# 30

# *Efficient Monte Carlo Methods*

This chapter discusses several methods for reducing random walk behaviour in Metropolis methods. The aim is to reduce the time required to obtain effectively independent samples. For brevity, we will say 'independent samples' when we mean 'effectively independent samples'.

## 30.1 Hamiltonian Monte Carlo

The Hamiltonian Monte Carlo method is a Metropolis method, applicable to continuous state spaces, that makes use of gradient information to reduce random walk behaviour. [The Hamiltonian Monte Carlo method was originally called hybrid Monte Carlo, for historical reasons.]

For many systems whose probability $P(\mathbf{x})$ can be written in the form

$$P(\mathbf{x}) = \frac{e^{-E(\mathbf{x})}}{Z}, \tag{30.1}$$

not only $E(\mathbf{x})$ but also its gradient with respect to $\mathbf{x}$ can be readily evaluated. It seems wasteful to use a simple random-walk Metropolis method when this gradient is available – the gradient indicates which direction one should go in to find states with higher probability!

### *Overview of Hamiltonian Monte Carlo*

In the Hamiltonian Monte Carlo method, the state space $\mathbf{x}$ is augmented by *momentum variables* $\mathbf{p}$, and there is an alternation of two types of proposal. The first proposal randomizes the momentum variable, leaving the state $\mathbf{x}$ unchanged. The second proposal changes both $\mathbf{x}$ and $\mathbf{p}$ using simulated Hamiltonian dynamics as defined by the Hamiltonian

$$H(\mathbf{x}, \mathbf{p}) = E(\mathbf{x}) + K(\mathbf{p}), \tag{30.2}$$

where $K(\mathbf{p})$ is a 'kinetic energy' such as $K(\mathbf{p}) = \mathbf{p}^{\mathsf{T}}\mathbf{p}/2$. These two proposals are used to create (asymptotically) samples from the joint density

$$P_H(\mathbf{x}, \mathbf{p}) = \frac{1}{Z_H} \exp[-H(\mathbf{x}, \mathbf{p})] = \frac{1}{Z_H} \exp[-E(\mathbf{x})] \exp[-K(\mathbf{p})]. \tag{30.3}$$

This density is separable, so the marginal distribution of $\mathbf{x}$ is the desired distribution $\exp[-E(\mathbf{x})]/Z$. So, simply discarding the momentum variables, we obtain a sequence of samples $\{\mathbf{x}^{(t)}\}$ that asymptotically come from $P(\mathbf{x})$.

```
g = gradE ( x ) ;                  # set gradient using initial x
E = findE ( x ) ;                  # set objective function too

for l = 1:L                        # loop L times
   p = randn ( size(x) ) ;         # initial momentum is Normal(0,1)
   H = p' * p / 2 + E ;            # evaluate H(x,p)

   xnew = x ;  gnew = g ;
   for tau = 1:Tau                 # make Tau 'leapfrog' steps

      p = p - epsilon * gnew / 2 ; # make half-step in p
      xnew = xnew + epsilon * p ;  # make step in x
      gnew = gradE ( xnew ) ;      # find new gradient
      p = p - epsilon * gnew / 2 ; # make half-step in p

   endfor

   Enew = findE ( xnew ) ;         # find new value of H
   Hnew = p' * p / 2 + Enew ;
   dH = Hnew - H ;                 # Decide whether to accept

   if ( dH < 0 )                accept = 1 ;
   elseif ( rand() < exp(-dH) ) accept = 1 ;
   else                         accept = 0 ;
   endif

   if ( accept )
      g = gnew ;   x = xnew ;    E = Enew ;
   endif
endfor
```

Algorithm 30.1. `Octave` source code for the Hamiltonian Monte Carlo method.

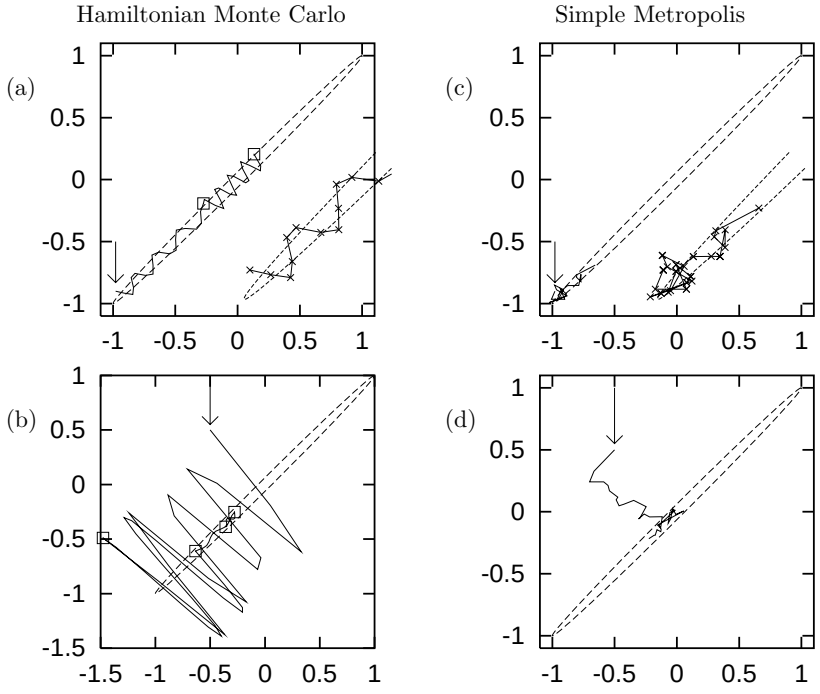



Figure 30.2. (a,b) Hamiltonian Monte Carlo used to generate samples from a bivariate Gaussian with correlation $\rho = 0.998$. (c,d) For comparison, a simple random-walk Metropolis method, given equal computer time.

*Details of Hamiltonian Monte Carlo*

The first proposal, which can be viewed as a Gibbs sampling update, draws a new momentum from the Gaussian density $\exp[-K(\mathbf{p})]/Z_K$. This proposal is always accepted. During the second, dynamical proposal, the momentum variable determines where the state $\mathbf{x}$ goes, and the *gradient* of $E(\mathbf{x})$ determines how the momentum $\mathbf{p}$ changes, in accordance with the equations

$$\dot{\mathbf{x}} = \mathbf{p} \tag{30.4}$$

$$\dot{\mathbf{p}} = -\frac{\partial E(\mathbf{x})}{\partial \mathbf{x}}. \tag{30.5}$$

Because of the persistent motion of $\mathbf{x}$ in the direction of the momentum $\mathbf{p}$ during each dynamical proposal, the state of the system tends to move a distance that goes *linearly* with the computer time, rather than as the square root.

The second proposal is accepted in accordance with the Metropolis rule. If the simulation of the Hamiltonian dynamics is numerically perfect then the proposals are accepted every time, because the total energy $H(\mathbf{x}, \mathbf{p})$ is a constant of the motion and so $a$ in equation (29.31) is equal to one. If the simulation is imperfect, because of finite step sizes for example, then some of the dynamical proposals will be rejected. The rejection rule makes use of the change in $H(\mathbf{x}, \mathbf{p})$, which is zero if the simulation is perfect. The occasional rejections ensure that, asymptotically, we obtain samples $(\mathbf{x}^{(t)}, \mathbf{p}^{(t)})$ from the required joint density $P_H(\mathbf{x}, \mathbf{p})$.

The source code in figure 30.1 describes a Hamiltonian Monte Carlo method that uses the 'leapfrog' algorithm to simulate the dynamics on the function `findE(x)`, whose gradient is found by the function `gradE(x)`. Figure 30.2 shows this algorithm generating samples from a bivariate Gaussian whose energy function is $E(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{A}\mathbf{x}$ with

$$\mathbf{A} = \begin{bmatrix} 250.25 & -249.75 \\ -249.75 & 250.25 \end{bmatrix}, \tag{30.6}$$

corresponding to a variance–covariance matrix of

$$\begin{bmatrix} 1 & 0.998 \\ 0.998 & 1 \end{bmatrix}. \tag{30.7}$$

In figure 30.2a, starting from the state marked by the arrow, the solid line represents two successive trajectories generated by the Hamiltonian dynamics. The squares show the endpoints of these two trajectories. Each trajectory consists of `Tau` $= 19$ 'leapfrog' steps with `epsilon` $= 0.055$. These steps are indicated by the crosses on the trajectory in the magnified inset. After each trajectory, the momentum is randomized. Here, both trajectories are accepted; the errors in the Hamiltonian were only $+0.016$ and $-0.06$ respectively.

Figure 30.2b shows how a sequence of four trajectories converges from an initial condition, indicated by the arrow, that is not close to the typical set of the target distribution. The trajectory parameters `Tau` and `epsilon` were randomized for each trajectory using uniform distributions with means 19 and 0.055 respectively. The first trajectory takes us to a new state, $(-1.5, -0.5)$, similar in energy to the first state. The second trajectory happens to end in a state nearer the bottom of the energy landscape. Here, since the potential energy $E$ is smaller, the kinetic energy $K = \mathbf{p}^2/2$ is necessarily larger than it was at the start of the trajectory. When the momentum is randomized before the third trajectory, its kinetic energy becomes much smaller. After the fourth

