ}}{t}\,\mathrm{d}t.$$

Now we can split the integrand and make the change of variables

$$\begin{aligned} I(Z) &= \lim_{\epsilon\to 0}\int_\epsilon^\infty \frac{\mathrm{e}^{-t}}{t}\,\mathrm{d}t - \int_{\epsilon Z}^\infty \frac{\mathrm{e}^{-s}}{s}\,\mathrm{d}s \\ &= \lim_{\epsilon\to 0}\int_\epsilon^{\epsilon Z} \frac{\mathrm{e}^{-t}}{t}\,\mathrm{d}t = \lim_{\epsilon\to 0}\int_\epsilon^{\epsilon Z}\frac{1}{t}\,\mathrm{d}t + \mathcal{O}(\epsilon) \\ &= \log(Z) \ \blacksquare \end{aligned}$$

series in $t\exp(\beta F)$. This allows the integral over F to be performed. Re-exponentiating the leading term in t and performing the integral over t we obtain

$$G_1^\infty(\beta) \approx \hat{G}^\infty(\beta) = \left\langle\!\!\left\langle \log(\hat{\rho}_\infty(\beta)) - \frac{1}{2\,P}\frac{\hat{\rho}_\infty(2\beta)}{\hat{\rho}_\infty^2(\beta)} \right\rangle\!\!\right\rangle \tag{2.2.16}$$

where we have dropped constant terms and terms of order $1/P^2$, and where

$$\hat{\rho}_\infty(\beta) = \int_{-\infty}^{\infty} \rho(F)\,w(F)\,\mathrm{d}F \tag{2.2.17}$$

$w(F)$ is the weight assigned to a member with fitness F. In the case of Boltzmann selection $w(F) = \exp(\beta F)$, and

$$\hat{\rho}_\infty(\beta) = \exp\left(\sum_{n=1} \frac{K_n\beta^n}{n!}\right). \tag{2.2.18}$$

Putting this into equation (2.2.16) we obtain

$$\hat{G}_1^\infty(\beta) = \left\langle\!\!\left\langle \sum_{n=1}^{n_c} \frac{\beta^n K_n}{n!} - \frac{1}{2P}\exp\left(\sum_{n=2}^{n_c}\frac{(2^n-2)\,\beta^n\,K_n}{n!}\right)\right\rangle\!\!\right\rangle. \tag{2.2.19}$$

We note that selection depends non-linearly on the cumulants so that it will not just depend on the average cumulants of the ensemble but also on its higher order statistics.

We will assume that we can approximate the ensemble by a multi-variant Gaussian distribution

$$P(\{K_n\}) = \frac{1}{\sqrt{(2\,\pi)^{n_c}\,\det[V]}}\,\mathrm{e}^{-\frac{1}{2}\sum_{n,m}(K_n-\bar{K}_n)[V^{-1}]_{n,m}(K_m-\bar{K}_m)} \tag{2.2.20}$$

where n_c is the number of cumulants we choose to keep. This assumption can be checked **a posteriori** by comparing the results with simulations (see figure 2.2). Performing the average in equation (2.2.19) using equation (2.2.20) we find

$$\hat{G}_1^\infty(\beta) = \sum_{n=1}^{n_c}\frac{\beta^n\,\bar{K}_n}{n!} - \frac{1}{2P}\exp\left(\sum_{n=2}^{n_c} b_n(\beta)\,\bar{K}_n + \frac{1}{2}\sum_{n,m=2}^{n_c} b_n(\beta)\,b_m(\beta)\,V_{n,m}\right) \tag{2.2.21}$$

where $b_n(\beta) = (2^n-2)\,\beta^n/n!$. The average cumulants after selection can be computed by take derivative of $\hat{G}_1^\infty(\beta)$. The first cumulant changes as

$$\bar{K}_1^s = \bar{K}_1 + \beta\bar{K}_2 + \cdots - \frac{1}{P}\beta\bar{K}_2\mathrm{e}^{\beta^2\bar{K}_2^2+\cdots} + \cdots. \tag{2.2.22}$$

To leading order, $\bar{K}_1$ changes in proportion to β times the variance, $\bar{K}_2$, of the fitness distribution (this is only a good approximation if $\beta \sim 1/\sqrt{\bar{K}_2}$). The first cumulant does not influence the higher cumulants because selection is invariant under a rescaling of the fitness measures (i. e. $F_\alpha \to F_\alpha + c$).

The second cumulant changes according to

$$\bar{K}_2^s = \bar{K}_2 + \beta\bar{K}_3 - \frac{1}{P}\left(\bar{K}_2 + \beta^2\bar{K}_2^2\right)\mathrm{e}^{\beta^2\bar{K}_2^2+\cdots} + \cdots. \tag{2.2.23}$$

Even when there is no selection (i. e. $\beta = 0$) the variance is reduced by $\mathcal{P}_2 = 1 - 1/P$ because, by chance, some members of the population are selected more than others. As the selection strength β increases the variance is reduced. To obtain the full dependence of $\bar{K}_2^s$ on β one needs to use the exact equations for $G_1^\infty(\beta)$ and perform the integrals numerically. This is shown if figure 2.4. (The scaling of the abscissa is chosen so that in the limit $P \to \infty$ there is a first order phase transition at 1.) We observe there is a shoulder on the curve which is a reflection of the first order phase transition. The optimal selection rate is around this shoulder as there is a reasonable increase in fitness at relatively little cost in terms of losing diversity. The curve for Baker selection is very similar to this curve except rescaled be a factor very close to $1/\mathcal{P}_2$. Thus there is little or no inherent loss of diversity and it pays to choose a much smaller selection rate.

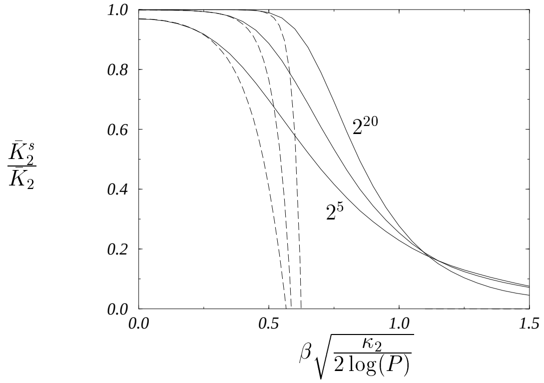


Figure 2.4: The change in width of fitness distribution after selection starting from a Gaussian distribution for $P = 2^5$, 2^{10} and 2^{20}. The solid lines are calculated by numerical integration. The dashed lines show the small β expansion. Taken from reference [21]
.

The variances is also effected by the higher cumulants. The third cumulant becomes negative ($\bar{K}_2^s$ will be a monotonically decreasing function of β so its derivative, $\bar{K}_3^s$, will always be negative). This will then reduce $\bar{K}_2^s$ even more. The effect of selection is schematically illustrated in figure 2.5. In reality the effect of selection is stochastic and the only accurate description is in terms of the whole ensemble.

For small β the direct effect of the ensemble covariances are small for the first few cumulants, however, even for $\beta = 0$ the fourth cumulant evolves as

$$\bar{K}_4^s\Big|_{\beta=0} = \left(1 - \frac{7}{P}\right)\bar{K}_4 - \frac{6}{P}\left(V_{2,2} + \bar{K}_2^2\right) \tag{2.2.24}$$

thus fluctuations (here $V_{2,2}$) produce important corrections in the evolution of the higher cumulants for finite populations. These corrections eventually effect the estimates for the evolution of the lower cumulants because of the coupling between the cumulants. The covariances of the ensemble effect the average cumulants because selection depends non-linearly on the cumulants in the population. This is not true of mutation and sexual

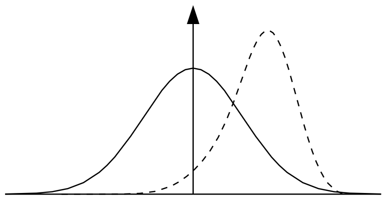


Figure 2.5: Schematic illustration of the effect of selection on the distribution of fitnesses. The solid curve shows the distribution before selection while the dashed curve shows the distribution after selection. The average fitness is increased while the width of the distribution is narrowed and a negative skewness develops.

recombination where we obtain linear relationships between the cumulants before and after the applying the operators.

2.2.3 The Covariances After Selection

Since selection depends on the covariances, $V_{n,m}$, as well as the average cumulants in the ensemble, we must also calculate how these are changed by selection. The covariances arise because the evolution operators are stochastic; that is, there will be fluctuations from population to population. To calculate the covariances of the ensemble after selection we use the generating function

$$G_2^\infty(z_1, z_2) = \left\langle\!\!\left\langle \left\langle \log\left(Z(z_1)\right)\, \log\left(Z(z_2)\right)\right\rangle_\rho \right\rangle\!\!\right\rangle - \left\langle\!\!\left\langle \left\langle \log\left(Z(z_1)\right)\right\rangle_\rho \right\rangle\!\!\right\rangle \left\langle\!\!\left\langle \left\langle \log\left(Z(z_2)\right)\right\rangle_\rho \right\rangle\!\!\right\rangle \tag{2.2.25}$$

where

$$Z(z) = \sum_{\alpha=1}^{P} w(F_\alpha)\mathrm{e}^{zF_\alpha}. \tag{2.2.26}$$

The covariances are given by

$$V_{n,m}^s = \frac{\partial^{n+m}}{\partial z_1^n z_2^m} G_2^\infty(z_1, z_2)\bigg|_{z_1=z_2=0} = G_2^{\infty(n,m)}(0,0). \tag{2.2.27}$$

Using equation (2.2.13) we can derive a new identity

$$\left\langle \log(Z_1)\log(Z_2)\right\rangle_c \equiv \int_0^\infty \frac{\mathrm{d}t_1}{t_1}\int_0^\infty \frac{\mathrm{d}t_2}{t_2}\left\langle \mathrm{e}^{-t_1 Z_1}\mathrm{e}^{-t_2 Z_2}\right\rangle_c \tag{2.2.28}$$

(we use the subscript c to denote the "connected" correlation functions $\langle A\, B\rangle_c = \langle A\, B\rangle - \langle A\rangle \langle B\rangle$).

