Statistical Mechanics Theory of Genetic Algorithms

J.L. Shapiro

Department of Computer Science
Manchester University
Manchester, UK
E-mail: *jls@cs.man.ac.uk*

Abstract. This paper gives an introduction to the statistical mechanics method of analysing genetic algorithm dynamics. The goals are to study genetic algorithms acting on specific problems which include realistic features such as: finite population effects, crossover, large search spaces, and realistic cost functions. Statistical mechanics allows one to derive deterministic equations of motion which describe average quantities of the population after selection, mutation, and crossover in terms of those before. The general ideas of this approach are described here, and some details given via consideration of a specific problem. Finally, a description of the literature is given.

1 Overview

One of the difficulties in understanding how genetic algorithms work is that it is difficult to solve specific examples. General notions, such as building-blocks, schema processing, big jumps in state space, and so forth, cannot be used to make quantitative predictions for particular problems. Alternatively, exact formulations of the dynamics, such as the Markov chain formulation, are too complicated to solve in specific cases containing many realistic aspects. This is in contrast with other optimisation methods. For example, if one wonders what effect the step-size has in gradient descent algorithms, one can solve the quadratic case exactly to study this. Such simple models do not exist for genetic algorithms. One road to understanding genetic algorithms is to find a way to study and solve specific examples. These solvable examples should contain some of the features found in realistic problems: large search spaces, relatively small population sizes, strong selection, realistic crossover models, fitness functions containing some complexity found in real problems, and so forth.

The statistical mechanics formulation of genetic algorithms allows one to study the dynamics of genetic algorithms searching for solutions to a number of simple combinatorial problems. This approach addresses some of the challenges raised above. The method can describe problems of realistic size and can include finite population effects, which prove to be essential to understanding how the GA searches. Results are possible for all values of selection pressure, although there are simplifications for weak selection. Both transient and equilibrium behaviour can be modelled under the formalism and fluctuations about the expected trajectory can also be calculated.

Comparisons with simulations show accurate agreement with high significance for most of the problems considered. The examples on which the GA has been analysed using the approach contain some aspects of realistic problems, including multiple optima and interactions between sites. However, they are nonetheless toy problems. In particular, the problems considered so far lack the types of spatial structure which reveal substantial differences between the different forms of crossover.

As an example of what the statistical mechanics formalism can achieve, consider the following problem: there are L files which are to be backed up to a disk of size D. The size of the i-th file is f_i, and the total size of all of the files exceeds the size of the disk. Which of the files should be stored to come closest to filling the disk? More formally, which subset of the f_i's comes closest to summing to D? This is an example of the *subset sum* problem, a simple version of a bin packing problem. It is strictly NP-hard, although for many sets of file sizes it can be solved in polynomial time using standard methods [7].

This problem was analysed using a statistical mechanics approach by Magnus Rattray [14,15]. Figure 1 shows the prediction of this analysis. What is shown in the left-hand figure is the population statistics of total amount stored on disk – the mean amount stored (κ_1) and the variance (κ_2). A measure of the similarity between the strings in the population (φ) is also shown. The right-hand figure shows the squared difference between the amount stored by the best member of the population and the disk size D which is denoted E_{best}. Also shown for comparison is the average of 2000 simulations of a GA. The predictions of the analysis are accurate, but there is some deviation between theory and simulation. Also observe that the best member of the population increases initially, but decreases towards the end of the run. This is due to a decrease in the effective population size due to loss of diversity in the population. This is an important finite population effect which is captured by this approach.

This paper presents an introductory overview of the statistical mechanics approach to analysing genetic algorithms. First, the general ideas of the approach are described. Then, these ideas are illustrated in more detail by consideration of a simple problem. Finally, some of the applications from the literature are briefly described.

2 What is a statistical mechanics of genetic algorithms?

The difficulties in studying the dynamics of evolutionary search come from two sources. First, the dynamics is stochastic – selection involves sampling, and mutation and crossover involve probabilistic changes to the strings. Second, the dynamics acts on the space of possible populations, a space of astronomical dimensionality. We need a way to derive a *deterministic* set of equations describing the evolution which are sufficiently simple to allow for some type of study.

A similar problem is faced by a physicist trying to use molecular properties to understand characteristics of matter, for example the heat capacity of a vessel of water. The state space of the system is of the order of the number of molecules which is

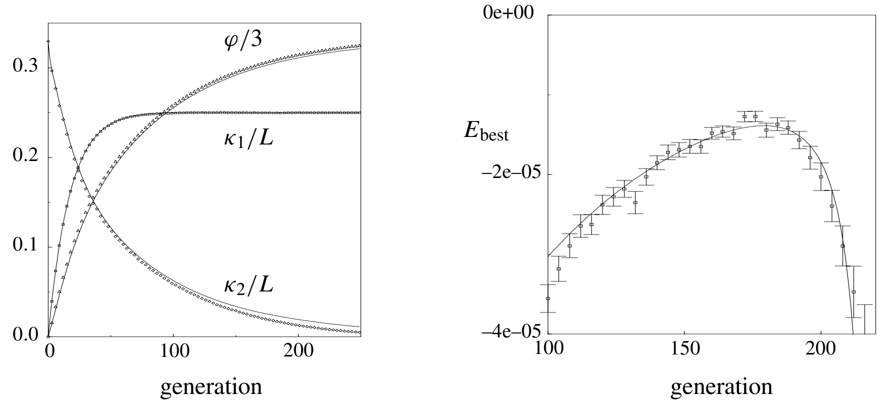


Fig. 1. Comparison of prediction of a statistical mechanics analysis (solid lines) with an average over 2000 runs of a genetic algorithm (symbols) for the subset sum problem. The left figure shows the population mean of the total size of stored files (κ_1), the variance in the population (κ_2), and a measure of the similarity of the strings in the population (φ). The first two are scaled by the string length L. The right figure shows the squared difference between the amount stored by the best member of the population and D. The disk size was $D = L/4$; other parameters were population size of 80, string length $L = 150$, selection strength $s = 0.03$, and uniform crossover was used. Figures are from reference [15].

horrendously large. And, although the dynamics may be deterministic (in classical physics), the initial conditions are unknown, this lack of knowledge introduces randomness.

Both situations can be treated using statistical mechanics. To get a simple description, the system is described in terms of a few *macroscopic* quantities, which are large-scale properties of the system. To get deterministic equations, these macroscopics will be averages over all possible realisation of the randomness.

There is an additional ingredient required to treat systems in this way. In order to do the averaging, one requires knowledge of the probability distribution of the states of the system. However, the formulation does not give access to this distribution. All that is known are the macroscopics which are statistical averages over this distribution. A mechanism is required to get from the known macroscopics other statistical quantities without knowledge of the underlying distribution of states. The statistical mechanics approach is to use an assumption about the distribution of states – that it is a maximum entropy distribution. Justification of this assumption is explained with respect to evolutionary algorithms in [23]; the general justification is the work of E. T. Jaynes [8]. It will discussed briefly in the next two sections.