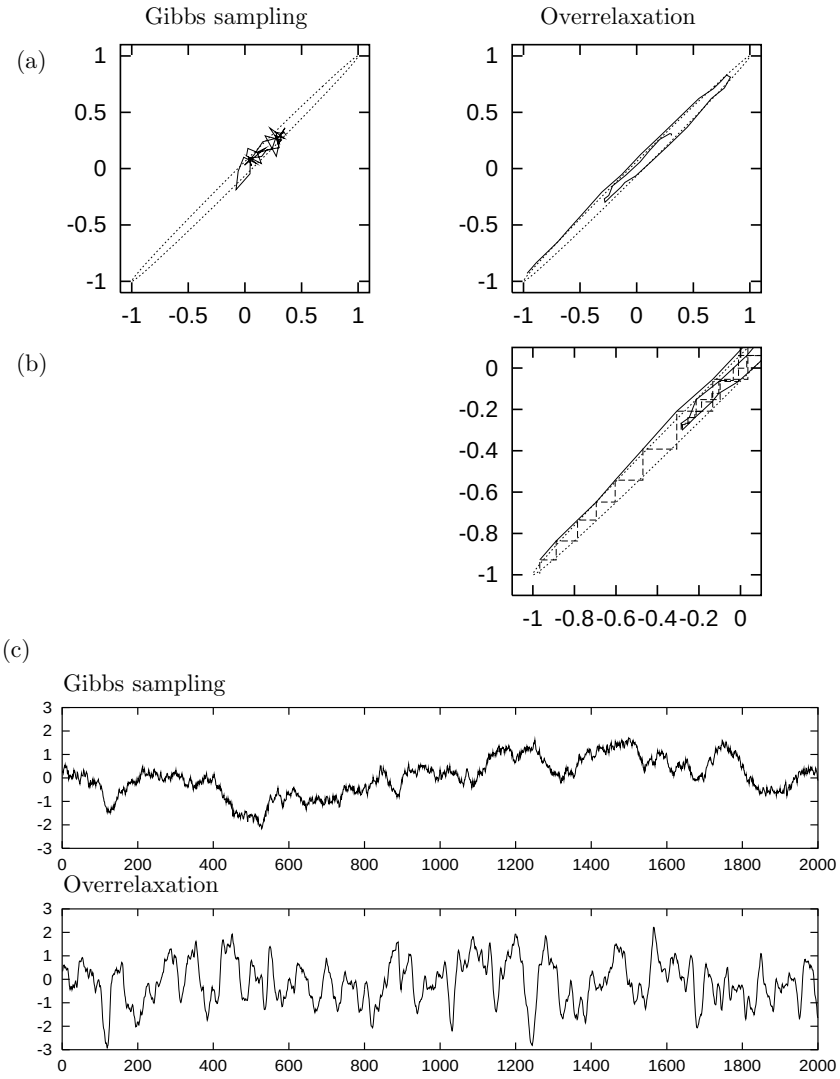



Figure 30.3. Overrelaxation contrasted with Gibbs sampling for a bivariate Gaussian with correlation $\rho = 0.998$. (a) The state sequence for 40 iterations, each iteration involving one update of both variables. The overrelaxation method had $\alpha = -0.98$. (This excessively large value is chosen to make it easy to see how the overrelaxation method reduces random walk behaviour.) The dotted line shows the contour $\mathbf{x}^{\mathrm{T}}\mathbf{\Sigma}^{-1}\mathbf{x} = 1$. (b) Detail of (a), showing the two steps making up each iteration. (c) Time-course of the variable $x_1$ during 2000 iterations of the two methods. The overrelaxation method had $\alpha = -0.89$. (After Neal (1995).)

trajectory has been simulated, the state appears to have become typical of the target density.

Figures 30.2(c) and (d) show a random-walk Metropolis method using a Gaussian proposal density to sample from the same Gaussian distribution, starting from the initial conditions of (a) and (b) respectively. In (c) the step size was adjusted such that the acceptance rate was 58%. The number of proposals was 38 so the total amount of computer time used was similar to that in (a). The distance moved is small because of random walk behaviour. In (d) the random-walk Metropolis method was used and started from the same initial condition as (b) and given a similar amount of computer time.

## ▶ 30.2 Overrelaxation

The method of *overrelaxation* is a method for reducing random walk behaviour in Gibbs sampling. Overrelaxation was originally introduced for systems in which all the conditional distributions are Gaussian.

> An example of a joint distribution that is *not* Gaussian but whose conditional distributions *are* all Gaussian is $P(x, y) = \exp(-x^2y^2 - x^2 - y^2)/Z$.

*Overrelaxation for Gaussian conditional distributions*

In ordinary Gibbs sampling, one draws the new value $x_i^{(t+1)}$ of the current variable $x_i$ from its conditional distribution, ignoring the old value $x_i^{(t)}$. The state makes lengthy random walks in cases where the variables are strongly correlated, as illustrated in the left-hand panel of figure 30.3. This figure uses a correlated Gaussian distribution as the target density.

In Adler's (1981) overrelaxation method, one instead samples $x_i^{(t+1)}$ from a Gaussian that is biased to the *opposite* side of the conditional distribution. If the conditional distribution of $x_i$ is Normal$(\mu, \sigma^2)$ and the current value of $x_i$ is $x_i^{(t)}$, then Adler's method sets $x_i$ to

$$x_i^{(t+1)} = \mu + \alpha(x_i^{(t)} - \mu) + (1 - \alpha^2)^{1/2}\sigma\nu, \tag{30.8}$$

where $\nu \sim$ Normal$(0, 1)$ and $\alpha$ is a parameter between $-1$ and 1, usually set to a negative value. (If $\alpha$ is positive, then the method is called under-relaxation.)

Exercise 30.1.[2] Show that this individual transition leaves invariant the conditional distribution $x_i \sim$ Normal$(\mu, \sigma^2)$.

A single iteration of Adler's overrelaxation, like one of Gibbs sampling, updates each variable in turn as indicated in equation (30.8). The transition matrix $T(\mathbf{x}'; \mathbf{x})$ defined by a complete update of all variables in some fixed order does not satisfy detailed balance. Each individual transition for one coordinate just described *does* satisfy detailed balance – so the overall chain gives a valid sampling strategy which converges to the target density $P(\mathbf{x})$ – but when we form a chain by applying the individual transitions in a fixed sequence, the overall chain is not reversible. This temporal asymmetry is the key to why overrelaxation can be beneficial. If, say, two variables are positively correlated, then they will (on a short timescale) evolve in a directed manner instead of by random walk, as shown in figure 30.3. This may significantly reduce the time required to obtain independent samples.

Exercise 30.2.[3] The transition matrix $T(\mathbf{x}'; \mathbf{x})$ defined by a complete update of all variables in some fixed order does not satisfy detailed balance. If the updates were in a *random order*, then $T$ would be symmetric. Investigate, for the toy two-dimensional Gaussian distribution, the assertion that the advantages of overrelaxation are lost if the overrelaxed updates are made in a random order.

*Ordered Overrelaxation*

The overrelaxation method has been generalized by Neal (1995) whose *ordered overrelaxation* method is applicable to *any* system where Gibbs sampling is used. In ordered overrelaxation, instead of taking one sample from the conditional distribution $P(x_i|\{x_j\}_{j\neq i})$, we create $K$ such samples $x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(K)}$, where $K$ might be set to twenty or so. Often generating $K - 1$ extra samples adds a negligible computational cost to the initial computations required for making the first sample. The points $\{x_i^{(k)}\}$ are then sorted numerically, and the current value of $x_i$ is inserted into the sorted list, giving a list of $K + 1$ points. We give them ranks $0, 1, 2, \ldots, K$. Let $\kappa$ be the rank of the current value of $x_i$ in the list. We set $x_i'$ to the value that is an equal distance from the other end of the list, that is, the value with rank $K - \kappa$. The role played by Adler's $\alpha$ parameter is here played by the parameter $K$. When $K = 1$, we obtain ordinary Gibbs sampling. For practical purposes Neal estimates that ordered overrelaxation may speed up a simulation by a factor of ten or twenty.

## 30.3 Simulated annealing

A third technique for speeding convergence is *simulated annealing*. In simulated annealing, a 'temperature' parameter is introduced which, when large, allows the system to make transitions that would be improbable at temperature 1. The temperature is set to a large value and gradually reduced to 1. This procedure is supposed to reduce the chance that the simulation gets stuck in an unrepresentative probability island.

We asssume that we wish to sample from a distribution of the form

$$P(\mathbf{x}) = \frac{e^{-E(\mathbf{x})}}{Z} \tag{30.9}$$

where $E(\mathbf{x})$ can be evaluated. In the simplest simulated annealing method, we instead sample from the distribution

$$P_T(\mathbf{x}) = \frac{1}{Z(T)}\, e^{-\frac{E(\mathbf{x})}{T}} \tag{30.10}$$

and decrease $T$ gradually to 1.

Often the energy function can be separated into two terms,

$$E(\mathbf{x}) = E_0(\mathbf{x}) + E_1(\mathbf{x}), \tag{30.11}$$

of which the first term is 'nice' (for example, a separable function of $\mathbf{x}$) and the second is 'nasty'. In these cases, a better simulated annealing method might make use of the distribution

$$P'_T(\mathbf{x}) = \frac{1}{Z'(T)}\, e^{-E_0(\mathbf{x}) - E_1(\mathbf{x})/T} \tag{30.12}$$

with $T$ gradually decreasing to 1. In this way, the distribution at high temperatures reverts to a well-behaved distribution defined by $E_0$.

Simulated annealing is often used as an optimization method, where the aim is to find an $\mathbf{x}$ that minimizes $E(\mathbf{x})$, in which case the temperature is decreased to zero rather than to 1.

As a Monte Carlo method, simulated annealing as described above doesn't sample exactly from the right distribution, because there is no guarantee that the probability of falling into one basin of the energy is equal to the total probability of all the states in that basin. The closely related 'simulated tempering' method (Marinari and Parisi, 1992) corrects the biases introduced by the annealing process by making the temperature itself a random variable that is updated in Metropolis fashion during the simulation. Neal's (1998) 'annealed importance sampling' method removes the biases introduced by annealing by computing importance weights for each generated point.

## 30.4 Skilling's multi-state leapfrog method

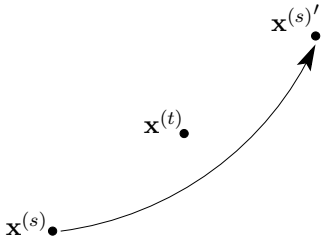
A fourth method for speeding up Monte Carlo simulations, due to John Skilling, has a similar spirit to overrelaxation, but works in more dimensions. This method is applicable to sampling from a distribution over a continuous state space, and the sole requirement is that the energy $E(\mathbf{x})$ should be easy to evaluate. The gradient is not used. This leapfrog method is not intended to be used on its own but rather in sequence with other Monte Carlo operators.

Instead of moving just one state vector $\mathbf{x}$ around the state space, as was the case for all the Monte Carlo methods discussed thus far, Skilling's leapfrog method simultaneously maintains a *set* of $S$ state vectors $\{\mathbf{x}^{(s)}\}$, where $S$

might be six or twelve. The aim is that all $S$ of these vectors will represent independent samples from the same distribution $P(\mathbf{x})$.

Skilling's leapfrog makes a proposal for the new state $\mathbf{x}^{(s)\prime}$, which is accepted or rejected in accordance with the Metropolis method, by leapfrogging the current state $\mathbf{x}^{(s)}$ over another state vector $\mathbf{x}^{(t)}$:

$$\mathbf{x}^{(s)\prime} = \mathbf{x}^{(t)} + (\mathbf{x}^{(t)} - \mathbf{x}^{(s)}) = 2\mathbf{x}^{(t)} - \mathbf{x}^{(s)}. \tag{30.13}$$




All the other state vectors are left where they are, so the acceptance probability depends only on the change in energy of $\mathbf{x}^{(s)}$.

Which vector, $t$, is the partner for the leapfrog event can be chosen in various ways. The simplest method is to select the partner at random from the other vectors. It might be better to choose $t$ by selecting one of the nearest neighbours $\mathbf{x}^{(s)}$ – nearest by any chosen distance function – as long as one then uses an acceptance rule that ensures detailed balance by checking whether point $t$ is still among the nearest neighbours of the new point, $\mathbf{x}^{(s)\prime}$.

*Why the leapfrog is a good idea*

Imagine that the target density $P(\mathbf{x})$ has strong correlations – for example, the density might be a needle-like Gaussian with width $\epsilon$ and length $L\epsilon$, where $L \gg 1$. As we have emphasized, motion around such a density by standard methods proceeds by a slow random walk.