For the case of Boltzmann weights we can also use a slightly simpler generating function $G_2^\infty(\beta_1, \beta_2)$, which is again given by equation (2.2.25) but now with

$$Z(\beta) = \sum_{\alpha=1}^{P} \mathrm{e}^{\beta F_\alpha} \tag{2.2.29}$$

and $V^s_{n,m} = G_2^{\infty(n,m)}(\beta,\beta)$. The covariances after selection can also be computed by numerical integration. In this case it is more efficient to use a slightly modified representation. The implementational details are again discussed in reference [17].

Weak selection expansion

We can also obtain an expansion in $1/P$ for $V^s_{n,m}$, provided $\beta\sqrt{K_2}$ is sufficiently small. The calculation is similar to that for G_1^∞. We find

$$\begin{aligned}\hat{G}_2^\infty(\beta_1,\beta_2) &= \left\langle\!\!\left\langle \log\left(\hat\rho_\infty(\beta_1)\right)\log\left(\hat\rho_\infty(\beta_2)\right)\right\rangle\!\!\right\rangle_c + \frac{1}{P}\left\langle\!\!\left\langle \frac{\hat\rho_\infty(\beta_1+\beta_2)}{\hat\rho_\infty(\beta_1)\,\hat\rho_\infty(\beta_2)}\right\rangle\!\!\right\rangle \\ &\quad -\frac{1}{2P}\left(\left\langle\!\!\left\langle \log\left(\hat\rho_\infty(\beta_1)\right)\frac{\hat\rho_\infty(2\beta_2)}{\hat\rho^2_\infty(\beta_2)}\right\rangle\!\!\right\rangle_c + \left\langle\!\!\left\langle \log\left(\hat\rho_\infty(\beta_2)\right)\frac{\hat\rho_\infty(2\beta_1)}{\hat\rho^2_\infty(\beta_1)}\right\rangle\!\!\right\rangle_c\right).\end{aligned} \tag{2.2.30}$$

(Where we use the notation $\hat{G}_2^\infty$ to denote the approximation to order $1/P$ of G_2^∞.) Assuming the fluctuations are of the form given in equation (2.2.20) we find

$$\left\langle\!\!\left\langle \log\left(\hat\rho_\infty(\beta_1)\right)\log\left(\hat\rho_\infty(\beta_2)\right)\right\rangle\!\!\right\rangle_c = \sum_{n,m=1}^{n_c}\frac{\beta_1^n\,\beta_2^m}{n!\,m!}V_{n,m} \tag{2.2.31}$$

while

$$\left\langle\!\!\left\langle \frac{\hat\rho_\infty(\beta_1+\beta_2)}{\hat\rho_\infty(\beta_1)\hat\rho_\infty(\beta_2)}\right\rangle\!\!\right\rangle = \exp\left(\sum_{n=2}^{n_c}c_n(\beta_1,\beta_2)\bar K_n + \frac{1}{2}\sum_{m,n=2}^{n_c}c_m(\beta_1,\beta_2)c_n(\beta_1,\beta_2)V_{n,m}\right) \tag{2.2.32}$$

where $c_n(\beta_1,\beta_2) = ((\beta_1+\beta_2)^n - \beta_1^n - \beta_2^n)/n!$ and finally

$$\begin{aligned}&\left\langle\!\!\left\langle \log\left(\hat\rho_\infty(\beta_1)\right)\frac{\hat\rho_\infty(2\beta_2)}{\hat\rho^2_\infty(\beta_2)}\right\rangle\!\!\right\rangle_c = \\ &\sum_{n=1}^{n_c}\sum_{m=2}^{n_c}\frac{\beta_1^n}{n!}b_m(\beta_2)V_{n,m}\exp\left(\sum_{m=2}^{n_c}b_m(\beta_2)\bar K_m + \frac{1}{2}\sum_{n,m=2}^{n_c}b_m(\beta_2)\,b_n(\beta_2)\,V_{n,m}\right).\end{aligned} \tag{2.2.33}$$

The first three covariances are

$$V^s_{1,1} = V_{1,1} + \beta V_{1,2} + \cdots + \frac{1}{P}\left(\bar K_2 + \cdots\right) \tag{2.2.34}$$

$$V^s_{1,2} = V_{1,2} + \beta V_{2,2} + \cdots + \frac{1}{P}\left(\bar K_3 + \cdots\right) \tag{2.2.35}$$

$$V^s_{2,2} = V_{2,2} + \cdots + \frac{1}{P}\left(\bar K_4 + 2\bar K_2^2 + \cdots\right). \tag{2.2.36}$$

These together with the equations for the average cumulants after selection provide a set of equations relating the ensemble variables before and after selection. Note that the covariances $V_{1,n}$ effect only themselves but do not influence the evolution of the other ensemble variables. Nevertheless as they are quantities of interest we keep them as ensemble variables.

Note on Using a Truncated Cumulant Expansion

To make the calculation tractable only the first few cumulants are used. The lower cumulants will have the largest influence on selection so this approximation may not be too bad. There is, however, a further plausibility argument for making this truncation. The higher cumulants are more and more sensitive to the location of particular members of the population, particularly the outliers. These will be largely determined by chance so that the very high cumulants are likely to average to zero. The fluctuations in the higher cumulants are likewise likely to be similar to those arising from drawing at random a finite population from an infinite population distribution. That is for cumulants above some critical order we might expect $\bar K_n = 0$ and $V_{m,n} = 0$ (but, of course not $v_{n,m} = 0$ — these fluctuations are already taken into account). This line of reasoning has not been tested against simulations.

2.2.4 The Founder Effect: Correlations in the Population

An important effect of selection is that it correlates the population by introducing multiple copies of the same gene sequences into the population. This correlation will accumulate over time and is an important factor in reducing the search capacity of the population since crossover becomes less effective. This correlation is often referred to as the **founder effect**.

There is a second type of correlation which arises naturally because the population evolves to a more restricted part of the phase space. We will refer to this as the **thermal correlation** as it would arise in a thermalized population (that is given a population whose only constraint is that its average energy is equal to some constant). This correlation is relatively simple to compute.

Calculating the correlations form the founder effect has been pioneered by Magnus Rattray [22, 28]. A related but more sophisticated approach is currently under development [29]. To calculate the correlations due to the founder effect we need to known the number of times n_α a member of the population is copied into the new population. More precisely we would like to know the distribution, $P(n_\alpha)$, of n_α. The moments of this distribution can be calculated using a similar approach to that used to calculate the change in the cumulants

$$\langle n_\alpha^r\rangle = \langle (P\,p_\alpha)^r\rangle = P^r\left\langle\left(\frac{w_\alpha}{\sum_\beta w_\beta}\right)^r\right\rangle. \tag{2.2.37}$$

where the average implies an average over all possible populations in the ensemble. Using the identity (which follows from the definition of the Gamma function)

$$\frac{1}{A^n} = \frac{1}{\Gamma(n)}\int_0^\infty t^{n-1}\,\mathrm{e}^{-t\,A}\,\mathrm{d}t. \tag{2.2.38}$$

we find

$$\begin{aligned}\langle n_\alpha^r\rangle &= \left\langle \frac{P^r}{\Gamma(r)}\int_0^\infty t^{r-1}w_\alpha \mathrm{e}^{-t\sum_\beta w_\beta}\right\rangle \\ &= \frac{P^r}{\Gamma(r)}\int_0^\infty t^{r-1}\left\langle w\mathrm{e}^{-tw}\right\rangle\left\langle \mathrm{e}^{-tw}\right\rangle^{P-1}.\end{aligned} \tag{2.2.39}$$

where again we use the independence of the members of the population. This can again be computed numerically or evaluated in the weak selection limit. In fact we only need to calculate the same number of moments as cumulants of the fitness distribution. We defer any further discussion of this until section 3.3.

2.2.5 Other Types of Selection

There are many other popular forms of selection. These can be calculated in a similar manner to Boltzmann selection. A paper on this in in preparation [30]. Here we briefly discuss these algorithms and outline how they can be calculated.

Ranking Selection

In ranking selection the members of the population are ranked in order of fitness and assigned a weight proportional to their rank. This prevents very fit members of the population from being over represented in the next generation.

To calculate the effect of ranking selection we can write the probability of member α being selected as

$$p_\alpha = \int_{-\infty}^{F_\alpha} \rho(F)\,\mathrm{d}F. \tag{2.2.40}$$

We can then proceed as for Boltzmann selection. For a finite population drawn from a Gaussian distribution the average first two cumulants after selection are

$$\begin{aligned} \bar{K}_1^s &= \bar{K}_1 + \sqrt{\frac{\bar{K}_2}{\pi}}\left(1 - \frac{2}{3\,P}\right) \\ \bar{K}_2^s &= \left(1 - \frac{1}{\pi} - \frac{0.795199}{P}\right)\bar{K}_2 \end{aligned} \tag{2.2.41}$$

Binary Tournament Selection

A second popular method of selection is to randomly pick two members of the population and keep the fitter if the two. The effect of binary tournament selection can be calculate by a simple extension of our previous calculation (see reference [30] for details). Up to corrections of order $1/P$ this produces the same effect on a Gaussian distributed population as Ranking selection.

Truncation Selection

In truncation selection all the members of the population below a certain fitness are discarded and the remaining members are then replicated to keep the population size fixed. The most popular cutting point is the half way point so that the remaining members of the population are duplicated. This is a popular method for its simplicity but probably not the best method. It is discussed in reference [21, section 6.2.2].

Lecture 3

The Genetic Operators.