2.1 Macroscopic versus microscopic descriptions

The first issue is to justify the use of a macroscopic description. Since the genetic operators take a population at time t and produce a new population at time $t + 1$, an obvious level of description would consider precisely which strings are in the population, and would attempt to predict how the constitution of the population changes in time. A description which takes into account precisely which individuals are in the population at each time step is *microscopic*, as it considers all of the small-scaled details of the population.

The advantages of a microscopic approach are clear. It acts directly on the atomic variables of the system, and since all of the information required to predict the changes are available at every time-step, it could be rigorous. However, there are drawbacks. There are an astronomical number of possible populations, many differing in insignificant ways, and a model at this level captures tiny changes as well is significant ones. Typically the types of questions one wishes to ask about GA behaviour involves much higher-level properties - the evolution of the best, the diversity, and so forth - properties which are considerably removed from a full description of all strings. In addition, high-dimensional, non-linear systems are notoriously difficult to analyse, and the dimension of a genetic algorithm described microscopically is horrendously large.

An alternative approach is to describe the GA in terms of statistical properties of the population. This is an example of a *macroscopic* level of description, and the particular statistical properties which are used to characterise the population, are called *macroscopics*. The advantages of a macroscopic formulation are evident: dynamics in a huge number of degrees of freedom is reduced to dynamics in a few quantities. A non linear system of a few degrees of freedom can be readily solved or numerically iterated. The number of macroscopic quantities need not scale with the size of the problem or the population, so perhaps systems of realistic sizes could be tackled. As the macroscopics could chosen to be closely related to the properties which one is interested in studying, it is perhaps easier to learn about the system works at this level of description.

There are obvious drawbacks. Since a small number of global quantities contains far less information about the population than a complete description, no matter how accurate the approach may turn out to be, it will not be derived from first principles, it will be approximate. Since a macroscopic description throws information away, it is essential that the appropriate macroscopics are chosen. Thus, it is not a mechanical, procedural method. Some insight about what is important and what is inessential is required. Finally, since the macroscopic quantities may not provide all of the information required to determine the effects of the genetic operators an inference mechanism is required. The principle of statistical mechanics is that that any quantity which cannot be determined trivially from known macroscopics is computed using a maximum entropy distribution with the known macroscopics as constraints.

2.2 Statistical Mechanics

These ideas are formalised as follows. Image a system which can be in states: $s_1, s_2, \ldots, s_{N_\Omega} \in \Omega$, where Ω is the state space and N_Ω is the number of states in the state space, which is typically huge. For a genetic algorithm, a state could correspond to a particular population with all strings specified. Assume that the system is stochastic, either in the initial conditions, or in the dynamics, or in both.

A natural way to deal with the stochasticity is to express the system in terms of the probability $P(s_i)$ of being in state s_i. For a GA, this probability would correspond to the frequency of runs which result in that population (at the generation in question). This probability expresses as much information about the system as is possible. If one could predict these N_Ω numbers, the $P(s_i)$'s, for all times, one would have a complete characterisation of the system. This is the goal of microscopic formulations.

In a macroscopic description, the $P(s_i)$'s will not be known. What will be know is some average (macroscopic) quantity Q.

$$Q = \sum_{s_i \in \Omega} \tilde{Q}(s_i) P(s_i). \tag{1}$$

Here $\tilde{Q}$ expresses the functional relationship between the state s_i and the value of the statistic associated with that state. Suppose in addition knowledge is required about some other quantity, G,

$$G = \sum_{s_i \in \Omega} \tilde{G}(s_i) P(s_i),$$

since the P's are not known, how can G be found from Q?

The answer is, of course, it cannot without an assumption being introduced. In statistical mechanics, maximum entropy inference is used. Choose the assignment of the $\{P(s_i)\}$ which is consistent with Q and otherwise *has the most 'uncertainty'* (or *assumes the least information*). The uncertainty of a distribution is given by the entropy

$$H = -\sum_{s_i \in \Omega} P(s_i) \log P(s_i). \tag{2}$$

This determines the amount of information required to remove the uncertainty.

The maximum entropy distribution is that which optimises the entropy (2) subject to the constraint (1), i.e. we optimise

$$H = -\sum_{s_i \in \Omega} P(s_i) \log P(s_i) + \lambda \sum_{s_i \in \Omega} \tilde{Q}(s_i) P(s_i).$$

The solution of this is

$$P(s_i) = \frac{e^{-\lambda \tilde{Q}(s_i)}}{\sum_{s_i' \in \Omega} e^{-\lambda \tilde{Q}(s_i')}},$$

where the Lagrange multiplier, λ is chosen to satisfy the constraint

$$\frac{\sum_{s_i \in \Omega} e^{-\lambda \tilde{Q}(s_i)} \tilde{Q}(s_i)}{\sum_{s_i' \in \Omega} e^{-\lambda \tilde{Q}(s_i')}} = Q.$$

The maximum entropy distribution is a reasonable one to assume, because it is the assumption which introduces the least assumptions beyond that which is known, the known macroscopic Q. It also occupies almost all of probability space as $N_\Omega \to \infty$ [6]. But is it an accurate assumption? It will work well if the macroscopic Q explains most of the non-random aspects of the system, and if Q itself does not fluctuate too widely. In physics, the average energy is often chosen as the macroscopic. Since energy is conserved, in many situations all of the randomness is in other combinations of degrees of freedom. If the macroscopic Q does not explain most of the non-random aspects, or if it is fluctuating greatly, additional macroscopic quantities may be required. This occurs, for example, in the vicinity of phase transitions, where order parameters may be required to determine the phase of the system.

2.3 Application to evolutionary dynamics

How this is applied to evolutionary dynamics is the topic of most of this paper. The principles can be illustrated by a simple example. A simple set of macroscopics could be the mean and variance of the fitness in the population averaged over all possible runs. If selection is sufficiently weak, and if every run of the GA is statistically similar (see section 3.3), one can easily derive the effects of selection on these two quantities. This is not so for crossover, however. The effectiveness of crossover will depends on how similar the strings are to each other, a quantity called the correlation. If the strings are very similar crossover will make very little change to the population; if the strings are very dissimilar crossover can have a big effect. One way to predict the effect of crossover is to infer the correlation from the mean and variance. How can this be done, given the fact that it is possible for strings to have similar fitness without being correlated? It can be done using a maximum entropy assumption. Consider all possible populations of the same population size and with the same mean fitness and fitness variance as the GA population. For each possible population, one can compute the average correlation in the population. For each correlation value one can assign an entropy, that is the minus the log of the number of populations which have that value. The correlation value which maximises the entropy is the one which is produced by the most populations. The correlation will be that of a population which is as random as possible given the values of the mean and variance. Crossover effects on such a population can be found.