Imagine now that our set of $S$ points is lurking initially in a location that is probable under the density, but in an inappropriately small ball of size $\epsilon$. Now, under Skilling's leapfrog method, a typical first move will take the point a little outside the current ball, perhaps doubling its distance from the centre of the ball. After all the points have had a chance to move, the ball will have increased in size; if all the moves are accepted, the ball will be bigger by a factor of two or so in all dimensions. The rejection of some moves will mean that the ball containing the points will probably have elongated in the needle's long direction by a factor of, say, two. After another cycle through the points, the ball will have grown in the long direction by another factor of two. So the typical distance travelled in the long dimension grows *exponentially* with the number of iterations.

Now, maybe a factor of two growth per iteration is on the optimistic side; but even if the ball only grows by a factor of, let's say, 1.1 per iteration, the growth is nevertheless exponential. It will only take a number of iterations proportional to $\log L / \log(1.1)$ for the long dimension to be explored.

▷ Exercise 30.3.[2, p.400] Discuss how the effectiveness of Skilling's method scales with dimensionality, using a correlated $N$-dimensional Gaussian distribution as an example. Find an expression for the rejection probability, assuming the Markov chain is at equilibrium. Also discuss how it scales with the strength of correlation among the Gaussian variables. [Hint: Skilling's method is invariant under affine transformations, so the rejection probability at equilibrium can be found by looking at the case of a *separable* Gaussian.]

This method has some similarity to the 'adaptive direction sampling' method of Gilks *et al.* (1994) but the leapfrog method is simpler and can be applied to a greater variety of distributions.

## ▶ 30.5 Monte Carlo algorithms as communication channels

It may be a helpful perspective, when thinking about speeding up Monte Carlo methods, to think about the information that is being communicated. Two communications take place when a sample from $P(\mathbf{x})$ is being generated.

First, the selection of a particular $\mathbf{x}$ from $P(\mathbf{x})$ necessarily requires that at least $\log 1/P(\mathbf{x})$ random bits be consumed. [Recall the use of inverse arithmetic coding as a method for generating samples from given distributions (section 6.3).]

Second, the generation of a sample conveys information about $P(\mathbf{x})$ from the subroutine that is able to evaluate $P^*(\mathbf{x})$ (and from any other subroutines that have access to properties of $P^*(\mathbf{x})$).

Consider a dumb Metropolis method, for example. In a dumb Metropolis method, the proposals $Q(\mathbf{x}'; \mathbf{x})$ have nothing to do with $P(\mathbf{x})$. Properties of $P(\mathbf{x})$ are only involved in the algorithm at the acceptance step, when the ratio $P^*(\mathbf{x}')/P^*(\mathbf{x})$ is computed. The channel from the true distribution $P(\mathbf{x})$ to the user who is interested in computing properties of $P(\mathbf{x})$ thus passes through a bottleneck: all the information about $P$ is conveyed by the string of acceptances and rejections. If $P(\mathbf{x})$ were replaced by a different distribution $P_2(\mathbf{x})$, the only way in which this change would have an influence is that the string of acceptances and rejections would be changed. I am not aware of much use being made of this information-theoretic view of Monte Carlo algorithms, but I think it is an instructive viewpoint: if the aim is to obtain information about properties of $P(\mathbf{x})$ then presumably it is helpful to identify the channel through which this information flows, and maximize the rate of information transfer.

Example 30.4. The information-theoretic viewpoint offers a simple justification for the widely-adopted rule of thumb, which states that the parameters of a dumb Metropolis method should be adjusted such that the acceptance rate is about one half. Let's call the acceptance history, that is, the binary string of accept or reject decisions, $\mathbf{a}$. The information learned about $P(\mathbf{x})$ after the algorithm has run for $T$ steps is less than or equal to the information content of $\mathbf{a}$, since all information about $P$ is mediated by $\mathbf{a}$. And the information content of $\mathbf{a}$ is upper-bounded by $TH_2(f)$, where $f$ is the acceptance rate. This bound on information acquired about $P$ is maximized by setting $f = 1/2$.

Another helpful analogy for a dumb Metropolis method is an evolutionary one. Each proposal generates a progeny $\mathbf{x}'$ from the current state $\mathbf{x}$. These two individuals then compete with each other, and the Metropolis method uses a noisy survival-of-the-fittest rule. If the progeny $\mathbf{x}'$ is fitter than the parent (i.e., $P^*(\mathbf{x}') > P^*(\mathbf{x})$, assuming the $Q/Q$ factor is unity) then the progeny replaces the parent. The survival rule also allows less-fit progeny to replace the parent, sometimes. Insights about the rate of evolution can thus be applied to Monte Carlo methods.

Exercise 30.5.[3] Let $\mathbf{x} \in \{0, 1\}^G$ and let $P(\mathbf{x})$ be a separable distribution,

$$P(\mathbf{x}) = \prod_g p(x_g), \tag{30.14}$$

with $p(0) = p_0$ and $p(1) = p_1$, for example $p_1 = 0.1$. Let the proposal density of a dumb Metropolis algorithm $Q$ involve flipping a fraction $m$ of the $G$ bits in the state $\mathbf{x}$. Analyze how long it takes for the chain to

converge to the target density as a function of $m$. Find the optimal $m$ and deduce how long the Metropolis method must run for.

Compare the result with the results for an evolving population under natural selection found in Chapter 19.

The insight that the fastest progress that a standard Metropolis method can make, in information terms, is about one bit per iteration, gives a strong motivation for speeding up the algorithm. This chapter has already reviewed several methods for reducing random-walk behaviour. Do these methods also speed up the rate at which information is acquired?

Exercise 30.6.[4] Does Gibbs sampling, which is a smart Metropolis method whose proposal distributions do depend on $P(\mathbf{x})$, allow information about $P(\mathbf{x})$ to leak out at a rate faster than one bit per iteration? Find toy examples in which this question can be precisely investigated.

Exercise 30.7.[4] Hamiltonian Monte Carlo is another smart Metropolis method in which the proposal distributions depend on $P(\mathbf{x})$. Can Hamiltonian Monte Carlo extract information about $P(\mathbf{x})$ at a rate faster than one bit per iteration?

Exercise 30.8.[5] In importance sampling, the weight $w_r = P^*(\mathbf{x}^{(r)})/Q^*(\mathbf{x}^{(r)})$, a floating-point number, is computed and retained until the end of the computation. In contrast, in the dumb Metropolis method, the ratio $a = P^*(\mathbf{x}')/P^*(\mathbf{x})$ is reduced to a single bit ('is $a$ bigger than or smaller than the random number $u$?'). Thus in principle importance sampling preserves more information about $P^*$ than does dumb Metropolis. Can you find a toy example in which this extra information does indeed lead to faster convergence of importance sampling than Metropolis? Can you design a Markov chain Monte Carlo algorithm that moves around adaptively, like a Metropolis method, and that retains more useful information about the value of $P^*$, like importance sampling?

In Chapter 19 we noticed that an evolving population of $N$ individuals can make faster evolutionary progress if the individuals engage in sexual reproduction. This observation motivates looking at Monte Carlo algorithms in which multiple parameter vectors $\mathbf{x}$ are evolved and interact.

## ▶ 30.6 Multi-state methods

In a multi-state method, multiple parameter vectors $\mathbf{x}$ are maintained; they evolve individually under moves such as Metropolis and Gibbs; there are also interactions among the vectors. The intention is either that eventually all the vectors $\mathbf{x}$ should be samples from $P(\mathbf{x})$ (as illustrated by Skilling's leapfrog method), or that information associated with the final vectors $\mathbf{x}$ should allow us to approximate expectations under $P(\mathbf{x})$, as in importance sampling.

### *Genetic methods*

Genetic algorithms are not often described by their proponents as Monte Carlo algorithms, but I think this is the correct categorization, and an ideal genetic algorithm would be one that can be proved to be a valid Monte Carlo algorithm that converges to a specified density.

I'll use $R$ to denote the number of vectors in the population. We aim to have $P^*(\{\mathbf{x}^{(r)}\}_1^R) = \prod P^*(\mathbf{x}^{(r)})$. A genetic algorithm involves moves of two or three types.

First, individual moves in which one state vector is perturbed, $\mathbf{x}^{(r)} \rightarrow \mathbf{x}^{(r)\prime}$, which could be performed using any of the Monte Carlo methods we have mentioned so far.

Second, we allow crossover moves of the form $\mathbf{x}, \mathbf{y} \rightarrow \mathbf{x}', \mathbf{y}'$; in a typical crossover move, the progeny $\mathbf{x}'$ receives half his state vector from one parent, $\mathbf{x}$, and half from the other, $\mathbf{y}$; the secret of success in a genetic algorithm is that the parameter $\mathbf{x}$ must be encoded in such a way that the crossover of two independent states $\mathbf{x}$ and $\mathbf{y}$, both of which have good fitness $P^*$, should have a reasonably good chance of producing progeny who are equally fit. This constraint is a hard one to satisfy in many problems, which is why genetic algorithms are mainly talked about and hyped up, and rarely used by serious experts. Having introduced a crossover move $\mathbf{x}, \mathbf{y} \rightarrow \mathbf{x}', \mathbf{y}'$, we need to choose an acceptance rule. One easy way to obtain a valid algorithm is to accept or reject the crossover proposal using the Metropolis rule with $P^*(\{\mathbf{x}^{(r)}\}_1^R)$ as the target density – this involves comparing the fitnesses before and after the crossover using the ratio

$$\frac{P^*(\mathbf{x}')P^*(\mathbf{y}')}{P^*(\mathbf{x})P^*(\mathbf{y})}. \tag{30.15}$$

If the crossover operator is reversible then we have an easy proof that this procedure satisfies detailed balance and so is a valid component in a chain converging to $P^*(\{\mathbf{x}^{(r)}\}_1^R)$.

▷ Exercise 30.9.[3] Discuss whether the above two operators, individual variation and crossover with the Metropolis acceptance rule, will give a more efficient Monte Carlo method than a standard method with only one state vector and no crossover.

The reason why the sexual community could acquire information faster than the asexual community in Chapter 19 was because the crossover operation produced diversity with standard deviation $\sqrt{G}$, then the Blind Watchmaker was able to convey lots of information about the fitness function by *killing off* the less fit offspring. The above two operators do *not* offer a speed-up of $\sqrt{G}$ compared with standard Monte Carlo methods because there is no killing. What's required, in order to obtain a speed-up, is two things: multiplication and death; and at least one of these must operate *selectively*. Either we must kill off the less-fit state vectors, or we must allow the more-fit state vectors to give rise to more offspring. While it's easy to sketch these ideas, it is hard to define a valid method for doing it.

Exercise 30.10.[5] Design a birth rule and a death rule such that the chain converges to $P^*(\{\mathbf{x}^{(r)}\}_1^R)$.

I believe this is still an open research problem.

### *Particle filters*

Particle filters, which are particularly popular in inference problems involving temporal tracking, are multistate methods that mix the ideas of importance sampling and Markov chain Monte Carlo. See Isard and Blake (1996), Isard and Blake (1998), Berzuini *et al.* (1997), Berzuini and Gilks (2001), Doucet *et al.* (2001).