Unlike the case of selection the **genetic operators**, mutation and selection, depend on the model or problem being studied. We will restrict our attention to using the energy of simple “spin systems” as fitness functions (these models have be studied extensively in reference [17, 18, 21]). This approach has also been applied to problems with more complicated fitness functions, namely the subset-sum problem [28] and generalization in a binary perceptron [22]. We begin this section by considering the simpler of the two genetic operators mutation.

3.1 Mutation.

3.1.1 Mutation for the Ideal Paramagnet

We consider first the simplest fitness function, the ideal paramagnet,

$$F_\alpha = \sum_{i=1}^{N} S_i^\alpha \tag{3.1.1}$$

with $S_i^\alpha \in \{-1, 1\}$. We will assume that each gene is mutated with a probability γ

$$S_i^\alpha \to S_i'^\alpha = \begin{cases} -S_i^\alpha & \text{with probability } \gamma \\ S_i^\alpha & \text{with probability } 1-\gamma. \end{cases} \tag{3.1.2}$$

Averaging over the two possible outcomes we find

$$\langle S_i'^\alpha \rangle_{mut} = \Gamma S_i^\alpha, \qquad \Gamma \equiv 1 - 2\gamma. \tag{3.1.3}$$

Using this result it is straightforward to calculate what happens to the average cumulants after applying the mutation operator. The fitness after mutation of a sequences $\vec{S}^\alpha$ is, on average,

$$\langle F_\alpha' \rangle_{mut} = \left\langle \sum_{i=1}^{N} S_i'^\alpha \right\rangle_{mut} = \Gamma \sum_{i=1}^{N} S_i^\alpha = \Gamma F_\alpha. \tag{3.1.4}$$

The first cumulant is

$$\bar{K}_1^m = \left\langle\!\left\langle \left\langle \, \langle F_\alpha' \rangle_{mut} \right\rangle_{\mathcal{P}} \right\rangle\!\right\rangle = \Gamma \left\langle\!\left\langle K_1 \right\rangle\!\right\rangle = \Gamma \bar{K}_1 \tag{3.1.5}$$

where we have commuted the average over the members of the population and the average over all possible mutations. The higher cumulants are only a little more complicated

$$\begin{aligned} \left\langle (F'_\alpha)^2 \right\rangle_{mut} &= \left\langle \sum_{i \neq j} S'^\alpha_i S'^\alpha_j + \sum_i (S'^\alpha_i)^2 \right\rangle_{mut} \\ &= \Gamma^2 \sum_{i \neq j} S^\alpha_i \, S^\alpha_j + \sum_i 1 = \Gamma^2 \, F^2_\alpha + (1 - \Gamma^2) \, N \end{aligned} \tag{3.1.6}$$

and we have used $(S'^\alpha_i)^2 = 1$. The second cumulant is

$$K_2 = \left\langle F^2_\alpha \right\rangle_{\mathcal{P}} - \left\langle F_\alpha \right\rangle^2_{\mathcal{P}}.$$

Using this result together with equations (3.1.4) and (3.1.6) we find

$$\bar{K}^m_2 = \Gamma^2 \, \bar{K}_2 + (1 - \Gamma^2) \, N \tag{3.1.7}$$

The higher cumulants are

$$\bar{K}^m_3 = \Gamma^3 \, \bar{K}_3 - 2 \, \Gamma \, (1 - \Gamma^2) \, \bar{K}_1 \tag{3.1.8}$$

$$\bar{K}^m_4 = \Gamma^4 \, \bar{K}_4 - 8 \, \Gamma^2 \, (1 - \Gamma^2) \, \bar{K}_2 - 2 \, (1 - \Gamma^2) \, (1 - 3 \, \Gamma^2) \, N. \tag{3.1.9}$$

The next two orders are given in reference [17]. The calculation of these terms is an exercise in combinatorics — it can be facilitated by using graphs; this is discussed in the appendices of reference [21]. The effect of mutation is schematically illustrated in figure 3.1.

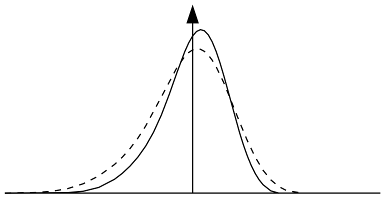


Figure 3.1: Schematic illustration of the effect of mutation on the distribution of fitnesses. The solid curve shows the distribution before mutation while the dashed curve shows the distribution after mutation. The average fitness is reduced while the width of the distribution is increased and the skewness reduce — these changes are all quite small.

The covariances can be found similarly. $V_{1,1}$ is defined as

$$V_{1,1} = \left\langle\!\left\langle K^2_1 \right\rangle\!\right\rangle - \left\langle\!\left\langle K_1 \right\rangle\!\right\rangle^2 = \left\langle\!\!\left\langle \frac{1}{P} \left(\sum_{\alpha=1}^{P} F_\alpha \right)^2 \right\rangle\!\!\right\rangle - \bar{K}^2_1 \tag{3.1.10}$$

so that

$$V^m_{1,1} = \Gamma^2 V_{1,1} \tag{3.1.11}$$

and similarly $V^m_{n,m} = \Gamma^{n+m} \, V_{n,m}$. This simple result arises because, for mutation on an infinite population, there are no fluctuations. If we had used the cumulants κ_n for the finite population distribution as ensemble variables then we would have found additional terms contributing to $v_{n,m}$ as we would expect. These results together with the selection results were used to calculate the evolution of an asexual population in a multiplicative fitness landscape shown in figure 2.2.

The effect of mutation on the correlation is also easy to calculate

$$q^m = \left\langle \left\langle \frac{1}{N} \sum_i S'^\alpha_i \, S'^\beta_i \right\rangle_{mut} \right\rangle_{\mathcal{P}} = \Gamma^2 q. \tag{3.1.12}$$

These equations are exact for all mutation rates. For this problem, selection only depends on the cumulants and not on the details of the sequence. This is not typical, but we might have anticipated that this would be the case here as the change in fitness due to mutation depends only on the fitness of the gene sequence and not on the sequence itself.

3.1.2 Mutation for Other Fitness Functions

The Random Field Paramagnet

The calculation presented above readily generalizes to other fitness functions but with some important differences. To illustrate this we consider the random field paramagnet

$$F_\alpha = \sum_{i=1}^{N} h_i \, S^\alpha_i \tag{3.1.13}$$

where $S^\alpha_i \in \{-1, 1\}$ and the h_i's are random. Following the argument we used above

$$\begin{aligned} \langle F'_\alpha \rangle_{mut} &= \left\langle \sum_{i=1}^{N} h_i \, S'^\alpha_i \right\rangle_{mut} = \Gamma \sum_{i=1}^{N} h_i \, S^\alpha_i = \Gamma \, F_\alpha \\ \left\langle (F'_\alpha)^2 \right\rangle_{mut} &= \Gamma^2 \, F^2_\alpha + (1 - \Gamma^2) \sum_{i=1}^{N} h^2_i \\ \left\langle (F'_\alpha)^3 \right\rangle_{mut} &= \Gamma^3 \, F^2_\alpha + 3\Gamma \, (1 - \Gamma^2) \, F_\alpha \sum_{i=1}^{N} h^2_i - 2\Gamma \, (1 - \Gamma^2) \sum_{i=1}^{N} h^3_i S^\alpha_i \end{aligned}$$

etc. Using these results we find

$$\bar{K}^m_1 = \Gamma \bar{K}_1 \tag{3.1.14}$$

$$\bar{K}^m_2 = \Gamma^2 \bar{K}_2 + (1 - \Gamma^2) \sum_{i=1}^{N} h^2_i \tag{3.1.15}$$

$$\bar{K}^m_3 = \Gamma^3 \bar{K}_3 - 2\Gamma \, (1 - \Gamma^2) \sum_{i=1}^{N} h^3_i \, \langle S^\alpha_i \rangle_{\mathcal{P}} \tag{3.1.16}$$

etc. The fourth order term is given in reference [21]. These results differ form the constant field case in that the fields now appear explicitly. The change in the second cumulant depends on $\sum_i h_i^2$ so we must use information about the statistic of the fields. The most significant difference to the constant field paramagnetic reveals itself first at third order. This is the appearance of the term

$$\sum_{i=1}^{N} h_i^3 \, \langle S_i^\alpha \rangle_{\mathcal{P}} \, .$$

It arises because mutation will depend on whether a few large fields are satisfied (i. e. $h_i \, S_i^\alpha > 0$) or whether many small fields are satisfied. The quantity

$$s_i = \langle S_i^\alpha \rangle_{\mathcal{P}} \tag{3.1.17}$$

is a measure of average **gene satisfaction** and will depend on the fields h_i. To calculate the effect of mutation on the higher cumulants we need to know about the distribution of s_i as a function of h_i. This information cannot be inferred *a priori* from the distribution of fitnesses. In the next section we will use a **maximum entropy ansatz** to estimate it.

The Spin-Glass Chain

We briefly consider mutation in the spin-glass chain as this involves spatial correlation and thus introduces some new terms. The fitness function for the spin-glass chain is defined by

$$F_\alpha = \sum_{i=1}^{N} J_i \, S_i^\alpha \, S_{i+1}^\alpha \tag{3.1.18}$$

where again the alleles take on Ising values, the J_i's are random variables, and where we have set the external fields to zero. This problem is interesting because there are many (typically $2^{N/3}$) local minima to single mutations (see references [31–34] for a discussion of its thermodynamic properties). Following the now familiar procedure, we find

$$\langle F'_\alpha \rangle_{mut} = \left\langle \sum_{i=1}^{N} J_i \, S_i'^\alpha \, S_{i+1}'^\alpha \right\rangle_{mut} = \Gamma^2 F_\alpha,$$

and thus $\bar{K}_1^m = \Gamma^2 \bar{K}_1$. The average fitness is reduced by a factor Γ^2 rather than Γ because mutating an allele breaks two bonds. Continuing to second order we must now separate out two terms: the $i = j$ term (mutations occurring at the same site) and the $i = j \pm 1$ term (mutations occurring on neighbouring sites)

$$\left\langle (F'_\alpha)^2 \right\rangle_{mut} = \Gamma^4 \, F_\alpha^2 + (1 - \Gamma^4) \sum_{i=1}^{N} J_i^2 + 2\Gamma^2 \, (1 - \Gamma^2) \sum_{i=1}^{N} J_i \, J_{i+1} \, S_i^\alpha \, S_{i+2}^\alpha$$

from which we find

$$\bar{K}_2^m = \Gamma^4 \bar{K}_2 + (1 - \Gamma^4) \sum_{i=1}^{N} J_i^2 + 2\Gamma^2 \, (1 - \Gamma^2) \sum_{i=1}^{N} J_i \, J_{i+1} \left\langle S_i^\alpha \, S_{i+2}^\alpha \right\rangle_{\mathcal{P}}$$

This last term introduces spatial correlations (i. e. the probability of neighbouring bond both being satisfied). Ignoring this spatial correlation the new term is approximately proportional to K_1^2/N. A fuller discussion of this calculations is given in reference [21, section 3.2 and appendix D].