The situation is slightly more complicated than described above. The above would work in an infinite population. In a finite population, higher cumulants than the mean and variance must be considered to characterise selection. Not all of the correlation can be deduced from the maximum entropy distribution. The increase in correlation due to founder effects must be treated separately. The basic idea is otherwise as described above. We will illustrate this in the next section by working through the one of the simplest example in maximum entropy inference is an issue – an inhomogeneous version of *one-max*.

3 Illustration of the details

3.1 The Problem

The GA will be applied to a class of problems consisting of L binary variables, $x_i \in \{0, 1\}$. An instance of the problem is defined by L 'weights' $J_i \in \mathbb{R}$, which determine the importance of each site. The J's are fixed (for a given problem); the task is to optimise a cost function E over the x's, where

$$E = \sum_{i=1}^{L} J_i x_i.$$

Obviously, this is a trivial problem. The solution is to set x_i to 1 on the sites where J_i is positive, and to 0 at the other sites. It is a generalisation of the *one-max* problem in which each site can contribute a different amount to the cost of a solution (which has important consequences). The methods described in this section can also be applied to cost functions which are non-linear functions of $\sum_{i=1}^{L} J_i x_i$, which includes hard problems. This is described later.

The genetic algorithm will consist of a population of size N of binary strings of length L; the i-th character of a strings is just x_i. Selection will be a form of roulette-wheel selection. At each generation, N parents will be selected from the current population, where the probability of choosing string α is

$$p^\alpha = \frac{W(E^\alpha)}{\sum_\gamma W(E^\gamma)}. \tag{3}$$

Here $W(E^\mu)$ is the 'selection weight' for a string with cost function E^μ. The selection weight depends on the selection operator, for example,

Fitness proportional selection: $W(E) = E$.
Boltzmann selection: $W(E) = \exp(sE)$, where s is a parameter controlling the selection strength.
Ranking selection: $W(E) = \int_{-\infty}^{E} \rho(E')dE'$.

Much of the analysis has used Boltzmann selection, but tournament selection, ranking selection, and Baker selection have also been looked at [15,19].

For reproduction, two parents are selected at random to produce an offspring, and this is done N times. Uniform crossover will be used in which at each site independently the value of the first parent is used with probability a. ($a = 1/2$ corresponds to the usual definition of uniform crossover.) The usual mutation operator will be used. Each site of each of the selected parents will be mutated with a probability γ.

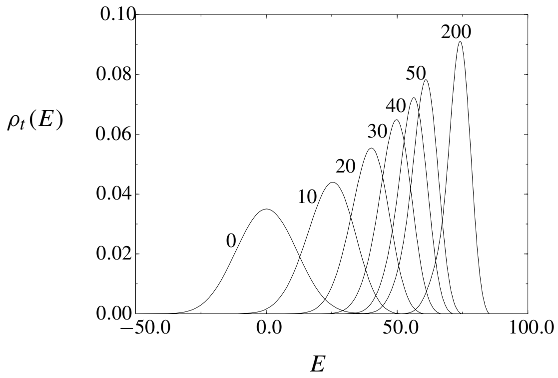


Fig. 2. The distribution of costs in a population, averaged over 10,000 runs of the genetic algorithm. The string length $L = 127$, population size $N = 50$, selection strength $s = 0.1$, mutation rate $\gamma = 1/(2L)$. Crossover probability is 1. Cumulants of this distribution are what we will predict. This figure is from [13].

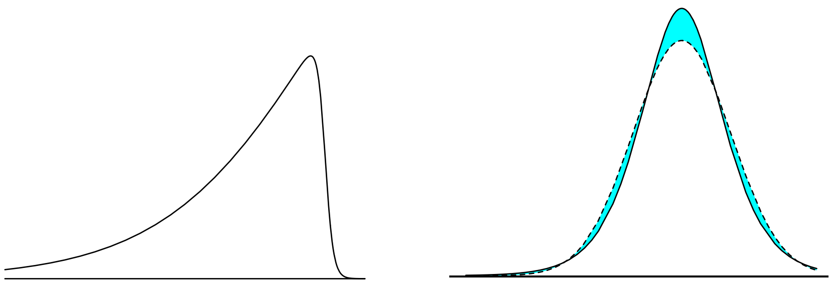
3.2 Choice of Macroscopics

In order to apply the statistical mechanics approach, we need to decide on a few macroscopic quantities with which to characterise the population. We are obviously interested in the the values of the cost function in the population. Figure 3.2 shows the distribution of cost function values in the population at different generations, averaged over many runs of the GA. We would like to evolve *statistics* of this distribution. The average distribution of costs in the population at time t will be denoted $\rho_t(E)$.

The distribution of cost function values in the population will be characterised by its *cumulants* [1]. The first two cumulants are the mean and the variance respectively. The third cumulant is related to the 'skewness' of the population; the fourth cumulant is related to the 'kurtosis' or 'excess'. See figure 3.2 for an illustration of these cumulants. Cumulants are chosen because they average well, and because the complete set of cumulants contains all of the information of the distribution itself. If we take the first few cumulants, we will lose information about the distribution, particularly about the tails. We denote the n-th cumulant by K_n.

The value of the cumulants will vary in each run of the genetic algorithm. In order to make a deterministic system, we will predict the *average* cumulants, denoted κ_n for the n-th one. The averaging is over all runs of the genetic algorithm (or all realizations of the stochasticity). These average cumulants will serve as our first set of of macroscopic quantities.

We will also require a measure of the similarity of the strings in the population. The effect of crossover will depend on the similarity of the strings, as well the expected best member of the population. The obvious measure of similarity is a correlation



(a) (b)

Fig. 3. (a) A distribution with negative skewness ($\kappa_3 < 0$). (b) The solid line is a distribution with positive kurtosis; the dashed line is a Gaussian. It is only barely visible, but the Gaussian falls off more quickly for large values.

measure. The one we use is

$$q = \frac{1}{LN^2} \sum_i \sum_\mu \sum_\nu J_i^2 (2x_i^\mu - 1)(2x_i^\nu - 1).$$

This is the average over all pairs of strings of a measure which counts $+J_i$ where the strings are the same, and $-J_i$ where they differ. This slightly strange form is chosen because it couples most directly to crossover. A completely correlated population will have $q = \sum_i J_i^2/L$; a completely uncorrelated population will have q reduced by a factor $1/N$. Once again, this will need to be averaged over ensembles; the averaged value of q will be denoted φ.

