Chapter 8

Advanced Monte Carlo Methods

The goal of Monte Carlo simulations is usually to represent the equilibrium ensemble, without requiring that the succession of simulation microstates bears any resemblance to the actual dynamics of the system. The resulting freedom to invent Monte Carlo moves (consistent with detailed balance, of course) has resulted in many ingenious techniques for more rapid sampling of equilibrium ensembles.

Lattice models in particular are simpler than more microscopically accurate representations of molecules. As a consequence, more sophisticated Monte Carlo techniques can be devised for lattice models to cope with various circumstances in which straightforward Monte Carlo simulation is frustratingly slow to equilibrate, or to explore configuration space.

One such circumstance is simulation near the critical point. Fluctuations in a system in the one-phase region near the critical point become large in spatial extent and relax very slowly, because the driving force for fluctuations to relax depends on their free energy cost, which becomes small near the critical point. This tendency for long-wavelength fluctuations to relax slowly as we approach the critical point is called “critical slowing down”; we shall discuss it more systematically in the next chapter. For now, it suffices to observe that critical slowing down is problematic for conventional Monte Carlo simulations, in which we flip spins one at a time. For such problems, methods have been devised in which entire clusters of spins can be flipped in a single move (Section 8.1).

Phase-separated systems, in which the configuration is essentially described by the shapes of interfaces between the coexisting phases, is another circumstance in which straightforward lattice Monte Carlo is very slow to explore equilibrium. Often, for such systems we would prefer to conserve the number of A and B species, for example in simulating the properties of a single interface between phases. Simple Monte Carlo moves attempting to exchange two randomly chosen sites will typically fail to be accepted, unless they have the effect of moving the interface a bit. Again for lattice models, special techniques have been developed to keep track of which exchange moves are most likely to succeed, and attempt these more often (Section 8.2.1).

For any equilibrium simulation technique, whether molecular dynamics or Monte Carlo, we often want to explore how the properties of the system vary with temperature, field, or some other parameter in the Hamiltonian. For example, in searching for the spinodal or phase boundary in a simulation, we evidently must vary the temperature in some systematic way. At first sight, it would seem that each time we want to change the temperature even a little bit, we must perform a new simulation, since the microstates in the partition function are weighted differently at each temperature. Fortunately, there is a technique for “reweighting” histograms of values for observables (energy, pressure, composition, ...) from simulations performed at one temperature, to obtain a histogram as if we had simulated at a nearby temperature (Section 8.3).
Histogram reweighting does not work if the simulation to be reweighted does not visit configurations typically occurring at the target conditions (different temperature, e.g.). More generally, we often want to use simulations to explore the probability of unlikely configurations, such as the probability that a pair of colloidal particles with a solvent-induced attraction are found at increasingly large separations from each other. If the particles are observed together in all of the simulation snapshots, we get no quantitative information about how likely the particles are to be unbound. In such cases, we can “encourage” the particles to move apart by adding a fictitious extra term to the Hamiltonian, which favors greater separations $|r|$ between the particles — for example, a potential of the form $(|r| - r_0)^2$. We can then “reweight” to undo the effects of the extra potential (Section 8.5).

Finally, there are many circumstances in which equilibration is slow because of what may be generally described as “constraints” operating in the system — too low a temperature for moves to succeed, or too high a hard-core potential barrier for particles to slip past each other, for example. It would be helpful in such cases to have a way of “temporarily relaxing” the tight restrictions on the molecular dynamics motion or success of Monte Carlo moves in such systems, so that the microstate could “slip past” a difficult bottleneck into a new configuration. For this purpose, a technique called “parallel tempering” was developed, in which simulations at different values of some parameter (temperature, e.g.) are performed in parallel, and sometimes allowed to swap configurations. Thus a given configuration is allowed sometimes to “visit” higher temperatures, where it can rearrange more freely, before returning to lower temperatures, where it “settles down” to a representative configuration (Section 8.6).

### 8.1 Cluster flipping

[Newman & Barkema, p. 92ff]

Note that near the critical point, we expect the system to contain many large connected clusters of aligned spins; these are the natural long-wavelength fluctuations of the system, which are slow to relax. (See Fig. 8.1.) This is evident in the coarse-grained model of the previous chapter; a small sinusoidal concentration fluctuation $\psi_k$ has a free energy cost $\beta F$ equal to $(1/2)(\tau + a^2 k^2)|\psi_k|^2$; for fluctuations of long wavelength (small $k$) near the critical point (at which $\tau$ vanishes), the cost of such fluctuations is indeed small. We may regard $\tau + a^2 k^2$ as the “spring constant” governing the energy of the harmonic fluctuation $\psi_k$. Intuitively, we expect that the smaller the cost of a fluctuation, the weaker the tendency for the fluctuation to relax, and the slower the relaxation rate will be.