## ▶ 30.7 Methods that do not necessarily help

It is common practice to use *many* initial conditions for a particular Markov chain (figure 29.19). If you are worried about sampling well from a complicated density $P(\mathbf{x})$, *can* you ensure the states produced by the simulations are well distributed about the typical set of $P(\mathbf{x})$ by ensuring that the initial points are 'well distributed about the whole state space'?

The answer is, unfortunately, no. In hierarchical Bayesian models, for example, a large number of parameters $\{x_n\}$ may be coupled together via another parameter $\beta$ (known as a hyperparameter). For example, the quantities $\{x_n\}$ might be independent noise signals, and $\beta$ might be the inverse-variance of the noise source. The joint distribution of $\beta$ and $\{x_n\}$ might be

$$\begin{aligned} P(\beta, \{x_n\}) &= P(\beta) \prod_{n=1}^{N} P(x_n|\beta) \\ &= P(\beta) \prod_{n=1}^{N} \tfrac{1}{Z(\beta)} e^{-\beta x_n^2/2}, \end{aligned}$$

where $Z(\beta) = \sqrt{2\pi/\beta}$ and $P(\beta)$ is a broad distribution describing our ignorance about the noise level. For simplicity, let's leave out all the other variables – data and such – that might be involved in a realistic problem. Let's imagine that we want to sample effectively from $P(\beta, \{x_n\})$ by Gibbs sampling – alternately sampling $\beta$ from the conditional distribution $P(\beta|x_n)$ then sampling all the $x_n$ from their conditional distributions $P(x_n|\beta)$. [The resulting marginal distribution of $\beta$ should asymptotically be the broad distribution $P(\beta)$.]

If $N$ is large then the conditional distribution of $\beta$ given any particular setting of $\{x_n\}$ will be tightly concentrated on a particular most-probable value of $\beta$, with width proportional to $1/\sqrt{N}$. Progress up and down the $\beta$-axis will therefore take place by a slow random walk with steps of size $\propto 1/\sqrt{N}$.

So, to the initialization strategy. Can we finesse our slow convergence problem by using initial conditions located 'all over the state space'? Sadly, no. If we distribute the points $\{x_n\}$ widely, what we are actually doing is favouring an initial value of the noise level $1/\beta$ that is *large*. The random walk of the parameter $\beta$ will thus tend, after the first drawing of $\beta$ from $P(\beta|x_n)$, always to start off from one end of the $\beta$-axis.

### Further reading

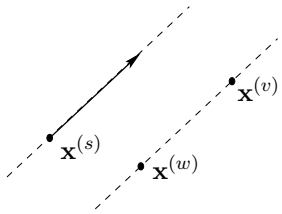
The Hamiltonian Monte Carlo method is reviewed in Neal (1993b). This excellent tome also reviews a huge range of other Monte Carlo methods, including the related topics of simulated annealing and free energy estimation.

## ▶ 30.8 Further exercises

Exercise 30.11.[4] An important detail of the Hamiltonian Monte Carlo method is that the simulation of the Hamiltonian dynamics, while it may be inaccurate, must be perfectly reversible, in the sense that if the initial condition $(\mathbf{x}, \mathbf{p}) \rightarrow (\mathbf{x}', \mathbf{p}')$, then the same simulator must take $(\mathbf{x}', -\mathbf{p}') \rightarrow (\mathbf{x}, -\mathbf{p})$, and the inaccurate dynamics must conserve state-space volume. [The leapfrog method in algorithm 30.1 satisfies these rules.]

Explain why these rules must be satisfied and create an example illustrating the problems that arise if they are not.

Exercise 30.12.[4] A multi-state idea for slice sampling. Investigate the following multi-state method for slice sampling. As in Skilling's multi-state leapfrog method (section 30.4), maintain a set of $S$ state vectors $\{\mathbf{x}^{(s)}\}$. Update one state vector $\mathbf{x}^{(s)}$ by one-dimensional slice sampling in a direction $\mathbf{y}$ determined by picking two other state vectors $\mathbf{x}^{(v)}$ and $\mathbf{x}^{(w)}$ at random and setting $\mathbf{y} = \mathbf{x}^{(v)} - \mathbf{x}^{(w)}$. Investigate this method on toy problems such as a highly-correlated multivariate Gaussian distribution. Bear in mind that if $S - 1$ is smaller than the number of dimensions $N$ then this method will not be ergodic by itself, so it may need to be mixed with other methods. Are there classes of problems that are better solved by this slice sampling method than by the standard methods for picking $\mathbf{y}$ such as cycling through the coordinate axes or picking $\mathbf{u}$ at random from a Gaussian distribution?




## ▶ 30.9 Solutions

Solution to exercise 30.3 (p.395). Consider the spherical Gaussian distribution where all components have mean zero and variance 1. In one dimension, the $n$th, if $x_n^{(1)}$ leapfrogs over $x_n^{(2)}$, we obtain the proposed coordinate

$$(x_n^{(1)})' = 2x_n^{(2)} - x_n^{(1)}. \tag{30.16}$$

Assuming that $x_n^{(1)}$ and $x_n^{(2)}$ are Gaussian random variables from Normal$(0, 1)$, $(x_n^{(1)})'$ is Gaussian from Normal$(0, \sigma^2)$, where $\sigma^2 = 2^2 + (-1)^2 = 5$. The change in energy contributed by this one dimension will be

$$\frac{1}{2}\left[(2x_n^{(2)} - x_n^{(1)})^2 - (x_n^{(1)})^2\right] = 2(x_n^{(2)})^2 - 2x_n^{(2)}x_n^{(1)} \tag{30.17}$$

so the typical change in energy is $2\langle(x_n^{(2)})^2\rangle = 2$. This positive change is bad news. In $N$ dimensions, the typical change in energy when a leapfrog move is made, at equilibrium, is thus $+2N$. The probability of acceptance of the move scales as

$$e^{-2N}. \tag{30.18}$$

This implies that Skilling's method, as described, is not effective in very high-dimensional problems – at least, not once convergence has occurred. Nevertheless it has the impressive advantage that its convergence properties are independent of the strength of correlations between the variables – a property that not even the Hamiltonian Monte Carlo and overrelaxation methods offer.

# *About Chapter 31*

Some of the neural network models that we will encounter are related to Ising models, which are idealized magnetic systems. It is not essential to understand the statistical physics of Ising models to understand these neural networks, but I hope you'll find them helpful.

Ising models are also related to several other topics in this book. We will use exact tree-based computation methods like those introduced in Chapter 25 to evaluate properties of interest in Ising models. Ising models offer crude models for binary images. And Ising models relate to two-dimensional constrained channels (c.f. Chapter 17): a two-dimensional bar-code in which a black dot may not be completely surrounded by black dots and a white dot may not be completely surrounded by white dots is similar to an antiferromagnetic Ising model at low temperature. Evaluating the entropy of this Ising model is equivalent to evaluating the capacity of the constrained channel for conveying bits.

If you would like to jog your memory on statistical physics and thermodynamics, you might find Appendix B helpful. I also recommend the book by Reif (1965).

# 31

# *Ising Models*

An Ising model is an array of spins (e.g., atoms that can take states $\pm 1$) that are magnetically coupled to each other. If one spin is, say, in the $+1$ state then it is energetically favourable for its immediate neighbours to be in the same state, in the case of a ferromagnetic model, and in the opposite state, in the case of an antiferromagnet. In this chapter we discuss two computational techniques for studying Ising models.

Let the state $\mathbf{x}$ of an Ising model with $N$ spins be a vector in which each component $x_n$ takes values $-1$ or $+1$. If two spins $m$ and $n$ are neighbours we write $(m, n) \in \mathcal{N}$. The coupling between neighbouring spins is $J$. We define $J_{mn} = J$ if $m$ and $n$ are neighbours and $J_{mn} = 0$ otherwise. The energy of a state $\mathbf{x}$ is

$$E(\mathbf{x}; J, H) = -\left[\frac{1}{2}\sum_{m,n} J_{mn} x_m x_n + \sum_n H x_n\right], \tag{31.1}$$

where $H$ is the applied field. If $J > 0$ then the model is ferromagnetic, and if $J < 0$ it is antiferromagnetic. We've included the factor of $1/2$ because each pair is counted twice in the first sum, once as $(m, n)$ and once as $(n, m)$. At equilibrium at temperature $T$, the probability that the state is $\mathbf{x}$ is

$$P(\mathbf{x}|\beta, J, H) = \frac{1}{Z(\beta, J, H)} \exp[-\beta E(\mathbf{x}; J, H)], \tag{31.2}$$

where $\beta = 1/k_{\mathrm{B}}T$, $k_{\mathrm{B}}$ is Boltzmann's constant, and

$$Z(\beta, J, H) \equiv \sum_{\mathbf{x}} \exp[-\beta E(\mathbf{x}; J, H)]. \tag{31.3}$$

*Relevance of Ising models*

Ising models are relevant for three reasons.

Ising models are important first as models of magnetic systems that have a phase transition. The theory of universality in statistical physics shows that all systems with the same dimension (here, two), and the same symmetries, have equivalent critical properties, i.e., the scaling laws shown by their phase transitions are identical. So by studying Ising models we can find out not only about magnetic phase transitions but also about phase transitions in many other systems.

Second, if we generalize the energy function to

$$E(\mathbf{x}; \mathbf{J}, \mathbf{h}) = -\left[\frac{1}{2}\sum_{m,n} J_{mn} x_m x_n + \sum_n h_n x_n\right], \tag{31.4}$$

where the couplings $J_{mn}$ and applied fields $h_n$ are not constant, we obtain a family of models known as 'spin glasses' to physicists, and as 'Hopfield

networks' or 'Boltzmann machines' to the neural network community. In some of these models, all spins are declared to be neighbours of each other, in which case physicists call the system an 'infinite range' spin glass, and networkers call it a 'fully connected' network.

Third, the Ising model is also useful as a statistical model in its own right.

In this chapter we will study Ising models using two different computational techniques.

### *Some remarkable relationships in statistical physics*

We would like to get as much information as possible out of our computations. Consider for example the heat capacity of a system, which is defined to be

$$C \equiv \frac{\partial}{\partial T} \bar{E}, \tag{31.5}$$

where

$$\bar{E} = \frac{1}{Z} \sum_{\mathbf{x}} \exp(-\beta E(\mathbf{x}))\, E(\mathbf{x}). \tag{31.6}$$

To work out the heat capacity of a system, we might naively guess that we have to increase the temperature and measure the energy change. Heat capacity, however, is intimately related to energy *fluctuations* at constant temperature. Let's start from the partition function,

$$Z = \sum_{\mathbf{x}} \exp(-\beta E(\mathbf{x})). \tag{31.7}$$

The mean energy is obtained by differentiation with respect to $\beta$:

$$\frac{\partial \ln Z}{\partial \beta} = \frac{1}{Z} \sum_{\mathbf{x}} -E(\mathbf{x}) \exp(-\beta E(\mathbf{x})) = -\bar{E}. \tag{31.8}$$

A further differentiation spits out the variance of the energy:

$$\frac{\partial^2 \ln Z}{\partial \beta^2} = \frac{1}{Z} \sum_{\mathbf{x}} E(\mathbf{x})^2 \exp(-\beta E(\mathbf{x})) - \bar{E}^2 = \langle E^2 \rangle - \bar{E}^2 = \mathrm{var}(E). \tag{31.9}$$

But the heat capacity is also the derivative of $\bar{E}$ with respect to temperature:

$$\frac{\partial \bar{E}}{\partial T} = -\frac{\partial}{\partial T} \frac{\partial \ln Z}{\partial \beta} = -\frac{\partial^2 \ln Z}{\partial \beta^2} \frac{\partial \beta}{\partial T} = -\mathrm{var}(E)(-1/k_{\mathrm{B}} T^2). \tag{31.10}$$

So for any system at temperature $T$,

$$C = \frac{\mathrm{var}(E)}{k_{\mathrm{B}} T^2} = k_{\mathrm{B}} \beta^2 \, \mathrm{var}(E). \tag{31.11}$$

Thus if we can observe the variance of the energy of a system at equilibrium, we can estimate its heat capacity.