The first goal of the formulation is to produce a set of equations of motion, expressing the macroscopics after selection, mutation, and crossover in terms of those before the operations. At generation t, we have a set of macroscopics:

$$\varphi(t), \kappa_1(t), \kappa_2(t), \ldots, \kappa_n(t),$$

where n is the truncation order (typically 4). Needed are equations taking us to time $t+1$:

$$\begin{pmatrix} \varphi(t) \\ \kappa_1(t) \\ \vdots \\ \kappa_n(t) \end{pmatrix} \xrightarrow{\text{selection}} \begin{pmatrix} \varphi^{(\mathrm{s})}(t) \\ \kappa_1^{(\mathrm{s})}(t) \\ \vdots \\ \kappa_n^{(\mathrm{s})}(t) \end{pmatrix} \xrightarrow{\text{mutation}} \begin{pmatrix} \varphi^{(\mathrm{m})}(t) \\ \kappa_1^{(\mathrm{m})}(t) \\ \vdots \\ \kappa_n^{(\mathrm{m})}(t) \end{pmatrix}$$

$$\xrightarrow{\text{crossover}} \begin{pmatrix} \varphi^{(\mathrm{c})}(t) \\ \kappa_1^{(\mathrm{c})}(t) \\ \vdots \\ \kappa_n^{(\mathrm{c})}(t) \end{pmatrix} = \begin{pmatrix} \varphi(t+1) \\ \kappa_1(t+1) \\ \vdots \\ \kappa_n(t+1) \end{pmatrix}.$$

Deriving these equations is the main topic of this paper. These must then be iterated or solved.

3.3 Selection Dynamics

We need to compute the effect of selection on the macroscopics. Remember that the macroscopics are ensemble averages, averages over all realizations of the stochasticity in the dynamics. Thus, there are two averages which must be done. First, we must average over all ways of averaging over selecting from a given population. Second, we must average over possible populations.

These two stages could be expressed as follows. Let $\{E^\mu\}$ denote the set of costs in a particular population. Let n^μ be the number of times string μ is selected in a particular population. As an example calculation, consider the mean in the population after selection, which equals

$$K_1^{(s)} = \sum_\mu n^\mu E^\mu.$$

The first stage is to average over the n^μ's. The probability of these, $P(\{n^\mu\}|\{E^\mu\})$, is just a multinomial distribution, with probabilities given in (3). Thus, after the first step,

$$K_1^{(s)} = \sum_\mu \frac{W^\mu}{\sum_\nu W^\nu} E^\mu. \tag{4}$$

The second stage is to average over all populations, to average (4) over $P(\{E^\mu\})$. This is a problem, however, because this distribution is unknown. All that is known is the average distribution of costs, averaged over all populations. $\rho_t(E)$. We therefore make the following assumption:

$$P(\{E^\mu\}) = \prod_\mu^N \rho_t(E^\mu). \tag{5}$$

The assumption is that the costs found in a specific population is statistically equivalent to that of ensemble of populations, i.e. that runs of the genetic algorithm are statistically similar. This would be false, for example, if for half the runs, the population converged about one value of the cost function, and for the runs the population converged about another. Covariation across populations is ignored in this approximation, which is in the spirit of the maximum entropy assumption mentioned earlier. A good example where this approximation breaks down is the *Royal Road* function [24].

Making this assumption allows expressions for the average cumulants after selection. For example, the average mean cost function is

$$\kappa_1^{(s)} = \left\langle \frac{\sum_\mu E^\mu W^\mu}{\sum_\mu W^\mu} \right\rangle, \tag{6}$$

where

$$\langle \cdot \rangle = \prod_{\mu}^{N} \int dE^{\mu} \rho_t(E^{\mu}).$$

This is a N-dimensional integral, several ways of computing these integrals have been described elsewhere [13]. I will describe here a simple approach, which is to expand around the infinite population result. In an infinite population, the sum in (6) would be equivalent to an average over all populations. Thus, in the infinite population limit,

$$\kappa_1^{(s)} = \frac{\overline{EW}}{\overline{W}},$$

where $\bar{\cdot}$ denotes averaging over the distribution $\rho_t(E)$. To expand around this, write the expression as

$$\kappa_1^{(s)} = \left\langle \frac{\sum_{\mu} E^{\mu} W^{\mu}}{N\overline{W} + \sum_{\mu} \left(W^{\mu} - \overline{W}\right)} \right\rangle$$

and expand in small $\sum_{\mu} \left(W^{\mu} - \overline{W}\right)$,

$$\kappa_1^{(s)} = \left\langle \frac{\sum_{\mu} E^{\mu} W^{\mu}}{N\overline{W}} \left[1 - \frac{\sum_{\mu} \left(W^{\mu} - \overline{W}\right)}{N\overline{W}} + \frac{\sum_{\mu} \left(W^{\mu} - \overline{W}\right)^2}{\left(N\overline{W}\right)^2} - \dots \right] \right\rangle.$$

This expansion gives the infinite population result as the leading term, and the corrections are powers of the inverse population size. This is an asymptotic expansion, and although it is not obvious from these equations, it is valid if selection is weak. If selection is strong, the population after selection will not be close to that found in an infinite population limit, for example the population might consist of N identical strings. In this limit, the corrections are not powers of N but logarithmic, see [12,13].

To give an explicit example, let $W(E) = 1 + sE$ with selection strength s. If s is very small, selection is almost random; if s is very large, selection is proportional to the cost function value. (This is the weak selection expansion of Boltzmann selection to first order in the selection strength s.) Then, the first two cumulants are

$$\kappa_1^{(s)} = \kappa_1 + s\left(1 - \frac{1}{N}\right)\kappa_2,$$
$$\kappa_2^{(s)} = \left(1 - \frac{1}{N}\right)\kappa_2 + s\left(1 - \frac{3}{N} + \frac{2}{N^2}\right)\kappa_3.$$

The increase in the mean is proportional to the variance in the population, and the variance is reduced through finite population sampling, but can be increased through

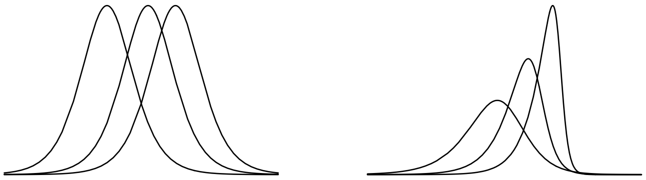


Fig. 4. In an infinite population, selection takes an approximately Gaussian distribution of cost functions and produces an approximately Gaussian distributions with a shifted mean. In a finite population, however, selection introduces higher cumulants because selection can only amplify the frequency of the better strings.