3.2 Maximum Entropy Method

We are faced with a common problem in statistical mechanics: we know some macroscopic quantities, **the macro-state**, of a system but the dynamics depends on the **micro-states** of the system. In our case we know the fitness of a gene sequence but we need to know about the distribution of alleles as a function of the fields h_i

$$P(s_i|h_i), \qquad s_i = \langle S_i^\alpha \rangle_{\mathcal{P}} \, .$$

The usual approach is to use a **maximum entropy ansatz**; that is we assume that there is sufficient **microscopic chaos** to ensure that all micro-states consistent with the known macro-state are equally probable (or at least there is no systematic bias). This is an assumption about the underlying microscopic dynamics which is seldom known *a priori* and is usually be justified *a posteriori* by the quality of the agreement achieved. (A discussion of the role of maximum entropy in statistical mechanics is given in section II of Ma's book [35].)

We will restrict our attention to the random field paramagnet. To estimate the distribution $P(s_i|h_i)$ we will average over all configurations of the alleles, S_i^α, in the population subject to the two constraints

$$\begin{aligned} K_1 &= \left\langle \sum_{i=1}^{N} h_i \, S_i^\alpha \right\rangle_{\mathcal{P}} = \sum_{i=1}^{N} h_i \, \langle S_i^\alpha \rangle_{\mathcal{P}} = \sum_{i=1}^{N} h_i \, s_i \\ q &= \left\langle \frac{1}{N} \sum_{i=1}^{N} h_i \, S_i^\alpha S_i^\beta \right\rangle_{\mathcal{P}} = \frac{1}{N} \sum_{i=1}^{N} \langle S_i^\alpha \rangle_{\mathcal{P}}^2 = \frac{1}{N} \sum_{i=1}^{N} s_i^2. \end{aligned} \tag{3.2.1}$$

(There are other constraints we could impose – these are not related to the cumulants but to the founder effect correlations. These correction terms have not, to date, been explored.) To impose these two constraints we introduce Lagrange multipliers x and z so that the probability for a particular set of s_i's is

$$P(\{s_i\}|\{h_i\}) = \frac{1}{Z} \left(\prod_i \mathcal{N}(s_i) \right) \mathrm{e}^{\displaystyle zP \sum_i h_i s_i + P^2 x^2 \sum_i s_i^2/2} \tag{3.2.2}$$

where Z is a normalization factor and the numerical factors multiplying the Lagrange multipliers are chosen for convenience. The Lagrange multipliers x and z must be chosen so that the constraints (3.2.1) are satisfied. The term $\mathcal{N}(s_i)$ is the number of ways that the alleles can be chosen so that $\langle S_i^\alpha \rangle_{\mathcal{P}} = s_i$,

$$\mathcal{N}(s_i) = \operatorname*{Tr}_{\{S_i^\alpha\}} \delta \left(P s_i - \sum_{\alpha=1}^{P} S_i^\alpha \right) = \binom{P}{P(1+s_i)/2}. \tag{3.2.3}$$

(This is just equal to the binomial coefficient). The **entropy** of the state s_i is

$$S(s_i) = \log\left(\mathcal{N}(s_i)\right) \approx -\frac{P}{2} \log\left(1 - s_i^2\right) + \frac{P s_i}{2} \log\left(\frac{1 - s_i}{1 + s_i}\right) \tag{3.2.4}$$

where we have used Stirling's approximation. The product over sites in equation (3.2.2) decouples

$$P(\{s_i\}|\{h_i\}) = \prod_{i=1}^{N} P(s_i).$$

We can use the familiar properties of Gaussians to decouple the quadratic term $P^2x^2s_i^2/2$ to yield

$$P(s_i) = \frac{1}{Z}\frac{\mathrm{d}\eta_i}{\sqrt{2\pi}}\mathrm{e}^{-\eta_i^2/2 + P\,G(s_i,h_i,\eta_i)} \tag{3.2.5}$$

where

$$G(s_i,h_i,\eta_i) = S(s_i)/P + zh_is_i + x\eta_is_i. \tag{3.2.6}$$

For large P the term $\exp(P\,G)$ will be a very sharply peaked function of s_i so that the distribution $P(s_i)$ will be dominated by the maximum of G which is given by

$$\frac{\partial G}{\partial s_i} = \frac{\partial s(s_i)}{\partial s_i} + zh_i + x\eta_i = -\operatorname{atanh}(s_i) + zh_i + x\eta_i = 0 \tag{3.2.7}$$

and thus

$$s_i = \langle\!\langle S_i \rangle\!\rangle = \tanh(zh_i + x\eta_i) \tag{3.2.8}$$

where η_i is a Gaussian distributed variable. The Lagrange multipliers x and z are chosen to satisfy the two constraints

$$K_1 = \sum_i h_i\,\overline{\tanh(zh_i + x\eta)} \tag{3.2.9}$$

$$q = \sum_i \overline{\tanh^2(zh_i + x\eta)} \tag{3.2.10}$$

where the bar denotes an integration over the Gaussian variable η. Thus the average field satisfaction s_i will be a random variable between -1 and 1 whose average and fluctuations depend of K_1 and q. The quantity needed to calculate K_3^m in equation (3.1.16) is

$$\sum_i h_i^3 s_i = \sum_i h_i^3\overline{\tanh(zh_i + x\eta)}.$$

In deriving these results we used a large P approximation in three places: firstly we used Lagrange multipliers which only ensures that the constraints are satisfied on average (we could have evoked the ensemble to justify this approximation); secondly we used Stirling's approximation; and finally we used a saddle point type approximation. *A priori* there appears to be no reason for these approximations to be malicious (this is not the case in selection where the $1/P$ corrections accumulate at each generation). We can test the maximum entropy ansatz independently by looking at the distribution $P(s_i|h_i)$. In figure 3.2 we show the first two moments, $\bar{s}_i$ and $\bar{s_i^2}$, of this distribution as a function of the field strength h_i measured from simulations and computed from the theory. The curves where obtained for the spin-glass chain after 25 generations. The form of the curves are what we would intuitively expect: the alleles associated with small fields have a good chance of being unsatisfied while those for the large fields are almost all satisfied. In addition the fluctuations are much larger for small fields. The cause of the discrepancy between the simulations and theory is anyone's guess. The analysis certainly needs to be repeated and should be done on the random field paramagnet where everything is simpler.

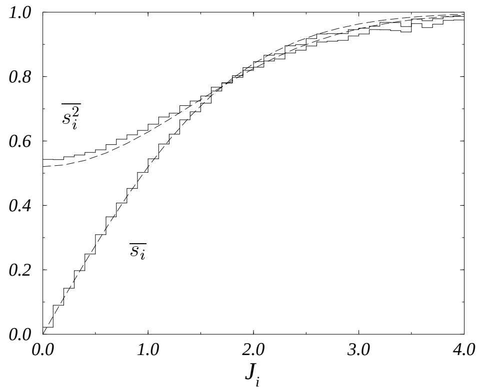


Figure 3.2: The curves show the average of s_i and s_i^2 measured from simulations (solid lines) and the theoretical prediction (dashed curve) plotted against the coupling strength h_i. The parameters are the same as those used in figure 2.

3.3 Crossover or Sexual Recombination.

The most interesting evolution operator is sexual recombination or crossover. This opens up a new way to explore the sequence space, which is not possible except with a population. We being this section by outlining different strategies before we discuss how crossover can be calculated.

3.3.1 Outline

There are many different ways of performing crossover both in Genetic Algorithms and in biology. We will again use the jargon of the GA community as this usually derives from English rather than Latin. The most popular form of crossover is pairwise crossover in which two **parents** $\vec{S}^\alpha$ and $\vec{S}^\beta$ are crossed together to form a child $\vec{S}^\mu$. (We will use the indices $\alpha, \beta, \ldots$ to refer to the parent gene sequences and $\mu, \nu, \ldots$ to refer to their children.) Pairwise crossover can be expressed symbolically by

$$S_i^\mu = \chi_i^\mu\,S_i^\alpha + (1-\chi_i^\mu)\,S_i^\beta \tag{3.3.1}$$

where $\chi_i^\mu \in \{0,1\}$. In Genetic Algorithm it is a frequent practice to take the complementary child

$$S_i^\nu = (1-\chi_i^\mu)\,S_i^\alpha + \chi_i^\mu\,S_i^\beta. \tag{3.3.2}$$

which ensures that both the population size and the alleles in the population remain unchanged.