It would be nice if we could design Monte Carlo moves that flipped an entire cluster of aligned spins at once, rather than nibbling away at such clusters one spin at a time. Such a scheme would consist of two elements: 1) the selection of a cluster to flip, and 2) a criterion for accepting the flip of the cluster. These two elements taken together must be designed to satisfy detailed balance. Naturally, we would prefer an economical method for building the clusters, with a high acceptance rate.

The Wolff algorithm, based on earlier work of Swendsen and Wang, satisfies all these requirements. Here we present the algorithm, and in the process show that detailed balance is satisfied.

The cluster to be flipped is “grown” by a stochastic process from a randomly selected “seed” site, which is the first site added to the cluster. When a site is added to the cluster, we consider adding to the cluster the nearest neighbors of that site as well. Only sites aligned with the seed site are eligible to be added (the cluster we grow will be a connected cluster of aligned spins). If a nearest neighbor of a site just added to the cluster is aligned, we add it to the cluster with some
8.1. CLUSTER FLIPPING

Figure 8.1: Typical Ising model configurations in the one-phase region with zero field on approach to the critical point ($\beta \approx 0.44$), in which the growth in the size of correlated regions is evident.
constant probability \( p \) (we shall determine \( p \) below). We continue adding sites to the cluster and considering the neighbors of sites just added, until all the neighbors so considered fail to join the cluster.

This recursive algorithm is implemented with the use of a “queue” — a list of sites in line to be added to the cluster, and to have their nearest neighbors considered:

1. Choose a seed site at random; add the seed site to the “queue” (a list of sites to be added to the cluster), and “mark” the site as added.

2. While the queue is not empty,
   
   (a) take the first site off the queue;
   (b) examine the unmarked neighbors of this site; if they are aligned with the seed, add them to the queue with probability \( p \).

The cluster then consists of the marked sites.

Fig. 8.2 shows an example of the building of a cluster by the above recipe. The gray sites are currently spin up, surrounded by a region of spin down (white). In the first step, a “seed” site is chosen (black dot); the surrounding sites (question marks) may or may not be added. In the next step, the seed site is part of the growing cluster (now white circle), and we have added and marked some of the neighboring sites (now black dots). In turn, the neighbors of these newly marked sites may be added (question marks). A site may be attempted to add to the cluster more than once, if it has multiple neighbors in the growing cluster. In the final frame, the perimeter bonds are shown as white bars if “good” bonds (i.e., between like spins when the cluster is flipped), and black bars if “bad” bonds (between unlike spins when the cluster is flipped).

![Figure 8.2: Example of cluster growth, following Wolff algorithm.](image)
8.1. CLUSTER FLIPPING

To prove detailed balance, we consider a particular cluster, grown starting from a particular seed site, and grown by adding neighboring sites in a particular order. For every such cluster flip, there is a matching reverse move that flips the cluster back again, in which we choose the same seed site, and add the same neighboring sites in the same order.

The probabilities for generating the forward and reverse moves differ only with respect to “broken” bonds. When we flip the cluster (forward move), suppose that on the boundary of the cluster there are $M$ spins that are initially aligned with the cluster and could have been added, but were not added to the cluster. The $M$ bonds with these spins will be “broken” by the flip. Not adding these $M$ spins gives rise to a factor $(1 - p)^M$ in the probability of constructing this cluster.

Likewise, suppose $N$ bonds are “broken” by the reverse flip of the same cluster. In the case of the Ising model, we would have $M + N$ total spins on the boundary of the cluster, $M$ aligned and $N$ antialigned with the original cluster. The $N$ spins not added to the cluster to be flipped in the reverse direction gives rise to a factor $(1 - p)^N$ in the probability of constructing this cluster.

Detailed balance requires that the forward and backward rates $R(m \rightarrow n)$ and $R(n \rightarrow m)$ satisfy

$$e^{-\beta E_n} R(m \rightarrow n) = e^{-\beta E_n} R(n \rightarrow m)$$

which ensures that the canonical distribution is stationary under the simulation.