coupling with the third cumulant. Likewise for the next two cumulants,

$$\kappa_3^{(s)} = \left(1 - \frac{3}{N} + \frac{2}{N^2}\right)\kappa_3 + s\left(1 - \frac{7}{N} + \frac{12}{N^2} - \frac{6}{N^3}\right)\kappa_4 - s\frac{6}{N}\left(1 - \frac{1}{N}\right)\kappa_2^2,$$

$$\kappa_4^{(s)} = \left(1 - \frac{7}{N} + \frac{12}{N^2} - \frac{6}{N^3}\right)\kappa_4 - \frac{6}{N}\left(1 - \frac{1}{N}\right)\kappa_2^2 + s\left(1 - \frac{15}{N} + \frac{50}{N^2} - \frac{60}{N^3}\right)\kappa_5 - \frac{60s}{N}\left(1 - \frac{3}{N} + \frac{6}{N^2}\right)\kappa_2\kappa_3.$$

These are reduced through finite population sampling, and can be increase through coupling to the next higher cumulant. However, there is also a non-linear coupling to lower cumulants, for example, the third cumulant couples to the variance squared. This is a finite population effect. It corresponds to the fact that the population becomes more and more skewed (and non-Gaussian) through selection alone. This is because selection can only amplify the strings which are in the population. In an infinite population, all strings are in the population, so the distribution can remain approximately Gaussian while increasing the mean. In a finite population, the population because increasingly non-Gaussian under selection. Selection introduces higher cumulants in a finite population; the stronger the selection pressure, the more important these effects are. This difference is illustrated schematically in figure 3.3.

We still have to consider the effect of selection on the average correlation measure, φ. This is difficult because selection does not act directly on the configuration of strings; it acts on the cost functions. As discussed in section 1, there is no functional relationship between the cost functions and the configurations. Many different populations of strings could give rise to the same set of cost function values, and these different populations could be correlated to differing degrees.

In general, we can use maximum entropy inference to to infer the distribution of strings from the distribution of cost function values [13] as described above as described in section 2.2. If $\{\boldsymbol{x}^\mu\}$ is the set of strings in a population, and $\kappa_i(\{\boldsymbol{x}^\mu\})$

denotes the functional relationship between the cumulant and the strings, the maximum entropy distribution is

$$P_{\mathrm{ME}}(\{\boldsymbol{x}\}|\kappa_1,\ldots) = Z^{-1}e^{-\lambda_1\tilde{\kappa_1}(\{\boldsymbol{x}\})-\lambda_2\tilde{\kappa_2}(\{\boldsymbol{x}\})-\ldots},$$

where Z is the normalisation and the λ's are Lagrange multipliers whose values are chosen to give the known cumulants. This distribution can be used to compute q; averaged over all ensembles would yield the desired macroscopic φ. For calculational reasons, we use

$$P_{\mathrm{ME}}(\boldsymbol{x}|\kappa_1,\varphi) = Z^{-1}e^{-\lambda_1\tilde{\kappa_1}(\boldsymbol{x})-\lambda_2\tilde{\varphi}(\boldsymbol{x})-\ldots}, \tag{7}$$

and use to calculate κ_2.

To apply this to the current task, prediction of the effects of selection on the correlation in the population, we could compute the effect of selection on the cumulants, as described above, and use those to compute the correlation. However, there is a problem with this, as pointed out by Rattray [14]. This approach would underestimate the correlations in a finite population. In a finite population, selection can only increase fitness by duplicating strings, and this correlates the population more than a maximum entropy distribution would predict. As a simple example, consider an cost function with two peaks, and the population in a valley between them. As the cost function increases, the population will climb one or both of the peaks. The maximum entropy distribution would predict that both peaks will be populated, because that gives the maximum randomness. However, in a finite population, only one peak may be populated due to 'founder effects'. A string part way up one of the peaks is generated stochastically, and selection duplicates that string.

In order to treat founder effects, Rattray divided the correlation after selection into two parts,

$$q^{(s)} \to \frac{1}{L}\sum_i J_i^2 \underbrace{\sum_{\mu} n^{\mu}n^{\mu}}_{\substack{\text{founder effect}\\ \text{due to duplication}}} + \underbrace{\sum_{\mu}\sum_{\nu} q^{\mu\nu}n^{\mu}n^{\nu}}_{\substack{\text{natural increase}\\ \text{treated via MaxEnt}}}.$$

The duplication term is treated as the cumulants, by averaging over all selections and populations, and expanding around the infinite population,

$$\begin{aligned}\left\langle \sum_{\mu}\left(\frac{W^{\mu}}{\sum_{\nu} W^{\nu}}\right)^2\right\rangle &\sim \frac{\overline{W^2}}{N\overline{W}^2} + \cdots \\ &\sim \frac{1}{N}\left(1+\kappa_2 s^2+\ldots\right).\end{aligned}$$

The 'natural increase' term would occur in an infinite population, due to the decrease in state space for higher fitness states. This is calculated from the maximum entropy distribution, using the cumulants after selection to predict the correlation after selection, as in (7).

To summarise the effects of selection, an expansion has been produced which predicts the macroscopics after selection in terms of those before selection. The results depend on the selection model; we have written down the equations for $W = 1+sE$. In an infinite population, each cumulant couples to the next higher – the mean increases through the variance, the variance is changed via the skewness, etc. The correlation is decreased through the decrease in state space associated with a population of increased value. Important finite population effects have been computed. Higher cumulants are increased through a coupling with lower ones. So selection produces skewness and kurtosis into the population, which makes the better members of the population worse than would be predicted in a Gaussian approximation. The population is more correlated than predicted by an infinite population theory due to founder effects.

3.4 Crossover Dynamics

Next we calculate the effects of crossover on the macroscopics. First, introduce some new crossover variables,

$$C_i^{\mu\nu} = \begin{cases} 1 & \text{if site } i \text{ in } \mu \text{ is replaced by site } i \text{ in } \nu, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

The strings crossover can be expressed,

$$\boldsymbol{x}^{\mu} \to \boldsymbol{x}^{\mu} + \left(\boldsymbol{x}^{\nu} - \boldsymbol{x}^{\mu}\right) \cdot \boldsymbol{C}^{\mu\nu}.$$

This must be averaged over all realizations of the $\boldsymbol{C}$'s and all populations. The averaging is straight-forward, for example,

$$K_1^{(c)} = \frac{1}{N^2} \sum_{\mu\nu} \sum_i J_i \left[x_i^{\mu} + C_i^{\mu\nu}\left(x_i^{\nu} - x_i^{\mu}\right)\right],$$

which must be averaged over the C's, the x_i's in the population, and over populations. The statistics of C's are determined by the crossover model. For simplicity, consider uniform crossover with probability a of crossing at each site.

It is clear that crossover does not affect the average correlation. For this problem, crossover will not effect the average value of the cost function either. Thus,

$$\kappa_1^{(c)} = \kappa_1,$$
$$\varphi^{(c)} = \varphi.$$

The higher cumulants are found by averaging; the variance obeys

$$\kappa_2^{(c)} = \kappa_2 - 2a(1-a)\left[\kappa_2 - \frac{1}{4}\left(\sum_i J_i^2 - L\varphi\right)\right].$$

The variance can be increased due to crossover, but that increase is reduced if the population is correlated. This shows the importance of using the correlation as one of the macroscopics, and of getting the finite population effects. The correlating duplication effects of selection mediate the ability of crossover to restore variation into the population, and this interplay is predicted quantitatively in this formulation.