There are different methods for performing crossover depending on how the χ_i^μ's are chosen. The classic method is **single-point crossover** where

$$\chi_i^\mu = \begin{cases} 1 & \text{if } i \le A_\mu \\ 0 & \text{if } i > A_\mu \end{cases} \tag{3.3.3}$$

and $A_\mu \in \{1, 2, \cdots, N\}$ is usually chosen uniformly. This type of crossover is clearly useful in cases where the fitness function has spatial correlation, as in the three colouring problem or the spin-glass chain. A second popular crossover technique is uniform crossover

$$\chi_i^\mu = \begin{cases} 1 & \text{with probability } a \\ 0 & \text{with probability } 1-a \end{cases} \tag{3.3.4}$$

This is the method of choice for problems with no spatial component such as the paramagnet, the subset-sum problem [28], or generalization in a binary perceptron [22]. The **bias** a controls the amount of mixing. Optimal mixing is achieved when $a = 1/2$, but this is also likely to be the most disruptive and consequently the most costly in terms of the average loss in fitness. In practical GA applications this may be an important parameter to control.

Pairwise crossover is certainly not necessary; any number of parents may be used to create a new child. The case when the alleles of a child is chosen from any parent in the population is known as **bit-simulated crossover**, although it is tempting to call this **orgy crossover**. This strategy really only makes sense in problems where there is no cost involved in crossover. Unfortunately these problems do not have any local minima so are not of great interest. Nevertheless Baum, Boneh and Garrett [36] have proved the rather intriguing result that in one class of problems with no local minima (a version of one's counting where the alleles take on values $\{1, 2, \ldots, L\}$, the fitness being the number of 1's) a GA using bit-simulated crossover could out perform a hill climbing.

Because crossover can be disruptive and cause an average loss in fitness a common practice is to keep some fraction of the population intact to prevent good members being lost. The folklore is that a good crossing fraction is 60%. This will be problem independent and would be worth investigating within this framework.

3.3.2 The Interface Energy

There is often a cost associated with crossover which arises when the fitness of the alleles depend on each other. We will refer to this as the **interface energy**. Nature tries to minimize this cost by allowing crossover to occur only as certain restriction sites. It would be hopeless to allow uniform crossover on base pairs between two alleles coding for the same gene. Thus protein sequences evolve mostly through mutation (at least, this is my understanding).

The interface energy will depend on the fitness of the parents, the number of "bonds" broken and the correlation between the parents. We can illustrate this by considering the spin-glass chain. The fitness of the child μ produced by single-point crossover is

$$\begin{aligned} F_\mu &= \sum_{i=1}^{A-1} J_i S_i^\alpha S_{i+1}^\alpha + J_A S_A^\alpha S_{A+1}^\beta + \sum_{i=A+1}^{N} J_i S_i^\beta S_{i+1}^\beta \\ &= \sum_{i=1}^{A} J_i S_i^\alpha S_{i+1}^\alpha + \sum_{i=A+1}^{N} J_i S_i^\beta S_{i+1}^\beta + \Delta_\mu \end{aligned} \tag{3.3.5}$$

where Δ_μ is the interface energy given by

$$\Delta_\mu = J_A S_A^\alpha (S_{A+1}^\beta - S_{A+1}^\alpha). \tag{3.3.6}$$

Averaging over all possible crossover sites

$$\langle \Delta_\mu \rangle = \frac{1}{N} \sum_{A=1}^{N} J_A \left\langle S_A^\alpha (S_{A+1}^\beta - S_{A+1}^\alpha) \right\rangle_{\mathcal{P}}. \tag{3.3.7}$$

The second term is equal to K_1/N while the first term we can write as

$$\frac{1}{N} \sum_{A=1}^{N} J_A \left\langle S_A^\alpha S_{A+1}^\alpha S_{A+1}^\alpha S_{A+1}^\beta \right\rangle_{\mathcal{P}} \tag{3.3.8}$$

where we have used the identity $(S_{A+1}^\alpha)^2 = 1$. However,

$$\frac{1}{P} \sum_{\beta \neq \alpha} S_{A+1}^\alpha S_{A+1}^\beta \approx q \tag{3.3.9}$$

where q is the average correlation between gene sequences (we ignore fluctuations depending on α and A). Thus

$$\langle \Delta_\mu \rangle = -\frac{K_1(1-q)}{N} \tag{3.3.10}$$

We can obtain corrections to higher order cumulants similarly. The change in the average fitness due to the interface energy is, as advertised, proportional to the average fitness K_1 and $(1-q)$. For the spin-glass chain only a single bond is broken; for problems such as the graph colouring problem where many "bonds" will be broken the interface energy will be proportional to the number of broken bonds. For non-spatial problems there may also be an average cost associated with crossover although there will not be a clearly defined interface.

It may be possible to reduce the interface energy by tailoring the crossover to the problem in question. For example, in graph colouring one could try to "relax" the interface to reduce the interface cost. Alternatively one could try to permute the colours in one of the parents to reduce this cost. I do not know how common this practice is nor has it been studied systematically within the framework.

3.3.3 The Shuffling Problem

Before discussing the mixing effect of crossover we consider a simplified problem, namely **"The Shuffling Problem"**. This is where there is only crossover with no selection or mutation. The problem is to understand how the initial genes are shuffled by the crossover operator. We will denote the initial generation by a Latin index $\vec{S}^a$. To measure the degree of shuffling it is useful to define the **ancestor function** $[\![\cdots]\!]_a$

$$[\![S_i^\alpha(t)]\!]_a = \begin{cases} 1 & \text{if } S_i^\alpha(t) \text{ originated from } a \\ 0 & \text{otherwise} \end{cases} \tag{3.3.11}$$

We denote the number of alleles in gene sequence $\vec{S}^\alpha(t)$ originating in gene sequence $\vec{S}^a(0)$ by the **ancestry variable**

$$K_a^\alpha(t) = \sum_{i=1}^{N} [\![S_i^\alpha(t)]\!]_a. \tag{3.3.12}$$

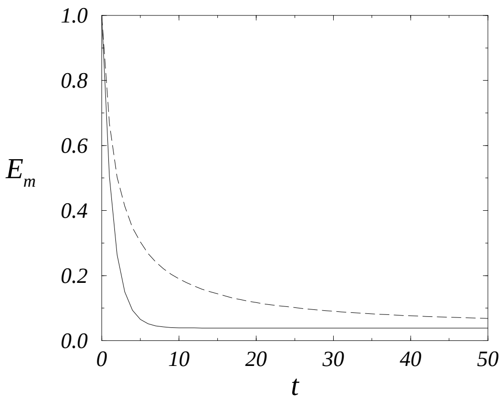


Figure 3.3: The curves show the time evolution of the mixing energy for shuffling using uniform crossover (solid line) and single-point crossover (dashed line). The simulations where for $P = 32$ and $N = 128$ and show the average over 100 runs. The errors are of the same magnitude as the line width. Taken from [29].

Finally we can define a **mixing energy** (density) E_m measuring how well mixed the genes sequences are

$$E_m(t) = \frac{1}{N^2 P} \sum_{a=1}^{P} \sum_{\alpha=1}^{P} \left(K_a^\alpha(t)\right)^2 . \tag{3.3.13}$$

Initially $E_m(0) = 1$ while when the gene sequences are totally random the average mixing energy is $(N+P-1)/NP$. In figure 3.3 we show the mixing energy for uniform crossover (solid curve) and single-point crossover (dashed curve). The mixing produced by uniform crossover is substantially faster than that for single-point crossover.

The curve for uniform crossover can be calculated exactly, which we will do below. The curve for single-point crossover can also be calculated to extremely high accuracy, but we do not derive this here as the theory of single-point crossover is still being developed [29].

To calculate the effect of uniform shuffling we consider the ancestry variable, K_a^μ, for a child produced by uniform crossover

$$K_a^\mu = \sum_{i=1}^{N} \left[\!\left[S_i^\mu \right]\!\right]_a = \sum_{i=1}^{N} \left(\chi_i^\mu \left[\!\left[S_i^\alpha \right]\!\right]_a + (1-\chi_i^\mu) \left[\!\left[S_i^\beta \right]\!\right]_a \right). \tag{3.3.14}$$

To calculate the mixing energy we must look at the square of this quantity

$$(K_a^\mu)^2 = \sum_{i \neq j} \left[\!\left[S_i^\mu \right]\!\right]_a \left[\!\left[S_j^\mu \right]\!\right]_a + \sum_i \left[\!\left[S_i^\mu \right]\!\right]_a \tag{3.3.15}$$

where we have separated out the $i = j$ term and used that fact that

$$\left[\!\left[S_i^\mu \right]\!\right]_a^2 = \left[\!\left[S_i^\mu \right]\!\right]_a$$

which follows because it is a binary valued function. To calculate what happens on average we average over χ_i^μ, using

$$\langle \chi_i^\mu \rangle_\chi = a. \tag{3.3.16}$$

This gives

$$\left\langle (K_a^\mu)^2 \right\rangle_\chi = \sum_{i \neq j} T_i^\mu T_j^\mu + \sum_i T_i^\mu = \left(\sum_i T_i^\mu \right)^2 + \sum_i \left(T_i^\mu - \left(T_i^\mu\right)^2 \right) \tag{3.3.17}$$

where

$$T_i^\mu = a \left[\!\left[S_i^\alpha \right]\!\right]_a + (1-a) \left[\!\left[S_i^\beta \right]\!\right]_a .$$

From this we find

$$\left(T_i^\mu\right)^2 = a^2 \left[\!\left[S_i^\alpha \right]\!\right]_a + (1-a)^2 \left[\!\left[S_i^\beta \right]\!\right]_a$$

since the cross term must vanish (both S_i^α and S_i^β cannot originate from the same ancestor in shuffling). Using the definition of K_a^α

$$\left\langle (K_a^\mu)^2 \right\rangle_\chi = \left(a K_a^\alpha + (1-a) K_a^\beta \right)^2 + a(1-a) \left(K_a^\alpha + K_a^\beta \right). \tag{3.3.18}$$

The average mixing energy at the next generation is

$$\langle E_m(t+1) \rangle_\chi = \frac{1}{N^2 P} \sum_{a=1}^{P} \sum_{\mu=1}^{P} \left(K_a^\mu \right)^2 . \tag{3.3.19}$$