We can write the rate $R(m \rightarrow n)$ as the product of the generation rate $G(m \rightarrow n)$ (probability that a given cluster move will be proposed) times the acceptance probability $A(m \rightarrow n)$ (probability that a given proposed move will be accepted). Using detailed balance, we have

$$\frac{G(m \rightarrow n)A(m \rightarrow n)}{G(n \rightarrow m)A(n \rightarrow m)} = \frac{(1 - p)^M A(m \rightarrow n)}{(1 - p)^N A(n \rightarrow m)} = e^{-\beta(E_n-E_m)}$$

Now the energy difference $E_n - E_m$ on flipping the cluster equals $2J(M - N)$, arising entirely from the changes in the bonds on the cluster boundary; the forward flip breaks $M$ bonds, changing the energy by $2JM$, and “makes” $N$ bonds, which changes the energy by $-2JN$.

Regrouping the above equation as

$$\frac{A(m \rightarrow n)}{A(n \rightarrow m)} = e^{-2\beta J(M-N)} = \left(\frac{1}{1-p}\right)^{M-N}$$

If we choose $(1 - p)$ equal to $e^{-2\beta J}$, or equivalently

$$p = 1 - e^{-2\beta J}$$

we can make the right-hand side equal to unity, whereupon we can take both the forward and backwards acceptance rates equal to unity.

That is, we can flip every cluster we build. Note finally that the Wolff algorithm is evidently ergodic, since with finite probability we flip a single spin, failing to add its neighbors to the cluster whether or not they are aligned.

The size of the clusters generated by the Wolff algorithm varies with temperature. At high temperatures ($\beta J << 1$), $p$ is nearly zero; we almost never add a site to the cluster beyond the original seed, and we flip single spins most of the time. As we approach the critical point, it turns out that the cluster sizes grow until they become as large as the system.

At low temperatures, where the system is typically aligned except for a few isolated antialigned spins, $p$ is nearly unity, and all cluster flips are as large as the system, omitting only a few scattered “mothholes” of sites that failed to add to the cluster. The probability of omitting an isolated site from the cluster is $(1 - p)^4$ (since the site must fail four times to be added, when invited by each
Figure 8.3: Examples of cluster flips, for $\beta$ values slightly above $T_c$ ($\beta = 0.42$) and slightly below $T_c$ ($\beta = 0.46$). At left, green sites are former red sites flipped to blue; at right, orange sites are former blue sites flipped to red.
8.1. CLUSTER FLIPPING

of its neighbors). Note that \((1 - p)^4\) equals \(e^{-8\beta J}\), which is the Boltzmann factor for flipping an isolated spin (which entails breaking four bonds, at a cost of \(2J\) each). Thus at low temperatures a single cluster flip leads to a new collection of isolated antialigned spins, renewing the configuration.

In the presence of a field (“magnetic” field, for the interpretation of the Ising model as a magnetic system),

\[ \delta H = -h \sum_i \sigma_i = -hM \tag{8.5} \]

the Wolff algorithm is modified, not with respect to the cluster growing, but with respect to the acceptance criterion. (Here \(M\) is the “magnetization” of the system.)

Instead of accepting every cluster flip proposed, we can satisfy detailed balance in the field by accepting cluster flips with the Metropolis weighting for the field energy \(-hM\) only,

\[ A(m \to n) = \begin{cases} 1 & -h\Delta M < 0 \\ e^{\beta h M} & h\Delta M > 0 \end{cases} \tag{8.6} \]

in which \(\Delta M = M_n - M_m\) is the change in magnetization. It can be shown that this prescription satisfies detailed balance, which we leave as an exercise for the reader.

The Wolff algorithm can be adapted to lattice models other than the Ising model, as long as a set of generalized “spin flip” moves exist that satisfy the following requirements:

1. The set of flips applied to a single spin are sufficient to visit all possible states of the spin (this guarantees the Wolff algorithm to be ergodic).

2. A flip applied twice restores the original state (this is part of what we mean by a “flip”, and necessary to ensure that the forward and reverse cluster flips are “mirror images” of each other).

3. The Hamiltonian is invariant under a global flip of every spin (this ensures that the energy of spins interior to a large cluster are the same before and after a cluster flip).

Examples of lattice models to which the Wolff algorithm can be adapted include the Potts model, as well as models with “vector” spins at each lattice site such as the Heisenberg model (in which each site holds a unit vector \(\hat{n}_i\), and the near-neighbor interaction is \(J\hat{n}_i \cdot \hat{n}_j\)).