The higher cumulants are also driven towards fixed point values by crossover. It is useful to think about the nature of the crossover fixed point. Obviously, if crossover is applied repeatedly, the population evolves to a state in which there is no memory of which string each bit came from. In other words, any correlation between different sites in individual strings is removed. The probability that any bit of any string is 1 is just determined by the frequency of 1's in the population before crossover was repeatedly applied. Such a population is said to be at *linkage equilibrium*. (The linkage equilibrium fixed point is what results from one application of *bit-simulated crossover* [22].)

The effects of crossover on the higher cumulants is expressed as follows:[1]

$$\kappa_3^{(c)} = \kappa_3 - 3a(1-a)\left(\kappa_3 - \kappa_3^{\mathrm{LE}}\right), \tag{9}$$

$$\kappa_4^{(c)} = \kappa_4 - 2a(1-a)(2-a(1-a))\left(\kappa_4 - \kappa_4^{\mathrm{LE}}\right), \tag{10}$$

where the superscript LE denotes the cumulants of the fixed-point of crossover, a distribution where every site is drawn independently from a binomial,

$$P_{\mathrm{LE}}(x_i^\mu = 1) = \frac{1}{N}\sum_\nu x_i^\nu \equiv \tau_i .$$

This does not completely characterise crossover, however, because we do not know the frequency of 1's in the current population. In particular, we do not know how the frequency of 1's depends upon depend upon the site? Each site contributes differently to the performance of the string through, J_i. We do not know how τ_i depends upon (J_i), and neither the average correlation nor the cost function statistics determine this unambiguously.

Again the principle of maximum entropy must be applied. The maximum entropy distribution decouples by site, so it is a linkage equilibrium distribution,

$$P_{\mathrm{ME}}(\{x_i^\mu\}) = \prod_i P_{\mathrm{ME}}(\tau_i | J_i),$$

$$P_{\mathrm{ME}}(\tau_i | J_i) = Z_i^{-1} e^{-\lambda_1 J_i \tau_i - \lambda_2 J_i^2 \tau_i^2} .$$

The λ's are chosen to give the values of κ_1 and φ, and then the third and fourth cumulant of this distribution are the predicted fixed point values to put into (9) and (10). Details of how to solve the maximum entropy model are in [13].

To summarise the effects of crossover, it has no effect on the average correlation in the population, and for this simple problem, no effect of the mean of of the cost

[1] These equations are not exact, but involve approximations I do not wish to explain here.

function, either. The diversity in the population (the variance) is increased typically, but that is related to how similar the strings in the population are. Higher cumulants are return towards the linkage equilibrium values, and these values must be compute for a set of macroscopics from maximum entropy, since they are functions of the frequency of 1's at each site, which is not known from the chosen macroscopics.

3.5 Mutation Dynamics

Mutation is treated in a similarly to crossover. Mutation variables are introduced, as in (8). These are averaged over, and the ensemble average is taken to produce, for mutation rate γ,

$$q^{(m)} = (1-2\gamma)^2 q,$$
$$\kappa_1^{(m)} = \kappa_1 + 2\gamma(\kappa_1^{\text{random}} - \kappa_1),$$
$$\kappa_2^{(m)} = \kappa_2 + 4\gamma(1-\gamma)(\kappa_2^{\text{random}} - \kappa_2).$$

The superscript 'random' denotes cumulants of a random population.

The higher cumulants, like those of crossover, relate to the frequency of 1's as a function of J_i. This is computed from the maximum entropy distribution precisely as in the crossover case. The details are in [13]. The reason that mutation depends on the frequency of 1's as a function of the J_i's is not hard to understand. Consider a string of particular cost E. The effect of mutation will depend on whether the strings obtains this cost by getting a few of the important sites correct (those with J_i's of large magnitude), or by getting many of the less important sites correct. On average it does not matter, but the higher cumulants are effected.

3.6 Putting it together

This process can be iterated to find the dynamics of the GA. At each stage, the nonlinear difference equations of selection, mutation, and crossover must be iterated, along with the maximum entropy model which couples the cumulants to the correlation, and effects of crossover and mutation. Figure 3.6 shows the result of this with four cumulants and φ for small and intermediate values of the mutation rate. The predictions for the cumulants are very accurate, although the predictions for the average correlation is less so. These figures come from reference [15].

For this problem, a picture emerges about the way in which the GA works. Selection increases the mean, reduces the variance, and introduces skewness and higher cumulants. These higher cumulants mean that the high fitness tail of the cost distribution is not very populated, while the low fitness tail is more populated. A schematic picture of this is shown in figure 3.6a. Crossover restores higher cumulants such as skewness to the linkage equilibrium value. This repopulates the high fitness tail, as shown schematically in figure 3.6b. The ability for crossover to have this effect is mediated by the degree of correlation in the population.

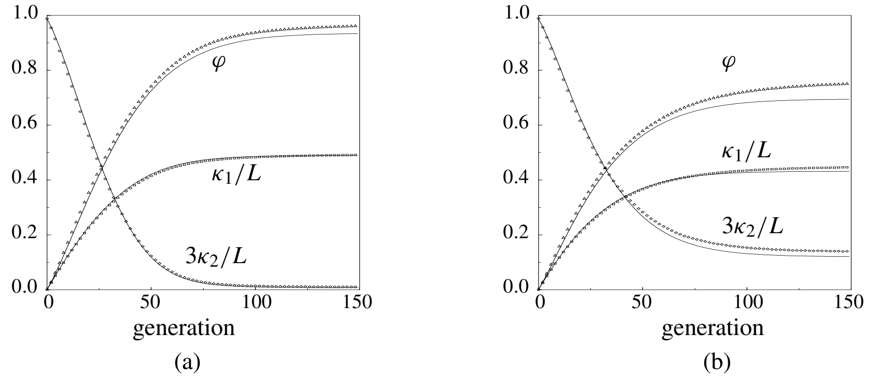


Fig. 5. (a) The theory is compared to averaged results for the paramagnet with a low mutation rate. averaged over 5000 runs, with solid lines showing the theory. The weights were chosen from a uniform distribution in the range [0, 1] so that the optimum was $N/2$ on average. The other parameters were $P = 80$, $N = 120$, $\gamma = 0.001$, $s = 0.25$ and uniform crossover was used with $a = 0.5$. (b) The theory is compared to averaged results for the paramagnet with a moderate mutation rate. averaged over 5000 runs, with solid lines showing the theory. As previous, except that the mutation rate is $\gamma = 0.005$. Figures are from [15].