I find this an almost paradoxical relationship. Consider a system with a finite set of states, and imagine heating it up. At high temperature, all states will be equiprobable, so the mean energy will be essentially constant and the heat capacity will be essentially zero. But on the other hand, with all states being equiprobable, there will certainly be fluctuations in energy. So how can the heat capacity be related to the fluctuations? The answer is in the words 'essentially zero' above. The heat capacity is not quite zero at high temperature, it just tends to zero. And it tends to zero as $\frac{\mathrm{var}(E)}{k_{\mathrm{B}} T^2}$, with

the quantity $\mathrm{var}(E)$ tending to a constant at high temperatures. This $1/T^2$ behaviour of the heat capacity of finite systems at high temperatures is thus very general.

The $1/T^2$ factor can be viewed as an accident of history. If only temperature scales had been defined using $\beta = \frac{1}{k_{\mathrm{B}}T}$, then the definition of heat capacity would be

$$C^{(\beta)} \equiv \frac{\partial \bar{E}}{\partial \beta} = \mathrm{var}(E), \tag{31.12}$$

and heat capacity and fluctuations would be identical quantities.

▷ Exercise 31.1.[2] [We will call the entropy of a physical system $S$ rather than $H$, while we are in a statistical physics chapter; we set $k_{\mathrm{B}} = 1$.]

The entropy of a system whose states are $\mathbf{x}$, at temperature $T = 1/\beta$, is

$$S = \sum p(\mathbf{x})[\ln 1/p(\mathbf{x})] \tag{31.13}$$

where

$$p(\mathbf{x}) = \frac{1}{Z(\beta)} \exp[-\beta E(\mathbf{x})]\,. \tag{31.14}$$

(a) Show that

$$S = \ln Z(\beta) + \beta \bar{E}(\beta) \tag{31.15}$$

where $\bar{E}(\beta)$ is the mean energy of the system.

(b) Show that

$$S = -\frac{\partial F}{\partial T}, \tag{31.16}$$

where the free energy $F = -kT \ln Z$ and $kT = 1/\beta$.

## ▶ 31.1 Ising models – Monte Carlo simulation

In this section we study two-dimensional planar Ising models using a simple Gibbs sampling method. Starting from some initial state, a spin $n$ is selected at random, and the probability that it should be $+1$ given the state of the other spins and the temperature is computed,

$$P(+1|b_n) = \frac{1}{1 + \exp(-2\beta b_n)}, \tag{31.17}$$

where $\beta = 1/k_{\mathrm{B}}T$ and $b_n$ is the local field

$$b_n = \sum_{m:(m,n)\in\mathcal{N}} J x_m + H. \tag{31.18}$$

[The factor of 2 appears in equation (31.17) because the two spin states are $\{+1, -1\}$ rather than $\{+1, 0\}$.] Spin $n$ is set to $+1$ with that probability, and otherwise to $-1$; then the next spin to update is selected at random. After sufficiently many iterations, this procedure converges to the equilibrium distribution (31.2). An alternative to the Gibbs sampling formula (31.17) is the Metropolis algorithm, in which we consider the change in energy that results from flipping the chosen spin from its current state $x_n$,

$$\Delta E = 2 x_n b_n, \tag{31.19}$$

and adopt this change in configuration with probability

$$P(\mathrm{accept}; \Delta E, \beta) = \begin{cases} 1 & \Delta E \leq 0 \\ \exp(-\beta \Delta E) & \Delta E > 0. \end{cases} \tag{31.20}$$

This procedure has roughly double the probability of accepting energetically unfavourable moves, so may be a more efficient sampler – but at very low temperatures the relative merits of Gibbs sampling and the Metropolis algorithm may be subtle.

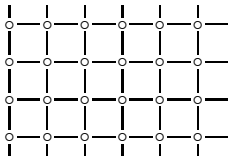


Figure 31.1. Rectangular Ising model.

### *Rectangular geometry*

I first simulated an Ising model with the rectangular geometry shown in figure 31.1, and with periodic boundary conditions. A line between two spins indicates that they are neighbours. I set the external field $H = 0$ and considered the two cases $J = \pm 1$ which are a ferromagnet and antiferromagnet respectively.

I started at a large temperature ($T = 33, \beta = 0.03$) and changed the temperature every $I$ iterations, first decreasing it gradually to $T = 0.1, \beta = 10$, then increasing it gradually back to a large temperature again. This procedure gives a crude check on whether 'equilibrium has been reached' at each temperature; if not, we'd expect to see some hysteresis in the graphs we plot. It also gives an idea of the reproducibility of the results, if we assume that the two runs, with decreasing and increasing temperature, are effectively independent of each other.

At each temperature I recorded the mean energy per spin and the standard deviation of the energy, and the mean square value of the magnetization $m$,

$$m = \frac{1}{N} \sum_n x_n. \tag{31.21}$$

One tricky decision that has to be made is how soon to start taking these measurements after a new temperature has been established; it is difficult to detect 'equilibrium' – or even to give a clear definition of a system's being 'at equilibrium'! [But in Chapter 32 we will see a solution to this problem.] My crude strategy was to let the number of iterations at each temperature, $I$, be a few hundred times the number of spins $N$, and to discard the first 1/3 of those iterations. With $N = 100$, I found I needed more than 100 000 iterations to reach equilibrium at any given temperature.

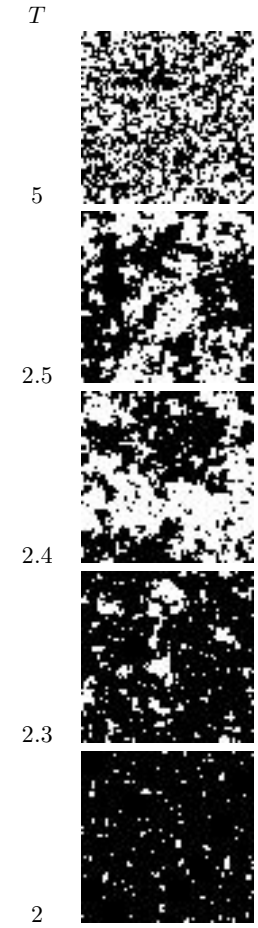



Figure 31.2. Sample states of rectangular Ising models with $J = 1$ at a sequence of temperatures $T$.

### *Results for small $N$ with $J = 1$.*

I simulated an $l \times l$ grid for $l = 4, 5, \ldots, 10, 40, 64$. Let's have a quick think about what results we expect. At low temperatures the system is expected to be in a ground state. The rectangular Ising model with $J = 1$ has two ground states, the all $+1$ state and the all $-1$ state. The energy per spin of either ground state is $-2$. At high temperatures, the spins are independent, all states are equally probable, and the energy is expected to fluctuate around a mean of 0 with a standard deviation proportional to $1/\sqrt{N}$.

Let's look at some results. In all figures temperature $T$ is shown with $k_{\rm B} = 1$. The basic picture emerges with as few as 16 spins (figure 31.3, top): the energy rises monotonically. As we increase the number of spins to 100 (figure 31.3, bottom) some new details emerge. First, as expected, the fluctuations at large temperature decrease as $1/\sqrt{N}$. Second, the fluctuations at intermediate temperature become relatively *bigger*. This is the signature of a 'collective phenomenon', in this case, a phase transition. Only systems with infinite $N$ show true phase transitions, but with $N = 100$ we are getting a hint of the critical fluctuations. Figure 31.5 shows details of the graphs for $N = 100$ and $N = 4096$. Figure 31.2 shows a sequence of typical states from the simulation of $N = 4096$ spins at a sequence of decreasing temperatures.

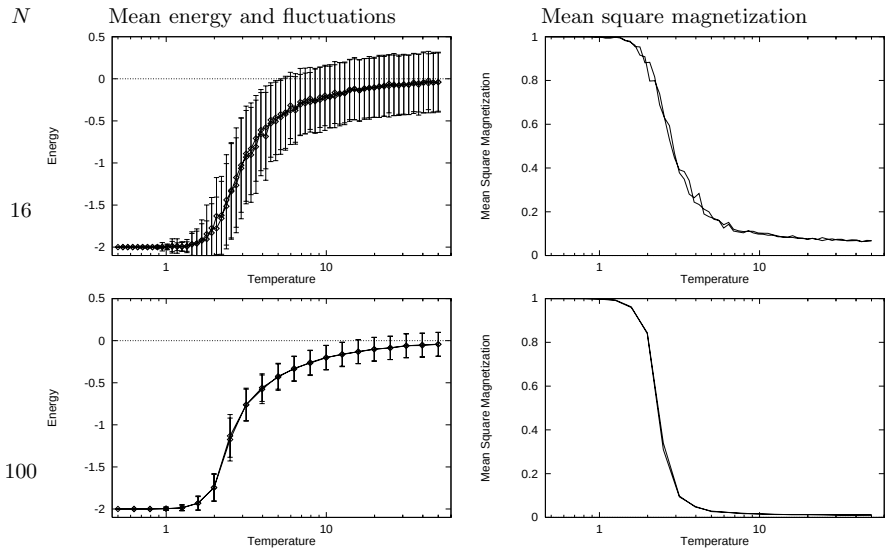



Figure 31.3. Monte Carlo simulations of rectangular Ising models with $J = 1$. Mean energy and fluctuations in energy as a function of temperature (left). Mean square magnetization as a function of temperature (right). In the top row, $N = 16$, and the bottom, $N = 100$. For even larger $N$, see later figures.

*Contrast with Schottky anomaly*

A peak in the heat capacity, as a function of temperature, occurs in any system that has a finite number of energy levels; a peak is not in itself evidence of a phase transition. Such peaks were viewed as anomalies in classical thermodynamics, since 'normal' systems with infinite numbers of energy levels (such as a particle in a box) have heat capacities that are either constant or increasing functions of temperature. In contrast, systems with a finite number of levels produced small blips in the heat capacity graph (figure 31.4).