For the \(q\)-state Potts model, in which spins take on values 0, 1, \ldots, \(q-1\), a spin flip operation \(ab\) “flips” a spin to \(b\) if its value was \(a\), and vice versa. Because the Potts Hamiltonian has the same cost \(J\) for any near-neighbor bond between unlike spins, the Hamiltonian is invariant under global spin flips of this kind. Evidently, the same flip applied twice returns us to the original configuration.

The rule for building the cluster remains the same: only spins aligned with the seed are invited to join, and are added with probability \(p = 1 - e^{-2\beta J}\). The cluster boundary now consists of three kinds of sites: \(M\) sites aligned with the original cluster, \(N\) sites aligned with the flipped cluster, and some other sites aligned with neither the original nor the flipped cluster. The detailed balance proof goes through as before.

For the Heisenberg model, the spin flip operation is to reflect a spin through a plane normal to some vector \(v\). The Heisenberg Hamiltonian is invariant to such an operation globally applied, and a configuration is restored to its original state when such a flip is applied twice. The rule for building the cluster is amended; since spins point any which way, there is no finite subset of spins that are evidently “aligned” with the seed.

The rule for inviting spins to be added is: 1) if the flip would improve the bond with the neighbor issuing the invitation to flip, the spin is never added; 2) otherwise, the spin \(j\) is flipped.
and added to the cluster via its neighbor $i$ with probability $p$ given by

$$p = 1 - e^{-\beta \Delta H_{\text{fail}}} = 1 - e^{\beta J R(s_i) \cdot (s_j - R(s_j))}$$

(8.7)

Here $R(s_j)$ denotes the direction of $s_j$ after being flipped. The energy $\Delta H_{\text{fail}}$ is the energy cost of failing to flip spin $j$, i.e., the difference in energy of a bond between $R(s_i)$ and $s_j$, and that of a bond between $R(s_i)$ and $R(s_j)$ (which is the same as that of a bond between $s_i$ and $s_j$).
8.2 Conserved order parameter

[Newman & Barkema p. 138ff, p. 271ff]

For certain problems, the relevant ensemble is one in which the number of up and down spins is conserved. For example, we may wish to simulate the concentration profile across an interface between coexisting phases in the Ising model well below the critical point, in the two-phase region. In a simulation with cluster flips, the entire system would change phase, and we could not hold an interface in place to study its profile.

With a simulation in which pairs of spins are exchanged, we can contrive to hold an approximately planar interface somewhere in the middle of the system by fixing spins on the upper and lower boundaries to be up and down respectively, and starting the system with equal numbers of up and down spins (most conveniently, in a configuration with a perfectly smooth interface in the middle of the system).

If we are only interested in the equilibrium properties of such a system, we can exchange distant pairs of spins. If we choose two spins at random, we will waste a lot of time choosing pairs of aligned spins to exchange. In a system with a fraction $f$ of up spins, only a fraction $2f(1 - f)$ of randomly chosen pairs are antialigned. We can save some time by keeping lists of the up spins and the down spins, and at each step selecting at random a spin from each list to exchange.

![Figure 8.4: Nonlocal spin exchange A, with energy $\Delta E = 16J$, will likely fail in a well-separated Ising system with $\beta J > 1$. Spin exchange B, with $\Delta E = -8J$, is downhill and will succeed. Exchange C, with $\Delta E = 4J$, is uphill but may succeed anyway.](image)

However, for a system in which the two components are well separated (well into the two phase region), we will still waste a lot of time selecting pairs of spins that are inside the pure component region, far from the interface, and therefore very unlikely to succeed in exchanging. In such a case, it would be nice if we could choose pairs of spins that would be more likely to succeed in exchanging with greater probability than pairs that would most often fail to exchange. In the next section, we describe a scheme to accomplish this.

8.2.1 Continuous-time MC

To be able to choose pairs of spins to exchange that will be more likely to be accepted, we must keep track of some additional information about the local environment of each spin. We can certainly keep track of the number \( a_i \) of neighbors of each spin \( i \) that are aligned with the spin (\( a_i \) is called the “spin coordination number”).

We note that when we exchange two spins \( i \) and \( j \), which takes us from the starting microstate \( m \) to another microstate \( n \), we can write the change in the energy in terms of the spin coordination numbers \( a_i,m \) and \( a_i,n \) before and after the spin exchange.