Substituting in equation (3.3.18) and making the replacement

$$\frac{1}{P} \sum_{\mu=1}^{P} = \frac{1}{P(P-1)} \sum_{\alpha=1}^{P} \sum_{\alpha \neq \beta}$$

that is, the children are produced from all pairs of parents, then

$$\langle E_m(t+1) \rangle_\chi = \left(a^2 + (1-a)^2 \right) E_m + \frac{a(1-a)}{P(P-1)N^2} \sum_a \sum_{\alpha \neq \beta} \left(2 K_a^\alpha K_a^\beta + K_a^\alpha + K_a^\beta \right). \tag{3.3.20}$$

Writing

$$\sum_{\alpha \neq \beta} K_a^\alpha K_a^\beta = \left(\sum_\alpha K_a^\alpha \right)^2 - \sum_\alpha (K_a^\alpha)^2$$

we get after collecting together terms

$$\langle E_m(t+1) \rangle_\chi = r E_m(t) + (1-r) E_m^\infty \tag{3.3.21}$$

where

$$r = 1 - \frac{2a(1-a)P}{P-1}, \qquad E_m^\infty = \frac{N+P-1}{NP}. \tag{3.3.22}$$

Since $\langle E_m(t+1) \rangle_\chi$ depends only linearly on $E_m(t)$ we can average over possible crossovers at each time step. Using the initial condition $E_m(0) = 1$ we find the average mixing energy at time t is

$$\langle E_m(t) \rangle_\chi = E_m^\infty - r^t \left(1 - E_m^\infty \right). \tag{3.3.23}$$

Thus for uniform crossover the population reaches the asymptotic mixing energy, E_m^∞, exponentially fast, with a rate $-\log(r)$. As we would expect, this is optimal when half the spins are crossed.

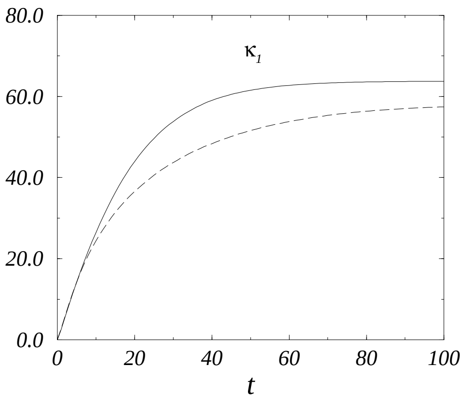


Figure 3.4: The curves show the average fitness for the ideal paramagnet problem. Both uniform crossover (solid curve) and single-point crossover (dashed curve) are shown. The simulations are for $P = 100$, $N = 64$ and $\beta = 0.05$. The results are averaged over 1000 runs. The error bars are the same thickness as the line.

3.3.4 The Mixing Effect of Crossover

We will restrict our attention to the ideal paramagnet,

$$F_\alpha = \sum_{i=1}^{N} S_i^\alpha \tag{3.3.24}$$

with $S_i^\alpha \in \{-1, 1\}$. Generalization of this calculation to the case of the random-field paramagnet is analogous to the case of mutation. Figure 3.4 illustrates the evolution of the average fitness for a Genetic Algorithm with selection and crossover only. The solid curve shows the case when uniform crossover is used and the dashed curve shows the case when single-point crossover is used. The same parameters were used in both cases. Uniform crossover clearly out performs single-point crossover on this problem (there is no interface energy). The reason is simply that uniform crossover mixes the alleles much more effectively than single-point crossover, as discussed in the previous section.

The theory of crossover is current under revision and a proper treatment of single-point crossover has still to be developed (I am working on it [29]). We will discuss just the case of uniform crossover which is far simpler. This calculation follows closely the calculation we just carried out for the shuffling problem.

The fitness function after selection is given by

$$F_\mu = \sum_{i=1}^{N} S_i^\mu = \sum_{i=1}^{N} \left(\chi_i^\mu \, S_i^\alpha + (1 - \chi_i^\mu) \, S_i^\beta \right). \tag{3.3.25}$$

Averaging over all possible crossovers (using equation (3.3.16)) we obtain

$$\left\langle F_\mu \right\rangle_\chi = \sum_{i=1}^{N} \left(a \, S_i^\alpha + (1-a) \, S_i^\beta \right) = a F_\alpha + (1-a) \, F_\beta \equiv \hat{F}_\mu \tag{3.3.26}$$

Averaging over all parents we obtain

$$K_1^c = K_1 \tag{3.3.27}$$

(we omit the average over the ensemble as this is trivial). The first cumulant is unchanged by crossover on average. If we had created the complementary child the first cumulant would be unchanged exactly since the all the alleles in the population would be conserved.

To calculate the change to the second cumulant we consider

$$F_\mu^2 = \sum_{i=1}^{N} \sum_{j \neq i} S_i^\mu \, S_j^\mu + \sum_{i=1}^{N} 1 \tag{3.3.28}$$

where we have separated out the term $i = j$. Averaging over all crossovers

$$\begin{aligned} \left\langle F_\mu^2 \right\rangle_\chi &= \sum_{i=1}^{N} \sum_{j \neq i} \langle S_i^\mu \rangle_\chi \left\langle S_j^\mu \right\rangle_\chi + \sum_{i=1}^{N} 1 \\ &= \left(\sum_{i=1}^{N} \langle S_i^\mu \rangle_\chi \right)^2 - \sum_{i=1}^{N} \left(1 - \langle S_i^\mu \rangle_\chi^2 \right) \\ &= \hat{F}_\mu^2 - \sum_{i=1}^{N} \left(1 - \langle S_i^\mu \rangle_\chi^2 \right). \end{aligned} \tag{3.3.29}$$

From which we find

$$K_2^c = \hat{K}_2 - \sum_{i=1}^{N} \left(1 - \left\langle \langle S_i^\mu \rangle_\chi^2 \right\rangle_{\mathcal{P}} \right). \tag{3.3.30}$$

where $\hat{K}_2$ to be the second cumulant of $\hat{F}_\mu$. To calculate $\hat{K}_n$ we use the generating function

$$\begin{aligned} \hat{G}(z) &= \log \left\langle e^{z \hat{F}_\mu} \right\rangle_{\mathcal{P}} = \log \left\langle e^{z(aF_\alpha + (1-a)F_\beta)} \right\rangle_{\mathcal{P}} \\ &= \log \left\langle e^{z a F_\alpha} \right\rangle_{\mathcal{P}} + \log \left\langle e^{z(1-a)F_\beta} \right\rangle_{\mathcal{P}} \end{aligned} \tag{3.3.31}$$

so that

$$\hat{K}_n = G^{(n)}(0) = \left(a^n + (1-a)^n \right) K_n. \tag{3.3.32}$$

The term $\langle S_i^\mu \rangle_\chi^2$ is equal to

$$\begin{aligned} \langle S_i^\mu \rangle_\chi^2 &= \left(a \, S_i^\alpha + (1-a) \, S_i^\beta \right) \\ &= \left(a^2 + (1-a)^2 \right) + 2a \, (1-a) S_i^\alpha \, S_i^\beta \end{aligned} \tag{3.3.33}$$

Putting equations (3.3.32) and (3.3.33) into equation (3.3.30) we obtain

$$K_2^c = \left(a^2 + (1-a)^2 \right) K_2 + 2a \, (1-a) N \, (1-q) \tag{3.3.34}$$

where q is the average string correlation defined by

$$q = \sum_{i=1}^{N} \left\langle S_i^\alpha \, S_i^\beta \right\rangle_{\mathcal{P}} = \sum_{i=1}^{N} \left\langle (S_i^\alpha)^2 \right\rangle_{\mathcal{P}} . \tag{3.3.35}$$

The n^{th} cumulant result is (postulated to be) of the form

$$K_n^c = r_n K_n + (1 - r_n) K_n^0 \tag{3.3.36}$$

where

$$r_n = \left(a^n + (1-a)^n \right) \tag{3.3.37}$$

and K_n^0 are the **natural cumulants** which one would obtain by a total shuffle of the alleles

$$\begin{aligned} K_2^0 &= \sum_{i=1}^{N} \left(1 - \left\langle (S_i^\alpha)^2 \right\rangle_{\mathcal{P}} \right) \\ K_3^0 &= -2 \sum_{i=1}^{N} \left(\langle S_i^\alpha \rangle_{\mathcal{P}} - \left\langle (S_i^\alpha)^3 \right\rangle_{\mathcal{P}} \right) . \end{aligned} \tag{3.3.38}$$

(I have no proof of equation (3.3.36) although it seems to be true, at least, up to forth order.)

The meaning of this result is that the n^{th} cumulant is moved towards its natural value at a rate $-\log(r_n)$ this is a maximum when $a = 1/2$ (half the genes are swapped) when the rate is $r_n = 2^{n-1}$. The second cumulant might will increase although this may be a small effect (this is a small effect for weak Boltzmann selection). The higher cumulants will all be reduced significantly. The effect of crossover is schematically illustrated in figure 3.5. At first sight the effect of selection might look modest but this is deceptive. Crossover re-populates the high fitness tail which are exactly the solutions of interested. Crossover dramatically increases the effectiveness of the search. Genetic Algorithms learn very quickly; their problem is that the population correlates rapidly and consequently crossover becomes ineffective.

3.3.5 Calculating Correlations

Both crossover and the maximum entropy calculation depend on the correlation between the gene sequences. We now consider how these may be calculated. This approach was first developed by Magnus Rattray [22,28]. We will take a slightly different approach still being developed [29]. To simplify the discussion we will ignore mutation and restrict our attention to the ideal paramagnet.