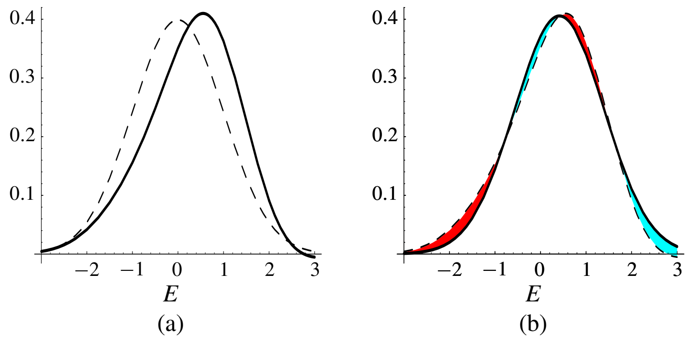


Fig. 6. (a) Dashed line represents the distribution of costs before selection; solid line is that after selection. Although the mean is increased, there is little increase in the high fitness tail. (b) Dashed line represents the distribution of costs before crossover but after selection; solid line is that after crossover. Gray area shows the increase in the better members of the population.

As the statistical mechanics approach is approximate, it is worth pointing out the sources of error and methods for reducing these where possible. The first source is truncation error. Only small number of the cumulants are used. If selection is strong or crossover is weak it may be important to use higher cumulants. This is because selection introduces higher cumulants, but crossover reduces the importance of these. This was the case for genetic algorithm without crossover in the limit of long strings, in which eight cumulants were used [10].

Selection introduces non-linearities, e.g. κ_2^2. The average of κ_2^2 is not the same as square of the average of κ_2, which is an assumption we have made. An approach to treat this more rigorously was developed by Prügel-Bennett [10]. He introduced another set of macroscopic quantities, which are the fluctuations around the cumulants.

An important assumption used in describing the effects of selection is encapsulated in (5). The assumption described in this equation is that there is no significant variation across populations, every run of the genetic algorithm carries outs similar dynamics. This is not the case in many problems. For example, if there is a path to the optimum which is hard to find and the optimum has a very high value, those populations which have found the optimum will be very different from those which have not. This is related to the problem discussed in the previous paragraph, because it is caused by the fluctuations of populations across different runs of the genetic algorithm. This type of problem has been well-studied in the Ph.D. thesis of Erik van Nimwegen [23,24]. He used macroscopics which were not averages over GA runs, and maximum inference applied to populations of a single run. In these types of problems, less of the dynamics can be predicted, but this is because less aspects of it are the same for different runs.

Finally, it is not always the case that the distribution will satisfy maximum entropy. In principle, it may always be possible to use an maximum entropy assumption of the appropriate macroscopics are found, but it may not be clear what these are. From this point of view, the maximum entropy assumption is not a controlled approximation.

4 Application to harder problems: a review of the literature

The statistical mechanics formalism has been applied to a number of problems. Some examples are in these proceedings. In this section, a brief description of some of this work will be given. The goal is to give the reader an introduction to the literature. In addition to the literature described here, there is a literature on the application of statistical mechanics to biological population genetics. See, for example, the review by Baake [2] and references therein.

First, let us return to the problem presented at the beginning of the paper, the subset sum problem. This problem can be analysed as the simple problem studied in the previous section, but now the cost function is *a function* of the weighted sum of the bits, e.g.

$$E = \left(\sum_i f_i x_i - D \right)^2 .$$

The same macroscopics can be used, cumulants of $\sum_i f_i x_i$ and φ. The only difference is that the selection weight is now a more complicated function. If we define

$h = \sum_i f_i x_i$, then the cumulants are statistics of the distribution of h, and the selection weight is a function of $(h - D)^2$.

Details are worked out in [14]. Selection is 'stabilising', which means that it after an initial transient the mean stops changing. Improvement occurs through the reduction in variance. However, as the population correlates, the effective population size decreases. This explains the decrease in the expected best string seen in figure 1b.

Other problems can also be treated as a functions of a weighted-sum of bits. Baum [3] showed that learning in a perceptron with binary (± 1) weights is equivalent to a noisy version on *one-max*. The noise is due to the the fact that the training examples constitute a finite sample of the possible examples. Baum analysed this and a simpler problem, and argued that a genetic algorithm with strong culling selection performed better than a hill-climbing algorithm, but not as well as the theoretical optimum. Rattray and Shapiro [16] analysed this same problem. They showed that noise increases the finite population effects which have a deleterious effect on the search algorithm. They also showed that the number of training examples could be optimised to minimise the computation costs. The training set needs to be large to decrease the noise induced finite-population effects, but the cost of the fitness evaluations increase with the size of the training set. The relationship between these quantities obeys a scaling relationship which allows a simplification of the analysis of this trade-off.

Another problem which can be cast as a function of the weighted sum of bits in the string, involves transitions between barriers in a multi-modal landscape. This was originally considered by Shapiro and Prügel-Bennett [21], and has been analysed more thoroughly by Rogers and Prügel-Bennett [18] (in this volume). Using methods related to those described above, one can analyse an cost function of the form $V(\sum_i x_i)$ where V had two peaks. Since transitions between wells is well understood for thermally-induced transitions, this is a case in which a comparison between genetic algorithms and simulated annealing can be carried out analytically. For certain forms of V, crossover-induced transitions can occur at a substantially higher rate than thermally induced ones; a situation in which a genetic algorithm will outperform simulated annealing.

One topic which is an important test of theory is the question of finding the optimal parameters of genetic algorithm search. Insufficient work has been done on this problem. Since we have derived equations of motion, we can do *greedy* search on parameters, i.e. find the parameters which make the biggest improvement at each time-step.

In weak selection limit shown here, optimal selection strength appears to be ∞. However, a more careful look [13] shows that up to a certain selection strength s^*, the mean increases with little loss diversity ($\sim 1/N$). Beyond that point diversity is lost with almost no increase in the mean,

$$s^* = \alpha\sqrt{\kappa_2/(2\log N)},$$
$$\alpha \sim 0.3.$$

This gives an appropriate rescaling of the selection pressure to compensate for the lose of diversity in the population.

In these simple problems, crossover causes very little disruption. Thus, very strong crossover is optimal. To study how to optimise crossover, very different types of problems will need to be considered.

There have been some considerations of the effects of time-varying mutation rates. We studied a the one-max problem in a special limit [17]. Crossover occurs to linkage equilibrium at each time-step (massive crossover). The diffusion limit is taken, $N \to \infty$, $sN \to$ constant, $\gamma N \to$ constant, discrete dynamics goes to a continuum limit. In this limit, selection + mutation dynamics is linear *even with finite population effects* and the only errors are truncation errors. For this problem, one can choose the mutation rate γ to get the largest increase in κ_1 after mutation + selection. One finds that the mutation rate decays monotonically to zero as the search brings the population towards the optimum.