When we change the spin at \( i \), the local energy change is equal to \( -2J \) times the increase in the number of aligned neighboring spins. Adding together the changes at sites \( i \) and \( j \), we have

\[
\Delta E_{mn} = -2J(a_{i,n} - a_{i,m}) - 2J(a_{j,n} - a_{j,m}) \tag{8.8}
\]

Detailed balance as usual requires that the ratio of forward to backward rates for the spin exchange be given by

\[
\frac{R(m \to n)}{R(n \to m)} = e^{-\beta \Delta E_{mn}} = e^{-2\beta J(a_{i,m} + a_{j,m})} \tag{8.9}
\]

The final form above suggests a strategy for selecting pairs of spins to exchange; namely, select each spin \( i \) to be flipped with a probability proportional to \( e^{-2\beta J a_{i,m}} \). This is manageable because the selection probability for the two spins are independent of each other, and of the properties of the final state.

We thus consider generating a pair of spins \( i \) and \( j \) to exchange, and hence a move from microstate \( m \) to \( n \), with probability

\[
G(m \to n) = \frac{e^{-2\beta J(a_{i,m} + a_{j,m})}}{F_m} \tag{8.10}
\]

with a normalization factor for the pair selection probability given by

\[
F_m = \sum_{i \text{ up}, j \text{ down}} e^{-2\beta J(a_{i,m} + a_{j,m})} \tag{8.11}
\]

in which the sum runs over pairs of antialigned spins, with \( i \) an up spin and \( j \) a down spin. The factor \( F_m \) is not constant, but depends on the microstate \( m \).

We can write the ratio of forward and backward rates once again as

\[
\frac{R(m \to n)}{R(n \to m)} = \frac{G(m \to n) A(m \to n)}{G(n \to m) A(n \to m)} = \frac{A(m \to n)}{A(n \to m)} \frac{F_n}{F_m} e^{-\beta \Delta E_{mn}} \tag{8.12}
\]

We need to adjust the acceptance rate \( A(m \to n) \) for the randomly selected exchange to satisfy detailed balance. We can accomplish this by making the acceptance rate \( A(m \to n) \) proportional to \( F_m \). But because \( F_m \) does not depend on the target microstate \( n \), we need not reject any of the proposed spin exchanges; rather, we can account for the microstate dependence of acceptance rates by incrementing the simulation “time clock” by \( \Delta t_m \) proportional to \( F_m^{-1} \). Then, accepting every proposed spin exchange results in an acceptance rate proportional to \( F_m \).
8.2. CONSERVED ORDER PARAMETER

In this continuous-time Monte Carlo scheme, the ensemble average of a quantity $Q$ from a set of values $\{Q_i\}$ is computed taking account of the varying time interval $\Delta t_i$ during which the value of $Q$ was equal to $Q_i$, according to

$$
\langle Q \rangle = \frac{\sum_i Q_i \Delta t_i}{\sum_i \Delta t_i}
$$

(8.13)

To implement this scheme for the Ising model, we must keep separate lists of up spins and down spins in the current microstate with each possible value of spin coordination number (ten lists altogether for the Ising model on a $d = 2$ dimensional square lattice).

To select an up spin at random with probability proportional to $e^{-2\beta J a_i}$, we proceed as follows. Suppose there are $N_a$ up spins with spin coordination number $a$. We first select from among the possible values of $a$ with probability proportional to $N_a e^{-2\beta J a}$; then, we choose a spin at random from the selected list.

(To choose $a$ from among $N$ index values with probability $f_k$, with $k = 1, 2, \ldots N$ and $\sum_k f_k = 1$, form the cumulative probability partial sums $c_k = \sum_{j=1}^{k} f_k$, which gives the probability that the index value is less than or equal to $k$. Then generate a random number $r$ between zero and unity; if $r$ falls between $c_k$ and $c_{k+1}$, select the $k$th index value.)

Since continuous-time Monte Carlo avoids entirely the problem of moves being rejected, one may well ask why it cannot be applied more generally, to single spin flip simulations of the Ising model for example. The answer is that it can; single spin flip moves could be selected proportional to the Metropolis acceptance probability, which could be stored and updated for each spin in the current microstate.

For a sufficiently simple model, this is workable. For a $d = 2$ dimensional Ising model on a square lattice with single spin-flip dynamics, there are five possible energy changes $8J, 4J, 0, -4J, -8J$. Lists could be kept of spins with each situation, spins selected to flip, and the time clock updated asynchronously, in a manner analogous to the spin exchange dynamics considered above. The extra overhead is worthwhile in cases where the acceptance rate for single spin flips is low, for example at low temperatures in well-separated systems, for which only spins at an interface have much chance to flip.
8.3 Histogram reweighting

[Newman & Barkema p. 211]

In studying phase behavior with simulations, we are always interested in how various quantities of interest such as the average system energy $\langle E \rangle$ vary with temperature or applied field. Naively, it would seem that each time we want to evaluate $\langle E \rangle$ at a new temperature, we would need to perform a new simulation. This would be laborious indeed. Of course, we could plot our results for $\langle E \rangle(\beta)$ as a function of $\beta$, and interpolate between adjacent $\beta$ values.