The correlation in the population is defined as

$$q(t) = \left\langle \frac{1}{N} \sum_{i=1}^{N} S_i^\alpha(t) \, S_i^\beta(t) \right\rangle_{\mathcal{P}} . \tag{3.3.39}$$

This correlation arises through two mechanisms: there is a natural correlation because the average fitness in the population and there is in addition a correlation due to the founder

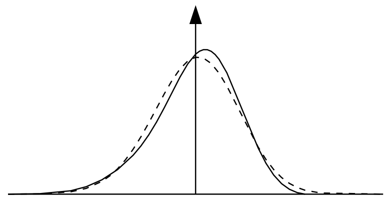


Figure 3.5: Schematic illustration of the effect of crossover on the distribution of fitnesses. The solid curve shows the distribution before crossover while the dashed curve shows the distribution after crossover. The average fitness is slightly reduced due to the interface energy while the width of the distribution is slightly increased. The most significant change is a large reduction in the skewness. The results in a re-population of the tail of high fitness.

effect. We can split up these two mechanisms. We denote by $c_2(t)$ the probability that $S_i^\alpha(t)$ and $S_i^\beta(t)$ originate from a common ancestor

$$c_2(t) \equiv \left\langle \frac{1}{N} \sum_{i=1}^{N} \frac{1}{P} \sum_{a=1}^{P} \left[\!\left[S_i^\alpha(t) \right]\!\right]_a \left[\!\left[S_i^\beta(t) \right]\!\right]_a \right\rangle_{\mathcal{P}} . \tag{3.3.40}$$

If two alleles originate from the came ancestor they, $S_i^\alpha(t) \, S_i^\beta(t) = 1$. If they come from different ancestors their correlation will be that expected for random alleles with an average fitness K_1,

$$\left\langle S_i^\alpha(t) \, S_i^\beta(t) \right\rangle_{\mathcal{P}} = k_1^2(t) \tag{3.3.41}$$

where

$$k_1(t) \equiv \frac{K_1(t)}{N} . \tag{3.3.42}$$

We have actually made an approximation that unrelated and related alleles have the same average fitness. This may not be true. For example, related alleles may be present in the population precisely because they have above average fitness. We may try appealing to microscopic chaos to argue this effect is small, alternatively we could try to treat it more systematically. Here I will assume that it is small. The correlation is then

$$q(t) = c_2(t) + k_1^2(t) \, \left(1 - c_2(t) \right) . \tag{3.3.43}$$

We are left with the problem of determining $c_2(t)$. This is determined by the selection operator as crossover only swaps the alleles but leaves their abundance constant (and we are ignoring mutations). To calculate $c_2(t)$ we introduce the **ancestor number** L_i^a defined as the number of alleles at site i coming from ancestor $\vec{S}^a$,

$$L_i^a(t) = \sum_{\alpha=1}^{P} \left[\!\left[S_i^\alpha(t) \right]\!\right]_a . \tag{3.3.44}$$

By definition

$$c_2(t+1) \; = \; \frac{1}{N} \sum_{i=1}^{N} \frac{1}{P} \sum_{a=1}^{P} \frac{1}{P(P-1)} \sum_{\alpha=1}^{P} \sum_{\alpha \neq \beta} \left[\!\left[S_i^\alpha(t) \right]\!\right]_a \left[\!\left[S_i^\beta(t) \right]\!\right]_a$$

$$= \frac{1}{N}\sum_{i=1}^{N}\frac{1}{P^2(1-P)}\sum_{a=1}^{P}\left(\left(\sum_{\alpha=1}^{P}[\![S_i^\beta(t)]\!]_a\right)^2 - \sum_{\alpha=1}^{P}\sum_{\alpha=1}^{P}\right) \tag{3.3.45}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\frac{1}{P^2(1-P)}\sum_{a=1}^{P} L_i^a(t)(L_i^a(t)-1).$$

But

$$\sum_{a=1}^{P} L_i^a(t) = P$$

since all alleles in the population originate from some ancestor. Denoting the number of times member α is selected by n_α we find that

$$\left(L_i^a(t+1)\right)^2 = \left(\sum_{\alpha=1}^{P} n_\alpha [\![S_i^\alpha(t)]\!]_a\right)^2 \tag{3.3.46}$$

$$= \sum_{\alpha=1}^{P} n_\alpha^2 [\![S_i^\alpha(t)]\!]_a + \sum_{\alpha=1}^{P}\sum_{\alpha\neq\beta} n_\alpha n_\beta [\![S_i^\alpha(t)]\!]_a [\![S_i^\beta(t)]\!]_a . \tag{3.3.47}$$

Appealing once more to microscopic chaos we assume that n_α is independent of $[\![S_i^\alpha(t)]\!]_a$; that is, the number of times a member is chosen is independent of the ancestry of a particular allele. Then averaging over all possible selections

$$\left(L_i^a(t+1)\right)^2 = \langle n_\alpha^2\rangle \sum_{\alpha=1}^{P}[\![S_i^\alpha(t)]\!]_a + \sum_{\alpha\neq\beta} n_\alpha n_\beta [\![S_i^\alpha(t)]\!]_a [\![S_i^\beta(t)]\!]_a \tag{3.3.48}$$

where we have used $\langle n_\alpha\rangle = 1$ (the size of the population is conserved). Substituting this into equation (3.3.45) we find

$$c_2(t+1) = c_2(t) + \frac{\langle n_\alpha^2\rangle - 1}{P(P-1)} \tag{3.3.49}$$

We calculated $\langle n_\alpha^2\rangle$ in section 2.2.4. This calculation generalizes to th higher order statistics of the founder effect need for calculating the effect of crossover on higher order cumulants.

3.4 The Benefits of Sexual Recombination.

Having developed a quantitative formalism for studying micro-evolution we may ask are there any qualitative lessons to be learned. One approach to answering this question is to simplify the formulae we have derived to obtain an approximate dynamics which can be solved in closed form. This approach was carried out in reference [21, section 6.3]. Here we will take a somewhat lighter approach and consider a model problem we have called "**The Basin with a Barrier**" [37]. The fitness function is a modification of the ideal paramagnet

$$F_\alpha = -V(m_\alpha), \qquad m_\alpha \equiv \frac{M_\alpha}{N} \equiv \frac{1}{N}\sum_{i=1}^{N} S_i^\alpha \tag{3.4.1}$$

where the potential $V(m_\alpha)$ is of the form

$$V(m) = \begin{cases} N\left(w(m-m_l)^2/2 + v_l\right) & \text{if } m \le m_b \\ 0 & \text{if } m > m_b \end{cases} \tag{3.4.2}$$

This is shown in the left graph of figure 3.6. There is a local minima at m_l and a barrier at m_b. The problem is to find the **well** or global minima beyond the barrier. We choose this potential for computational convenience. It clearly favours Genetic Algorithms, but the conclusions hold for much smoother barriers.

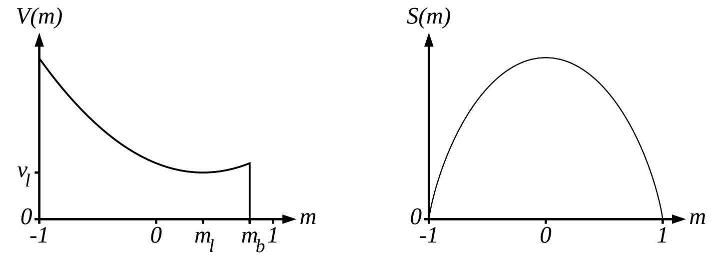


Figure 3.6: Potential and Entropy for the "Basin with a Barrier".

The number of states, $\mathcal{N}(M)$, with magnetization M is given by

$$\mathcal{N}(M) = \operatorname*{Tr}_{\{S_i\}} \delta\left(M - \sum_{i=1}^{N} S_i^\alpha\right) = \binom{N}{(N-2M)/2} \tag{3.4.3}$$

The entropy is

$$S(M) = \log\left(\mathcal{N}(M)\right) \approx -N\left(\frac{1-2m}{2}\log\left(\frac{1-2m}{2}\right) + \frac{1+2m}{2}\log\left(\frac{1+2m}{2}\right)\right) \tag{3.4.4}$$

This is shown in the right graph of figure 3.6. Thus there is a large free energy barrier separating the local and global minima. The problem is to over come this barrier.

This potential is intended to capture a common feature of many hard optimization problems, namely a global minima surrounded by many local minima close to the global minima. There is often a whole hierarchy of such local minima. The further these local minima are from the global minima the higher their energy will have.

3.4.1 Stochastic Hill Climbing

In stochastic hill climbing we consider a "walker" who starts at some randomly chosen configuration and walks around the fitness landscape. Since there is only a single walker we can drop the member index α. For this problem there in no point in annealing the temperature as the walker will find the local minima very quickly. The problem is to overcome the large potential barrier through thermal fluctuations and end up in the well.

We consider a Monte Carlo dynamics in which

$$S_i \to S_i' = \begin{cases} -S_i & \text{with prob. } W(S_i, M) \\ S_i & \text{with prob. } 1 - W(S_i, M) \end{cases} \tag{3.4.5}$$

where, for $m \leq m_b$,

$$W(S_i, M) = \frac{1}{2}\left(1 - \tanh\left(\frac{\beta \Delta E(S_i, M)}{2}\right)\right) \tag{3.4.6}$$

and

$$\Delta E(S_i, M) = E(M - 2\,S_i) - E(M) \tag{3.4.7}$$

This dynamics defines a Markov process (or random walk on line where the transition probabilities depend on where the walker is). In the language of Markov process the global minima is an **absorbing boundary** and we ask what is the expected time to reach this state. To calculate this we define

$$T(M) \equiv \text{expected time to reach the well starting from M.}$$

We can obtain a recursion relation for $T(M)$ by equating $T(M)$ with one plus the expected time to reach the well one time step after being at a point M. Thus

$$\begin{aligned} T(M) &= 1 + \frac{(N-2M)}{2N}\Big[W(-1,M)\,T(M+2) + (1 - W(-1,M))\,T(M)\Big] \\ &\quad + \frac{(N+2M)}{2N}\Big[W(1,M)\,T(M-2) + (1 - W(1,M))\,T(M).\Big] \end{aligned} \tag{3.4.8}$$

The expected time to reach the well one time step after being at a point M is made up of four terms

1. Probability of choosing an allele $S_i^\alpha = -1$ and accepting the mutation.
2. Probability of choosing an allele $S_i^\alpha = -1$ and rejecting the mutation.
3. Probability of choosing an allele $S_i^\alpha = 1$ and accepting the mutation.
4. Probability of choosing an allele $S_i^\alpha = 1$ and rejecting the mutation.