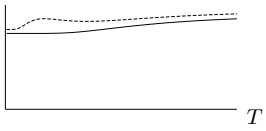



Figure 31.4. Schematic diagram to explain the meaning of a Schottky anomaly. The curve shows the heat capacity of two gases as a function of temperature. The lower curve shows a normal gas whose heat capacity is an increasing function of temperature. The upper curve has a small peak in the heat capacity, which is known as a Schottky anomaly (at least in Cambridge). The peak is produced by the gas having magnetic degrees of freedom with a finite number of accessible states.

Let us refresh our memory of the simplest such system, a two-level system with states $x = 0$ (energy 0) and $x = 1$ (energy $\epsilon$). The mean energy is

$$E(\beta) = \epsilon \frac{\exp(-\beta\epsilon)}{1 + \exp(-\beta\epsilon)} = \epsilon \frac{1}{1 + \exp(\beta\epsilon)} \tag{31.22}$$

and the derivative with respect to $\beta$ is

$$\mathrm{d}E/\mathrm{d}\beta = -\epsilon^2 \frac{\exp(\beta\epsilon)}{[1 + \exp(\beta\epsilon)]^2}. \tag{31.23}$$

So the heat capacity is

$$C = \mathrm{d}E/\mathrm{d}T = -\frac{\mathrm{d}E}{\mathrm{d}\beta} \frac{1}{k_{\mathrm{B}}T^2} = \frac{\epsilon^2}{k_{\mathrm{B}}T^2} \frac{\exp(\beta\epsilon)}{[1 + \exp(\beta\epsilon)]^2} \tag{31.24}$$

and the fluctuations in energy are given by $\mathrm{var}(E) = Ck_{\mathrm{B}}T^2 = -\mathrm{d}E/\mathrm{d}\beta$, which was evaluated in (31.23). The heat capacity and fluctuations are plotted in figure 31.6. The take-home message at this point is that whilst Schottky anomalies do have a peak in the heat capacity, there is *no* peak in their *fluctuations*; the variance of the energy simply increases monotonically with temperature to a value proportional to the number of independent spins. Thus it is a peak in the *fluctuations* that is interesting, rather than a peak in the heat capacity. The Ising model has such a peak in its fluctuations, as can be seen in the second row of figure 31.5.

*Rectangular Ising model with $J = -1$*

What do we expect to happen in the case $J = -1$? The ground states of an infinite system are the two checkerboard patterns (figure 31.7), and they have

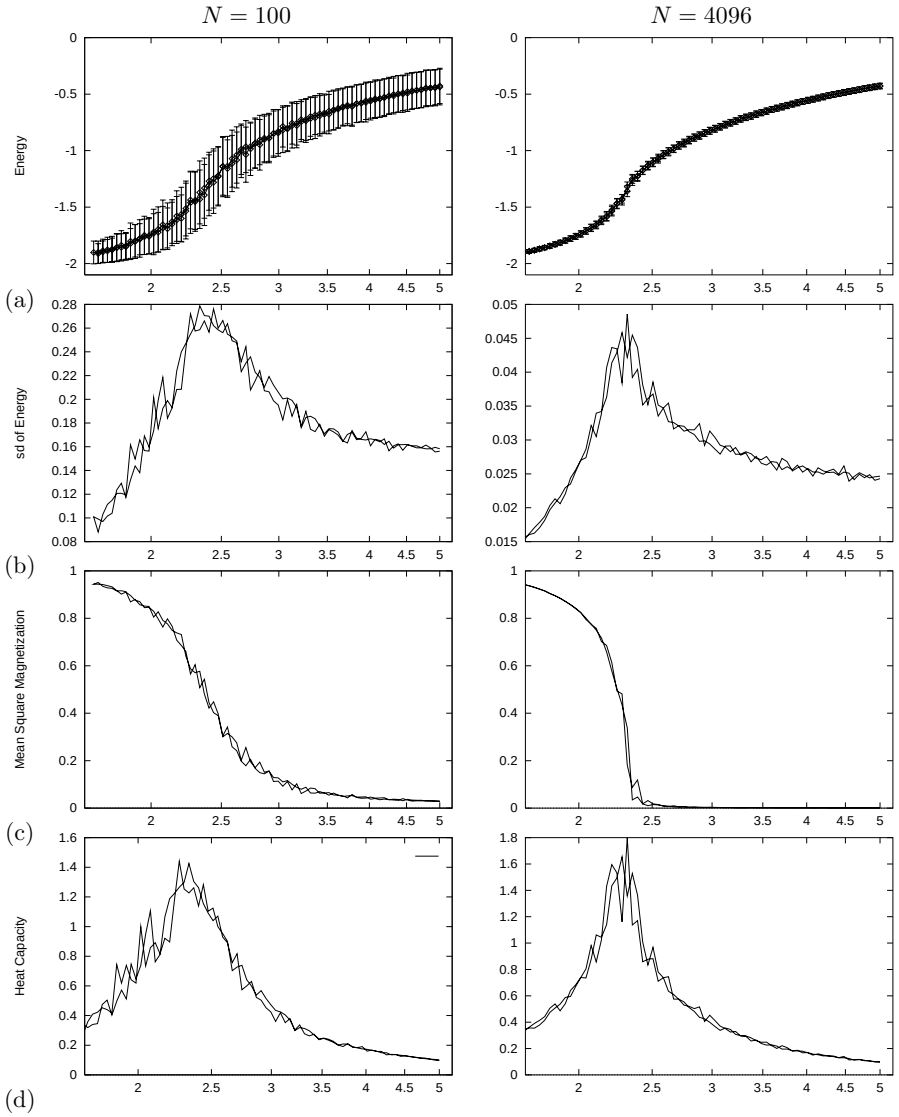



Figure 31.5. Detail of Monte Carlo simulations of rectangular Ising models with $J = 1$. (a) Mean energy and fluctuations in energy as a function of temperature. (b) Fluctuations in energy (standard deviation). (c) Mean square magnetization. (d) Heat capacity.

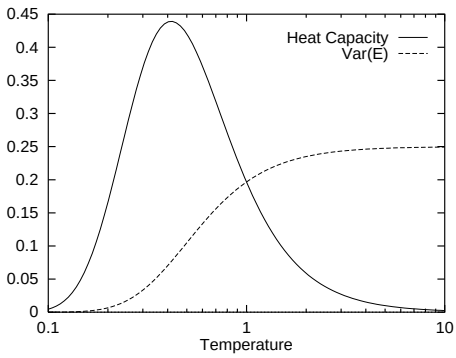



Figure 31.6. Schottky anomaly – Heat capacity and fluctuations in energy as a function of temperature for a two-level system with separation $\epsilon = 1$ and $k_{\rm B} = 1$.

energy per spin $-2$, like the ground states of the $J=1$ model. Can this analogy be pressed further?A moment's reflection will confirm that the two systems are equivalent to each other under a checkerboard symmetry operation. If you take an infinite $J = 1$ system in some state and flip all the spins that lie on the black squares of an infinite checkerboard, and set $J = -1$ (figure 31.8), then the energy is unchanged. (The magnetization changes, of course.) So all thermodynamic properties of the two systems are expected to be identical in the case of zero applied field.

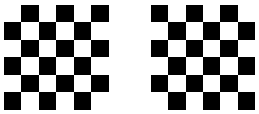


Figure 31.7. The two ground states of a rectangular Ising model with $J = -1$.

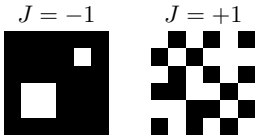



Figure 31.8. Two states of rectangular Ising models with $J = \pm 1$ that have identical energy.

But there is a subtlety lurking here. Have you spotted it? We are simulating finite grids with periodic boundary conditions. If the size of the grid in any direction is *odd*, then the checkerboard operation is no longer a symmetry operation relating $J = +1$ to $J = -1$, because the checkerboard doesn't match up at the boundaries. This means that for systems of odd size, the ground state of a system with $J = -1$ will have degeneracy greater than 2, and the energy of those ground states will not be as low as $-2$ per spin. So we expect qualitative differences between the cases $J = \pm 1$ in odd sized systems. These differences are expected to be most prominent for small systems. The frustrations are introduced by the boundaries, and the length of the boundary grows as the square root of the system size, so the fractional influence of this boundary-related frustration on the energy and entropy of the system will decrease as $1/\sqrt{N}$. Figure 31.9 compares the energies of the ferromagnetic and antiferromagnetic models with $N = 25$. Here, the difference is striking.

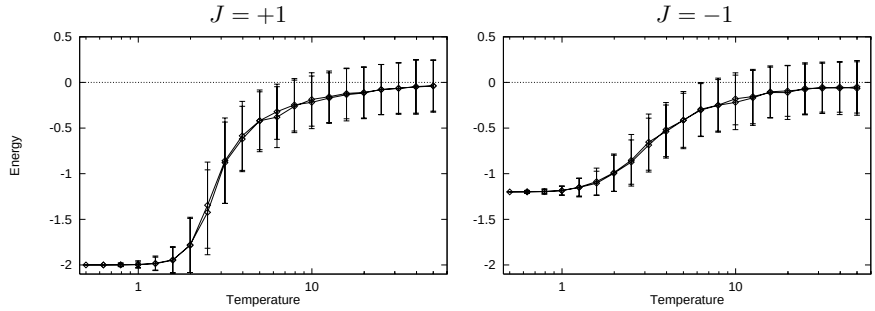



Figure 31.9. Monte Carlo simulations of rectangular Ising models with $J = \pm 1$ and $N = 25$. Mean energy and fluctuations in energy as a function of temperature. (a) $J = 1$. (b) $J = -1$.

### *Triangular Ising model*

We can repeat these computations for a triangular Ising model. Do we expect the triangular Ising model with $J = \pm 1$ to show different physical properties from the rectangular Ising model? Presumably the $J = 1$ model will have broadly similar properties to its rectangular counterpart. But the case $J = -1$ is radically different from what's gone before. Think about it: *there is no unfrustrated ground state*; in any state, there *must* be frustrations – pairs of neighbours who have the same sign as each other. Unlike the case of the rectangular model with odd size, the frustrations are not introduced by the periodic boundary conditions. *Every set of three mutually neighbouring spins must be in a state of frustration,* as shown in figure 31.10. (Solid lines show 'happy' couplings which contribute $-|J|$ to the energy; dashed lines show 'unhappy' couplings which contribute $|J|$.) Thus we certainly expect different behaviour at low temperatures. In fact we might expect this system to have a non-zero entropy at absolute zero. ('Triangular model violates third law of thermodynamics!')

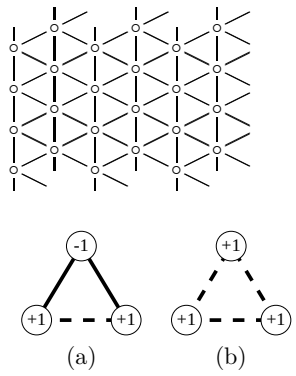



Figure 31.10. In an antiferromagnetic triangular Ising model, any three neighbouring spins are frustrated. Of the eight possible configurations of three spins, six have energy $-|J|$ (a), and two have energy $3|J|$ (b).

Let's look at some results. Sample states are shown in figure 31.12, and figure 31.11 shows the energy, fluctuations, and heat capacity for $N = 4096$.