There is a better way to proceed, in which we make use of the entire set of values generated by a canonical simulation at some temperature $\beta$, to compute average quantities as if the simulation had been performed at a nearby temperature $\beta'$. This method, called histogram reweighting and developed by Ferrenberg and Swendsen, carries out averages in a canonical ensemble at $\beta'$ with the data from a canonical simulation at $\beta$, by using weight factors that account for the different frequency of occurrence of microstates in the two canonical ensembles.

Let $\{Q_i, E_i\}$ be the set of values for some quantity of interest $Q$ and the associated system energy, generated by a canonical simulation at temperature $\beta$. We write the average of $Q$ in the canonical ensemble formally as a Boltzmann-weighted average of $Q_r$ over microstates $r$ which is approximately represented by the average of values $Q_i$ over configurations $i$ in a canonical simulation:

$$\langle Q \rangle(\beta) = \sum_r Q_r e^{-\beta E_r} \frac{Z(\beta)}{Z(\beta')} = \sum_r Q_r P_r(\beta) \approx (1/N) \sum_i Q_i$$

in which $N$ is the number of samples.

We want to evaluate the average $\langle Q \rangle(\beta')$ at some nearby temperature $\beta'$. Formally we may write

$$\langle Q \rangle(\beta') = \sum_r Q_r e^{-\beta' E_r} \frac{Z(\beta)}{Z(\beta')} = \sum_r \left( Q_r e^{-(\beta'-\beta)E_r} \right) e^{-\beta E_r} \frac{Z(\beta)}{Z(\beta')} = \langle Q e^{-(\beta'-\beta)E} \rangle(\beta) \frac{Z(\beta)}{Z(\beta')} \approx \langle Q e^{-(\beta'-\beta)E} \rangle(\beta)$$

The above equation relates the average of $Q$ at the new temperature $\beta'$, to the average of a new quantity $Q e^{-(\beta'-\beta)E}$ at the old temperature $\beta$, times a ratio of partition functions. We can then compute the average of the new quantity in the usual way, as

$$\langle Q e^{-(\beta'-\beta)E} \rangle(\beta) \approx (1/N) \sum_i Q_i e^{-(\beta'-\beta)E_i}$$

The occurrences of the quantity $Q_i$ have been “reweighted” according to the accompanying value of the microstate energy $E_i$, in such a way that microstates will appear in the new sum with the relative importance they would have in the partition sum at temperature $\beta'$. 

8.3. HISTOGRAM REWEIGHTING

The ratio of partition functions can be dealt with in a similar way:

\[
\frac{Z(\beta')}{Z(\beta)} = \frac{\sum_r e^{-\beta' E_r}}{Z(\beta)} = \frac{\sum_r \left( e^{-(\beta' - \beta) E_r} \right) e^{-\beta E_r}}{Z(\beta)} = (1/N) \sum_i e^{-(\beta' - \beta) E_i}
\] (8.17)

Combining these results, we have finally

\[
\langle Q \rangle(\beta') = \frac{\sum_i Q_i e^{-(\beta' - \beta) E_i}}{\sum_i e^{-(\beta' - \beta) E_i}}
\] (8.18)

In practice, the reweighting formula only works well for small temperature shifts, such that there is significant overlap between the microstates that appear in the simulation at the original temperature \(\beta\) and the new temperature \(\beta'\). Of course, if a simulation were performed long enough, it would include many instances of even very rare microstates, so that reweighting would work even for quite distinct \(\beta\) and \(\beta'\). However, for simulations of finite duration, a good rule of thumb is to shift the temperature no further than would move the mean energy by of order the standard deviation of the energy in the simulation.

Figure 8.5: Histogram reweighting for energy histogram \(P(E)\) for a 32 \(\times\) 32 2d Ising model. In (a), data for \(\beta J = 0.4\) (red) are reweighted to \(\beta J = 0.41\) (blue), and compared to simulation results at this temperature (green). In (b), same but with \(\beta J = 0.4\) shifted to \(\beta J = 0.42\).