To solve this equation we subtracting $T(M)$ from both sides and define

$$b_M = T(M) - T(M-2). \tag{3.4.9}$$

The recursion relation (3.4.8) reduces to

$$2N - (N - 2M)\,W(-1,M)\,b_{M+2} + (N + 2M)\,W(1,M)\,b_M = 0. \tag{3.4.10}$$

We can solve this recursion relation subject to two boundary conditions

$$T(M) = 0, \qquad M > M_b \tag{3.4.11}$$

and

$$T(2) - T(0) = b_2 = W^{-1}(-1,0). \tag{3.4.12}$$

To find the average time, $\langle T \rangle$, it takes to end up in the global minimum we average over all initial conditions

$$\langle T \rangle = \frac{1}{2^N} \sum_{M=0}^{M \leq M_B} \binom{N}{M} T(M). \tag{3.4.13}$$

This is easy to put on a computer and solve exactly. The time taken to reach the well will depend on the "inverse temperature" β. For the case when $m_l = 0.6$ and $m_b = 0.8$ the optimal average times to reach the well for systems of size $N = 16$, 32 and 64 are $\langle T \rangle = 2389.43$, 869780 and 4.89987×10^{10}.

To calculate how these numbers scale we can evoke a thermodynamic arguement. The walker will to a good approximation thermalize in the local minima so the probability of the walker having a magnetization m_b is given by the ratio of the free energy of the barrier to the free energy at equilibruim m_{equil}

$$\frac{1}{\langle T \rangle} \approx \mathrm{e}^{F(m_b) - F(m_{equil})} \tag{3.4.14}$$

where

$$F(m) = V(m) - TS(m). \tag{3.4.15}$$

The equilibrium magnetization is given by the condition that the free energy is minimized

$$\left.\frac{\partial F(m)}{\partial m}\right|_{m=m_{equil}} = 0. \tag{3.4.16}$$

In the large N limit this becomes exact. Solving these equations numerically we find

$$\langle T \rangle \propto \mathrm{e}^{0.285781\,N}. \tag{3.4.17}$$

3.4.2 Genetic Algorithms

We can ask how a Genetic Algorithm behaves on the same problem. A GA with just mutation and no crossover (which many GA afficionados would refuse to call a GA) would perform terribly. The population would move to the local minima and get stuck. The chances of a member of the population climbing over the barrier is strongly suppressed by the presence of the other members of the population. The time for a population to overcome the barrier is likely to scale as

$$\langle T \rangle \propto \mathrm{e}^{c\,P\,N} \tag{3.4.18}$$

where c is some constant. *Populations are conservative!* They know about their absolute position through the other members of the population in contrast to the single walker who only knows the local gradient.

What about the case of a GA with crossover. Here the situation is entirely different. The population again reaches the bottom of the local minima but now it can jump the energy barrier straight into the well. This is still exponentially difficult because of the entropy barrier, but the situation is far less severe than for the single walker. For large N the time for a GA to find the well will scale something like

$$\frac{1}{\langle T \rangle} \approx \mathrm{e}^{S(m_b) - S(m_l)} \tag{3.4.19}$$

or for the same potential used above

$$\langle T \rangle \propto \mathrm{e}^{0.175319\,N}. \tag{3.4.20}$$

This is of course an optimal asymptotic performance which a real GA would not achieve because of correlations. (A more accurate estimate of the performance of a GA is given in reference [37].) Nevertheless the GA substantially outperforms stochastic hill climbing on this problem.

This potential was hand constructed and clearly favours the GA. In practice a GA outperforms stochastic hill climbing even on fairer potentials with a more rounded barrier. A cluster algorithm in which many alleles are mutated at once might well perform similarly to this ideal GA.

Bibliography

[1] B. Derrida. Random-energy model: An exactly solvable model of disordered systems. *Phys. Rev.*, B24:2613–2626, 1984.

[2] Michael J. Balick and Paul Alan Cox. The Ethnobotanical Approach To Drug Discovery. *Scientific American*, June, 1994.

[3] G. F. Joyce. Directed Molecular Evolution. *Scientific American*, 267(6):90–97, 1992.

[4] J. Moore and F. H. Arnold. Directed Evolution of a para-Nitrobenzyl Esterase for Aqueous and Organic Solvents. *Nature Biotechnology*, 14:458–467, 1996.

[5] M. Mézard, G. Parisi, and M. A. Virasoro. *Spin-Glass Theory and Beyond*, volume 9 of *Lecture Notes in Physics*. World Scientific, Singapore, 1987.

[6] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computation*. Cambridge University Press, Cambridge, UK., 2nd edition, 1992.

[7] John H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press (Ann Arbor), 1975.

[8] David E. Goldberg. *Genetic Algorithms in Search, Optimization & Machine Learning*. Addison-Wesley (Reading, Mass), 1989.

[9] Melanie Mitchell. *An Introduction to Genetic Algorithms*. Complex Adaptive Systems. Bradford, 1996.

[10] Richard Dawkins. *The Blind Watchmaker*. Penguin (Harmondsworth), 1988.

[11] Richard Dawkins. *The Extended Phenotype*. Oxford University press (Oxford), 1982.

[12] S. A. Kauffman. *The Origins of Order*. Oxford University Press, 1993.

[13] H. Flyvbjerg and B. Lautrup. Evolution in a rugged fitness landscape. *Phys. Rev. A.*, 46(10):6714–6723, 1992.

[14] B. Bak, H. Flyvbjerg, and B. Lautrup. Co-evolution in a rugged fitness landscape. *Phys. Rev. A.*, 46(10):6723–6730, 1992.

[15] P. G. Higgs and G. Woodcock. The accumulation of mutations in asexual populations and the structure of genealogical trees in the presence of selection. *J. Math. Biol.*, 33(7):677–702, 1995.

[16] G. Woodcock and P. G. Higgs. Population evolution in a single peak fitness landscape. *J. Theor. Biol.*, 176:61–73, 1996.

[17] A. Prügel-Bennett. Modelling Evolving Populations. Submitted to the *Journal of Theoretical Biology*, 1996.

[18] A. Prügel-Bennett and J. L. Shapiro. An Analysis of Genetic Algorithms Using Statistical Mechanics. *Phys. Rev. Lett.*, 72(9):1305–1309, 1994.

[19] Shapiro J. L., A. Prügel-Bennett, and M. Rattray. A Statistical Mechanical Formulation of the Dynamics of Genetic Algorithms. In T. C. Fogarty, editor, *Lecture notes in Computer Science* **865**, pages 17–27, Berlin, 1994. Springer-Verlag.

[20] Shapiro J. L. and A. Prügel-Bennett. Maximum Entropy Analysis of Genetic Algorithm Operators. In T. C. Fogarty, editor, *Lecture notes in Computer Science* **993**, pages 14–24, Berlin, 1995. Springer-Verlag.

[21] A. Prügel-Bennett and J. L. Shapiro. The Dynamics of a Genetic Algorithm for Simple Random Ising Systems. Accepted for publication in Physica D, 1995.

[22] M. Rattray and J. Shapiro. The dynamics of Genetic Algorithms for a simple learning problem problem. *J. Phys. A.*, 1996. Submitted for publication.

[23] A Nix and Michael D. Vose. Modeling genetic algorithms with markov chains. *Annals of Mathematics and Artificial Intelligence*, 5:79–88, 1991.

[24] Michael D. Vose and Gunar E. Liepins. Punctuated equilibria in genetic search. *Complex Systems*, 5:31–44, 1991.

[25] Michael D. Vose. Modelling simple genetic algorithms. In Darrell Whitley, editor, *Foundations of Genetic Algorithms 2*. Morgan Kaufmann (San Mateo, CA), 1992.

[26] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computation*. Cambridge University Press, Cambridge, UK., 1 edition, 1988.

[27] James Edward Baker. Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the Second International Conference on Genetic Algorithms*. Lawrence Erlbaum Associates (Hillsdale), 1987.

[28] M. Rattray. The dynamics of a genetic algorithm under stabilizing selection. *Complex Systems*, 9(3):213–234, 1995.

[29] A. Prügel-Bennett. Modelling Sexual Reproduction in Evolving Populations. In preparation., 1996.

[30] M. Rattray, A. Prügel-Bennett, and J. L. Shapiro. Selection in Genetic Algorithms. In preparation, 1996.

[31] T Li. Structure of metastable states in random Ising chain. *Phys. Rev. B*, 24(11):6579–6587, 1981.

[32] B. Derrida and E. J. Gardner. Metastable states of a spin-glass chain. *J. Physique*, 47:959, 1986.

[33] G. Grinstein, A. N. Berker, J. Chalupa, and Wortis M. Exact renormalization group with griffiths singularities and spin-glass behaviour: The random ising chain. *Phys. Rev. Lett.*, 36:1508–1511, 1976.

[34] H. H. Chen and S. K. Ma. Low-temperature behaviour of a one-dimensional random ising model. *J. Stat. Phys.*, 29(4):717–746, 1982.

[35] Shang-Keng Ma. *Statistical Mechanics*. Singapore:World Scientific, 1985.

[36] E. B. Baum, D. Boneh, and C. Garrett. Where Genetic Algorithms Excel. NEC preprint, Expanded version of *On Genetic Algorithms*, COLT '95: Preceedings of the Eigth Annual Conference on Computational Learning Theory, Association for Computing Machinery, New York, (1995) pp230–239., 1996.

[37] J. L. Shapiro and A. Prügel-Bennett. When Genetic Algorithms Beats Stochastic Hill Climbing: The Basin with a Barrier. In preparation., 1996.