Note how different the results for $J = \pm 1$ are. There is no peak at all in the standard deviation of the energy in the case $J = -1$. This indicates that the antiferromagnetic system does not have a phase transition to a state with long-range order.

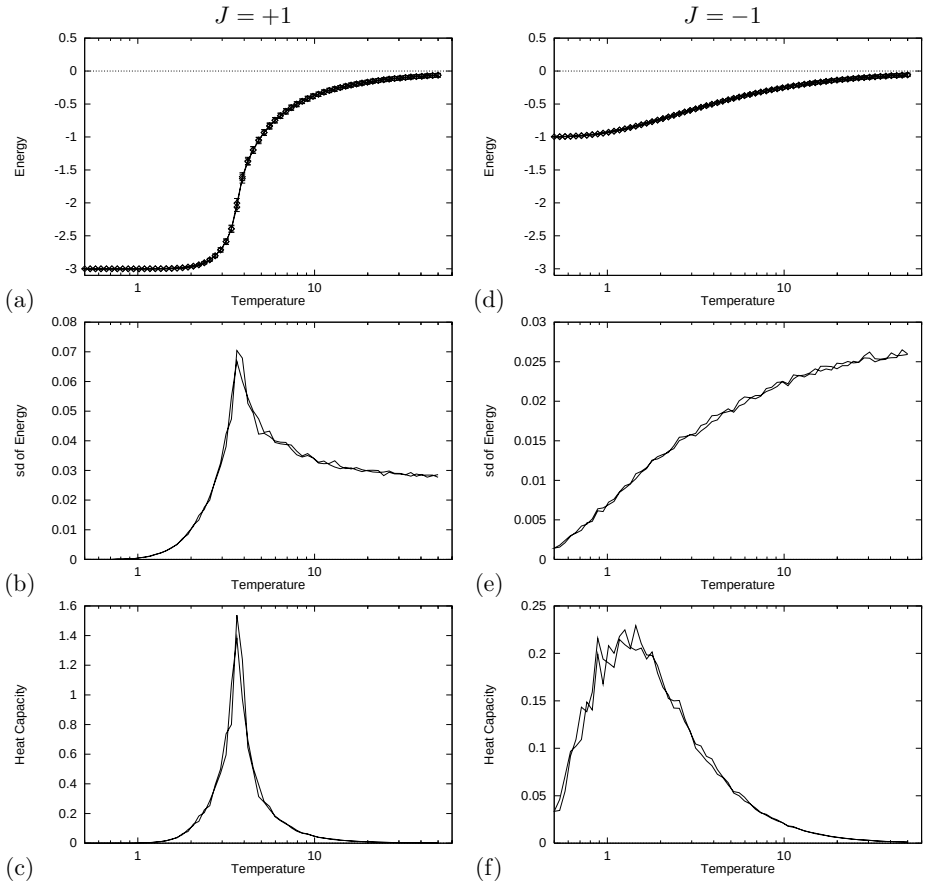



Figure 31.11. Monte Carlo simulations of triangular Ising models with $J = \pm 1$ and $N = 4096$. (a–c) $J = 1$. (d–f) $J = -1$. (a, d) Mean energy and fluctuations in energy as a function of temperature. (b, e) Fluctuations in energy (standard deviation). (c, f) Heat capacity.

## ▶ 31.2 Direct computation of partition function of Ising models

We now examine a completely different approach to Ising models. The *transfer matrix method* is an exact and abstract approach that obtains physical properties of the model from the partition function

$$Z(\beta, \mathbf{J}, \mathbf{b}) \equiv \sum_{\mathbf{x}} \exp[-\beta E(\mathbf{x}; \mathbf{J}, \mathbf{b})] \,, \tag{31.25}$$

where the summation is over all states $\mathbf{x}$, and the inverse temperature is $\beta = 1/T$. [As usual, Let $k_{\rm B} = 1$.] The free energy is given by $F = -\frac{1}{\beta} \ln Z$. The number of states is $2^N$, so direct computation of the partition function is not possible for large $N$. To avoid enumerating all global states explicitly, we can use a trick similar to the sum–product algorithm discussed in Chapter 25. We concentrate on models that have the form of a long thin strip of width $W$ with periodic boundary conditions in both directions, and we iterate along the length of our model, working out a set of *partial partition functions* at one location $l$ in terms of partial partition functions at the previous location $l - 1$. Each iteration involves a summation over all the states at the boundary. This operation is exponential in the width of the strip, $W$. The final clever trick

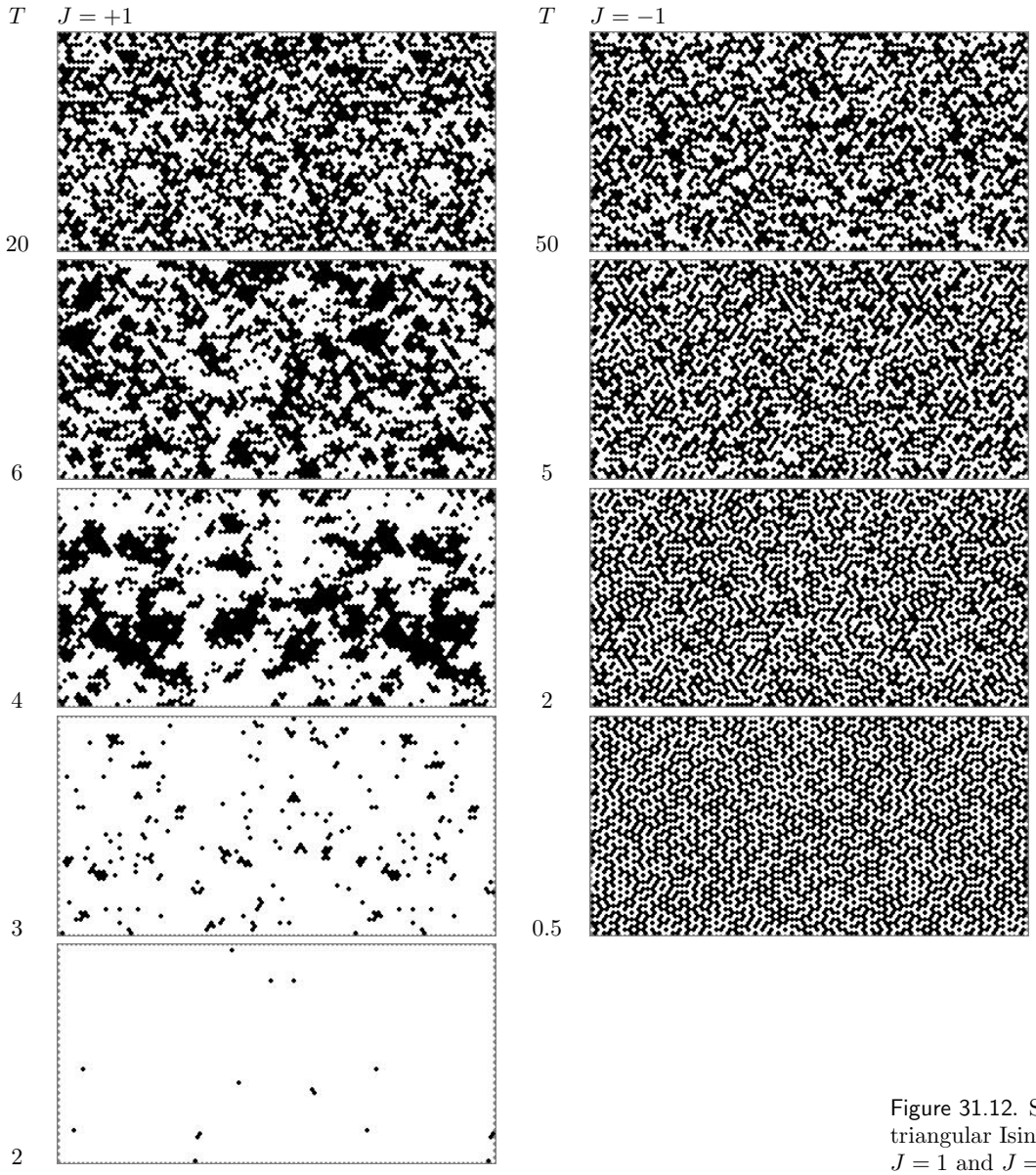



Figure 31.12. Sample states of triangular Ising models with $J = 1$ and $J = -1$.

is to note that if the system is translation-invariant along its length then we only need to do *one* iteration in order to find the properties of a system of *any* length.

The computational task becomes the evaluation of an $S \times S$ matrix, where $S$ is the number of microstates that need to be considered at the boundary, and the computation of its eigenvalues. The eigenvalue of largest magnitude gives the partition function for an infinite-length thin strip.

Here is a more detailed explanation. Label the states of the $C$ columns of the thin strip $s_1, s_2, \ldots, s_C$, with each $s$ an integer from 0 to $2^W - 1$. The $r$th bit of $s_c$ indicates whether the spin in row $r$, column $c$ is up or down. The partition function is

$$Z = \sum_{\mathbf{x}} \exp(-\beta E(\mathbf{x})) \tag{31.26}$$

$$= \sum_{s_1} \sum_{s_2} \cdots \sum_{s_C} \exp\left( -\beta \sum_{c=1}^{C} \mathcal{E}(s_c, s_{c+1}) \right) \tag{31.27}$$

where $\mathcal{E}(s_c, s_{c+1})$ is an appropriately defined energy, and, if we want periodic boundary conditions, $s_{C+1}$ is defined to be $s_1$. One definition for $\mathcal{E}$ is:

$$\mathcal{E}(s_c, s_{c+1}) = \sum_{\substack{(m,n)\in\mathcal{N}: \\ m\in c, n\in c+1}} J\, x_m x_n + \frac{1}{4} \sum_{\substack{(m,n)\in\mathcal{N}: \\ m\in c, n\in c}} J\, x_m x_n + \frac{1}{4} \sum_{\substack{(m,n)\in\mathcal{N}: \\ m\in c+1, n\in c+1}} J\, x_m x_n. \tag{31.28}$$

This definition of the energy has the nice property that (for the rectangular Ising model) it defines a matrix that is symmetric in its two indices $s_c, s_{c+1}$. The factors of 1/4 are needed because vertical links are counted four times. Let us define

$$M_{ss'} = \exp\left(-\beta \mathcal{E}(s, s')\right). \tag{31.29}$$

Then continuing from equation (31.27),

$$Z = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_C} \left[ \prod_{c=1}^{C} M_{s_c, s_{c+1}} \right] \tag{31.30}$$

$$= \text{Trace}\left[\mathbf{M}^C\right] \tag{31.31}$$

$$= \sum_a \mu_a^C, \tag{31.32}$$

where $\{\mu_a\}_{a=1}^{2^W}$ are the eigenvalues of $\mathbf{M}$. As the length of the strip $C$ increases, $Z$ becomes dominated by the largest eigenvalue $\mu_{\max}$:

$$Z \to \mu_{\max}^C. \tag{31.33}$$

So the free energy per spin in the limit of an infinite thin strip is given by:

$$f = -kT \ln Z/(WC) = -kTC \ln \mu_{\max}/(WC) = -kT \ln \mu_{\max}/W. \tag{31.34}$$

It's really neat that *all* the thermodynamic properties of a long thin strip can be obtained from just the largest eigenvalue of this matrix $\mathbf{M}$!