Fig. 8.5 gives an example of what histogram reweighting looks like when it works, and when it does not work. In figure (a), the red data are an energy histogram \(P(E)\) from a 2d Ising simulation at \(\beta J = 0.4\), and the green data are likewise \(P(E)\) for \(\beta J = 0.41\). The blue points are the result of a “weighted histogram”, using the data from \(\beta J = 0.4\) and weighting each entry \(E_i\) in the histogram with a weight \(e^{-(\beta' - \beta) E_i}\). The reweighted histogram reproduces the simulation at \(\beta J = 0.41\) very well. Note that the red and green histograms overlap substantially; both histograms cover substantially the same range of energies, although with a greater bias towards lower energies for the higher \(\beta\) value (lower temperature).
In figure (b), $P(E)$ data at $\beta J = 0.4$ (red points) are again used construct a weighted histogram for $\beta J = 0.42$ (blue points), and compared to $P(E)$ from a simulation at $\beta J = 0.42$ (green). Now, the two original histograms do not overlap much at lower energies. As a result, the reweighting fails to capture the shape of the $\beta J = 0.42$ histogram at lower energies.

We can make a simple estimate as to how large a shift $\delta \beta$ we can make before reweighting begins to fail, as follows. The basic requirement is that the shift in the mean energy $\delta \langle E \rangle$ resulting from the shift in $\beta$ (i.e., the shift in the center of $P(E)$) should be less than the width of the distribution $P(E)$, as measured by the standard deviation $(\Delta E^2)^{1/2}$.

The shift $\delta \langle E \rangle$ is linearly proportional to $\delta \beta$, from the definition of the (extensive) heat capacity as

$$C = \frac{\partial \langle E \rangle}{\partial T} = -k\beta^2 \frac{\partial \langle E \rangle}{\partial \beta}$$

(8.19)

from which we can write

$$\delta \beta = \frac{(k\beta^2/C)\delta \langle E \rangle < (k\beta^2/C)\langle \Delta E^2 \rangle^{1/2}}{(8.20)}$$

Now recalling the equivalent definition of the heat capacity in terms of fluctuations, namely $C = k\beta^2 \langle \Delta E^2 \rangle$, we can express the energy variance in the above equation in terms of $C$, to obtain after a bit of rearrangement

$$\frac{\delta \beta}{\beta} < \left(\frac{Nc}{k}\right)^{-1/2}$$

(8.21)

in which $c = C/N$ is the heat capacity per site ($N$ is the number of lattice sites in the system).

Evidently, histogram reweighting is more limited for larger systems, because the energy fluctuations are smaller relative to the mean, so that there is less overlap between the energies in two such systems at nearby temperatures. We expect $c/k$ to be a number of order unity (heat capacities are of order $k$ per degree of freedom), so the relative shift $\delta \beta / \beta$ must be less than $O(N^{-1/2})$ for reweighting to work. (In the example of Fig. 8.5, reweighting was successful for $\delta \beta / \beta = 1/40$, which is indeed less than $N^{-1/2} = 1/32$, but failed for $\delta \beta$ two times larger.)

Histogram reweighting can be used to shift an ensemble with respect to parameters other than the temperature. Indeed, any coupling constant (coefficient of a term in the Hamiltonian) can be be shifted, including the external field. To carry out the reweighting in such a case, we must keep track separately of the portion of the energy multiplied by the coefficient to be shifted. For the particular example of shifting an ensemble to a different value of external field, the energy contribution we must keep track of is simply the magnetization $M$, and we can shift $\langle Q \rangle$ according to

$$\langle Q \rangle(h') = \frac{\sum_i Q_i e^{(h'-h)M_i}}{\sum_i e^{(h'-h)M_i}}$$

(8.22)

One of the circumstances in which histogram reweighting is particularly useful, is in identifying the locations of phase transitions. For a second-order transition such as exhibited by the Ising model in zero field as the temperature is reduced, the transition is indicated by a peak in the heat capacity or magnetic susceptibility (rendered nondivergent and smooth by the finite size of the system). To find the precise location of the heat capacity peak, we can use histogram reweighting to move continuously between simulations carried out in the vicinity of the critical temperature, computing the variance in the system energy (by computing reweighted averages of $\langle E^2 \rangle$ and $\langle E \rangle$, from which the variance and thus the heat capacity can be computed).