Bornholdt [4] proposed a phenomenological method for determining the mutation rate at each generation. He defines

$$\tilde{m} = \frac{\langle E E^{(m)} \rangle - \langle E \rangle \langle E^{(m)} \rangle}{\langle E^2 \rangle - \langle E \rangle^2},$$

where $\langle \cdot \rangle$ denotes population and mutation averaging, and $E^{(m)}$ is the objective function after some mutation. (This is like a parent-child correlation function of Manderick.) He also assumes the cumulants after mutation are linear combinations of cumulants before (which is approximately true for one-max):

$$\begin{aligned}
\kappa_1^{(m)} &= \tilde{m}\kappa_1 + (1-\tilde{m})\kappa_1^{\text{random}}, \\
\kappa_2^{(m)} &= \tilde{m}^2\kappa_2 + (1-\tilde{m}^2)\kappa_2^{\text{random}}, \\
\kappa_3^{(m)} &= \tilde{m}^3\kappa_3, \\
\kappa_4^{(m)} &= \tilde{m}^4\kappa_4.
\end{aligned}$$

The optimisation criterion is to maximise the best in the population at each generation. Assume,

$$E_{\text{best}} \approx \kappa_1^{(m)} + \sqrt{2\kappa_2^{(m)} \log \tilde{N}},$$

where

$$\tilde{N} \sim \begin{cases} N & \text{stochastic hill-climbing,} \\ 1 & \text{steady state genetic algorithm.} \end{cases}$$

Then one can solve $\tilde{m}$ as a function of mutation rate for specific problem and optimise it. Bornholdt applied these ideas to the one-max problem, and to Kauffman's NK-landscape which is a rugged fitness function.

4.1 Future work

The statistical mechanics formulation of genetic algorithm dynamics produces accurate predictions on complex problems. But many of the questions which motivates theoretical research into genetic algorithms remain unanswered. One important question involves the relationship between the search parameters and how to optimise them. We need to study weaker forms of crossover on problems in which the trade-offs between disruption and search are more important. We also need to develop a better understanding of possible annealing schedules, for mutation, selection, and especially for crossover.

Genetic algorithms are often used in learning, particularly in reinforcement learning. We have only studied the simplest learning system; these methods could be applied to harder learning problems. Of particular interest would be evolutionary reinforcement learning, and how it performs relative to other reinforcement learning algorithms (a question which has not been adequately studied experimentally, either). It would also be interesting to study co-evolving systems or interacting systems [20]. This is a very difficult problem, but is applicable to a wide range of situations.

References

1. M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions*. Dover Publications, New York, 1967.
2. E. Baake and W. Gabriel. Biological evolution through mutation, selection, and drift: An introductory review. Preprint, available from LANL preprint server as cond-mat/9907372, *http://xxx.lanl.gov/abs/cond-mat/9907372*, 1999.
3. E. Baum, D. Boneh, and C. Grant. *Where Genetic Algorithms Excel*. NEC Research Institute Publication, 1995.
4. S. Bornholdt. Annealing schedule from population dynamics. Preprint, available from LANL preprint server as cond-mat/9903077, *http://xxx.lanl.gov/abs/cond-mat/-9903077*, 1999.
5. M. G. Bulmer. *Mathematical Theory of Quantitative Genetics*. Clarendon Press, Cambridge, 1980.
6. T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 1991.
7. M. R. Garey and D. S. Johnson. *Computers and Intractability – A Guide to the Theory of NP-Completeness*. W.H. Freeman and Co., 1979.
8. E. T. Jaynes. *E.T. Jaynes: papers on probability, statistics, and statistical physics*. Kluwer, London, 1989.
9. H. Mühlenbein and D. Schlierkamp-Voosen. Analysis of selection, mutation and recombination in genetic algorithms. In *Lecture Notes in Computer Science 899*, pages 188–214, 1995.
10. A. Prügel-Bennett. Modelling evolving populations. *J. Theoretical Biology*, 1997.
11. A. Prügel-Bennett. On the long string limit. In W. Banzhaf and C. R. Reeves, editors, *Foundations of Genetic Algorithms 5*, pages 45–56. Morgan Kaufmann Publishers, 1999.

12. A. Prügel-Bennett and J. L. Shapiro. Analysis of genetic algorithms using statistical mechanics. *Physical Review Letters*, 72(9):1305–1309, 1994.
13. A. Prügel-Bennett and J. L. Shapiro. The dynamics of a genetic algorithm for the ising spin-glass chain. *Physica D*, 104:75–114, 1997.
14. L. M. Rattray. Dynamics of a genetic algorithm under stabilizing selection. *Complex Systems*, 9:213–234, 1995.
15. L. M. Rattray. *Modelling the Dynamics of Genetic Algorithms Using Statistic Mechanics*. PhD thesis, Computer Science Dept., University of Manchester, Oxford Road, Manchester M13 9PL, UK, 1996.
16. L. M. Rattray and J. L. Shapiro. The dynamics of a genetic algorithm for a simple learning problem. *Journal of Physics A*, 29:7451–7473, 1996.
17. L. M. Rattray and J. L. Shapiro. Cumulant dynamics in a finite population: Linkage equilibrium theory. Preprint, available in from LANL preprint server as adap-org/9907009, *http://xxx.lanl.gov/abs/adap-org/9907009*, 1999.
18. A. Rogers and A. Prügel-Bennett. The dynamics of a genetic algorithm on a model hard optimization problem. This volume.
19. A. Rogers and A. Prügel-Bennett. Modelling the dynamics of a steady-state genetic algorithm. In W. Banzhaf and C. R. Reeves, editors, *Foundations of Genetic Algorithms 5*, pages 57–68. Morgan Kaufmann Publishers, 1999.
20. J. L. Shapiro. Does data-model co-evolution improve generalization of evolving learners? In A. E. Eiben, Th. Bäck, M. Schoenauer, and H.-P. Schwefel, editors, *Proceedings of the 5th Conference on Parallel Problem Solving from Nature*, pages 140–149. Springer-Verlag, LNCS 1498, 1998.
21. J. L. Shapiro and A. Prügel-Bennett. Genetic algorithm dynamics in a two-well potential. In R. K. Belew and M. D. Vose, editors, *Foundations of Genetic Algorithms 4*, pages 101–116. Morgan Kaufmann Publishers, 1997.
22. G. Syswerda. Simulated crossover in genetic algorithms. In L. D. Whitley, editor, *Foundations of Genetic Algorithms 2*, pages 239–256. Morgan Kaufmann Publishers, 1993.
23. E. van Nimwegen. *The Statistical Dynamics of Epochal Evolution*. PhD thesis, Universiteit Utrecht, The Netherlands, 1999.
24. E. van Nimwegen, J. Chrutchfield, and M. Mitchell. Finite populations induce metastability in evolutionary search. *Physics Letters A*, 229:144–150, 1997.