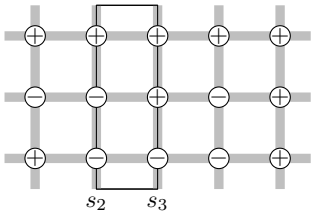



Figure 31.13. Illustration to help explain the definition (31.28). $\mathcal{E}(s_2, s_3)$ counts all the contributions to the energy in the rectangle. The total energy is given by stepping the rectangle along. Each horizontal bond inside the rectangle is counted once; each vertical bond is half-inside the rectangle (and will be half-inside an adjacent rectangle) so half its energy is included in $\mathcal{E}(s_2, s_3)$; the factor of 1/4 appears in the second term because $m$ and $n$ both run over all nodes in column $c$, so each bond is visited twice.

For the state shown here, $s_2 = (100)_2$, $s_3 = (110)_2$, the horizontal bonds contribute $+J$ to $\mathcal{E}(s_2, s_3)$, and the vertical bonds contribute $-J/2$ on the left and $-J/2$ on the right, assuming periodic boundary conditions between top and bottom. So $\mathcal{E}(s_2, s_3) = 0$.

*Computations*

I computed the partition functions of *long thin strip* Ising models with the geometries shown in figure 31.14.

As in the last section, I set the applied field $H$ to zero and considered the two cases $J = \pm 1$ which are a ferromagnet and antiferromagnet respectively. I computed the free energy per spin, $f(\beta, J, H) = F/N$ for widths from $W = 2$ to 8 as a function of $\beta$ for $H = 0$.

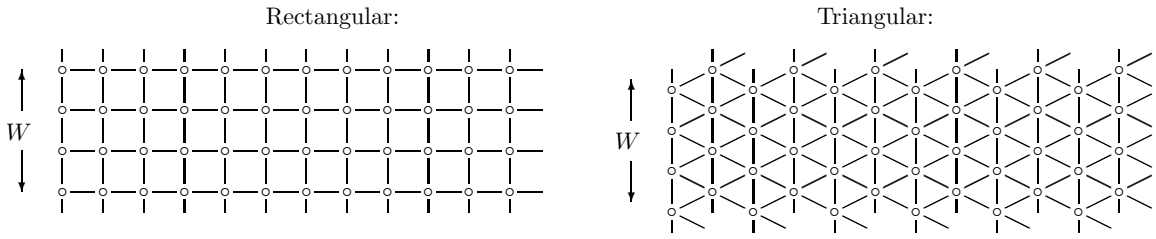



Figure 31.14. Two long thin strip Ising models. A line between two spins indicates that they are neighbours. The strips have width $W$ and infinite length.

*Computational ideas:*

Only the largest eigenvalue is needed. There are several ways of getting this quantity, for example, iterative multiplication of the matrix by an initial vector. Because the matrix is all positive we know that the principal eigenvector is all positive too (Frobenius–Perron theorem), so a reasonable initial vector is $(1, 1, \ldots, 1)$. This iterative procedure may be faster than explicit computation of all eigenvalues. I computed them all anyway, which has the advantage that we can find the free energy of finite length strips – using equation (31.32) – as well as infinite ones.

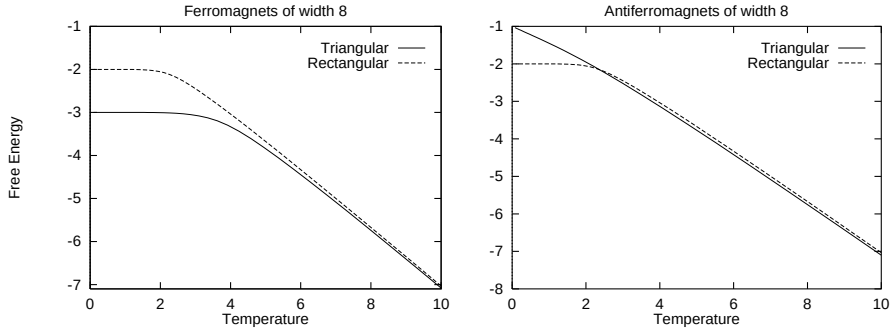



Figure 31.15. Free energy per spin of long-thin-strip Ising models. Note the non-zero gradient at $T = 0$ in the case of the triangular antiferromagnet.

*Comments on graphs:*

For large temperatures all Ising models should show the same behaviour: the free energy is entropy-dominated, and the entropy per spin is $\ln(2)$. The mean energy per spin goes to zero. The free energy per spin should tend to $-\ln(2)/\beta$. The free energies are shown in figure 31.15.

One of the interesting properties we can obtain from the free energy is the degeneracy of the ground state. As the temperature goes to zero, the Boltzmann distribution becomes concentrated in the ground state. If the ground state is degenerate (i.e., there are multiple ground states with identical

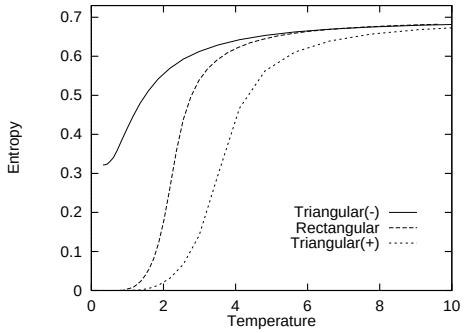



Figure 31.16. Entropies (in nats) of width 8 Ising systems as a function of temperature, obtained by differentiating the free energy curves in figure 31.15. The rectangular ferromagnet and antiferromagnet have identical thermal properties. For the triangular systems, the upper curve $(-)$ denotes the antiferromagnet and the lower curve $(+)$ the ferromagnet.

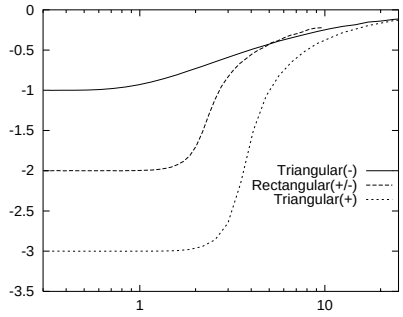



Figure 31.17. Mean energy versus temperature of long thin strip Ising models with width 8. Compare with figure 31.3.

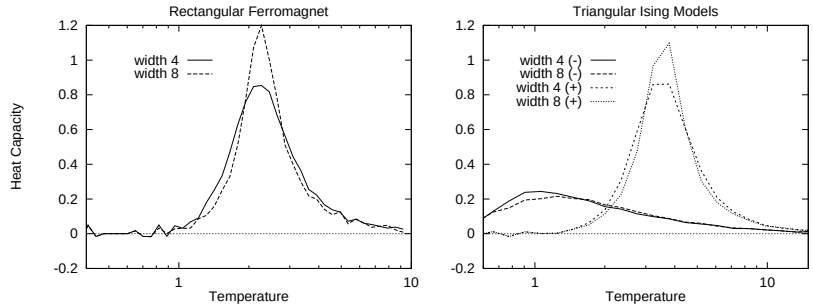



Figure 31.18. Heat capacities of (a) rectangular model; (b) triangular models with different widths, (+) and (−) denoting ferromagnet and antiferromagnet. Compare with figure 31.11.

energy) then the entropy as $T \to 0$ is non-zero. We can find the entropy from the free energy using $S = -\partial F/\partial T$.

The entropy of the triangular antiferromagnet at absolute zero appears to be about 0.3, that is, about half its high temperature value (figure 31.16). The mean energy as a function of temperature is plotted in figure 31.17. It is evaluated using the identity $\langle E \rangle = -\partial \ln Z/\partial \beta$.

Figure 31.18 shows the estimated heat capacity (taking raw derivatives of the mean energy) as a function of temperature for the triangular models with widths 4 and 8. Figure 31.19 shows the fluctuations in energy as a function of temperature. All of these figures should show smooth graphs; the roughness of the curves is due to inaccurate numerics. The nature of any phase transition is not obvious, but the graphs seem compatible with the assertion that the ferromagnet shows, and the antiferromagnet does not show a phase transition.

The pictures of the free energy in figure 31.15 give some insight into how we could predict the transition temperature. We can see how the two phases of the ferromagnetic systems each have simple free energies: a straight sloping line through $F = 0$, $T = 0$ for the high temperature phase, and a horizontal line for the low temperature phase. (The slope of each line shows what the entropy per spin of that phase is.) The phase transition occurs roughly at the intersection of these lines. So we predict the transition temperature to be linearly related to the ground state energy.

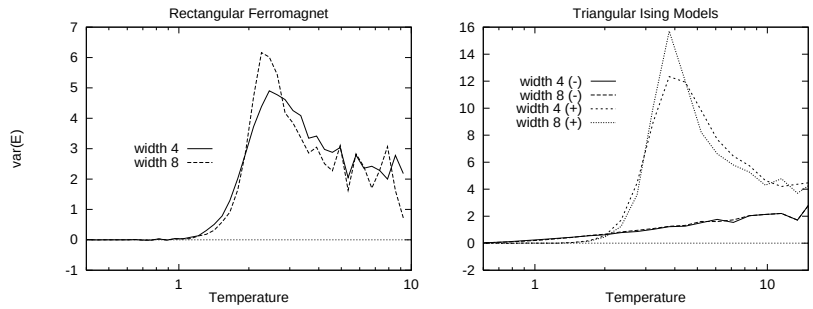



Figure 31.19. Energy variances, per spin, of (a) rectangular model; (b) triangular models with different widths, (+) and (−) denoting ferromagnet and antiferromagnet. Compare with figure 31.11.

*Comparison with the Monte Carlo results*

The agreement between the results of the two experiments seems very good. The two systems simulated (the long thin strip and the periodic square) are not quite identical. One could a more accurate comparison by finding all eigenvalues for the strip of width $W$ and computing $\sum \lambda^W$ to get the partition function of a $W \times W$ patch.

## 31.3 Exercises

▷ Exercise 31.2.[4] What would be the best way to extract the entropy from the Monte Carlo simulations? What would be the best way to obtain the entropy and the heat capacity from the partition function computation?

Exercise 31.3.[3] An Ising model may be generalized to have a coupling $J_{mn}$ between any spins $m$ and $n$, and the value of $J_{mn}$ could be different for each $m$ and $n$. In the special case where all the couplings are positive we know that the system has two ground states, the all-up and all-down states. For a more general setting of $J_{mn}$ it is conceivable that there could be *many* ground states.

Imagine that it is required to make a spin system whose local minima are a given list of states $\mathbf{x}_{(1)}, \mathbf{x}_{(2)}, \ldots, \mathbf{x}_{(S)}$. Can you think of a way of setting $\mathbf{J}$ such that the chosen states are low energy states? You are allowed to adjust all the $\{J_{mn}\}$ to whatever values you wish.