This technique is also useful for first-order transitions. For the Ising model with an external field, we have a first-order transition below the critical point at zero field. This transition may only be accessed by varying $h$, and evidently occurs at $h = 0$ because of the up-down symmetry of the problem. However, in other lattice models, first-order transitions can be accessed both by cooling.
or by varying some external field, and the location of the transition in $\beta$ or $h$ may not be known \textit{a priori} from symmetry arguments.

Figure 8.6: (a) Sketch of phase diagram for Ising model in T-h plane. First-order transition between spin-up and spin-down phases only accessible by varying field (at $h = 0$). (b) Corresponding sketch for liquid-gas system, or any model in general without $\psi \rightarrow -\psi$ symmetry in the Hamiltonian. First-order transition accessible by cooling as well as by varying field ($\mu$).

In a finite system with non-conserving spin-flip dynamics (including cluster flipping), this transition may be observed by the variation in the histogram of magnetization $M$, which displays a double peak. At the transition, the two peaks for magnetization up and down have equal amplitudes.

In such cases, we can reweight the magnetization values $\{M_i\}$. In fact, we can construct a “reweighted histogram” $P(M; \beta')$ by constructing a histogram in which each value $M_i$ is added to the histogram with weight $e^{-(\beta' - \beta)E_i}$ rather than the usual weight of unity. Then, the magnetization histogram itself may be explored as a continuous function of $\beta$, from which we can determine the value of $\beta$ at which the two peaks have the same height.

Figure 8.7: Sketch of sequence of order parameter probability distributions $P(\psi)$ (with $\psi$ the number of up minus down spins, or A minus B sites), as we cool through a first-order transition.
8.4 WHAM (weighted histogram analysis method)

A closely related extension of histogram reweighting is the weighted histogram analysis method (WHAM), also developed by Ferrenberg and Swendsen. The WHAM method gives a systematic, optimized approach to combine overlapping histograms for some order parameter or other observable, to result in a single approximate partition function.

As an example, consider a Hamiltonian for a system of spins \( \{\sigma_i\} \), with a perturbative term coupled to some observable \( S \), of the form

\[
\beta H(\sigma) = \beta H_0(\sigma) + KS(\sigma)
\]  

The corresponding partition function can be written

\[
Z(K) = \sum_\sigma e^{-\beta H(\sigma)} = \sum_S W(S) e^{-\beta KS}
\]  

in which \( W(S) \) is the density of states as a function of the observable \( S \), i.e., proportional to the number of states with a given value of \( S \) (or between \( S \) and \( S + dS \), for a continuous order parameter).

Now suppose we carry out a series of simulations, at different values \( \{K_n\} \) of the coupling constant \( K \). We store histograms \( \{N_n(S)\} \) of the number of “counts” of \( S \) in each “bin”; let \( n_n \) be the total number of counts for the \( n \)th simulation. The expected variance in the histogram is simply the number of counts,

\[
\Delta^2 N_n(S) = \langle N_n(S) \rangle \approx N_n(s)
\]  

From the histograms, we can write a corresponding set of approximations to the partition function,

\[
Z_n(K) = \sum_S N_n(S) e^{-\beta(K-K_n)S}
\]  

in which we have reweighted each simulation back to some common value of \( K \).

If we repeatedly generated histograms \( N_n(S) \) for the same value \( K_n \), we could compute the average histogram \( \langle N_n(S) \rangle \), which would give the exact density of states for the weighted simulation. Then we could write

\[
\langle Z_n(K) \rangle / n_n = \sum_S P_n(S) e^{-\beta(K-K_n)S}
\]  

in which \( P_n(S) = \langle N_n(S) \rangle / n_n \) is the probability of a given value of \( S \) in the simulation with \( K = K_n \).

We can relate the probability \( P_n(S) \) to the unweighted density of states, as

\[
P_n(S) = \frac{e^{-\beta K_n S} W(S)}{\sum_S e^{-\beta K_n S} W(S)} = \frac{e^{-\beta K_n S} W(S)}{Z(K_n)}
\]  

Using this relation in the above, we have

\[
\frac{\langle Z_n(K) \rangle}{n_n} = \frac{Z(K)}{Z(K_n)}
\]  

Now we introduce the free energy, writing \( Z = e^{-\beta F} \) and \( Z(K_n) = Z_n = e^{-\beta F_n} \), which allows us to write

\[
P_n(S) = W(S) e^{-\beta K_n S + \beta F_n}
\]
With these preliminaries, our goal is to compute the best available estimate for the unweighted density of states \( W(S) \), from our histograms \( \{N_n(S)\} \). Formally, if we had \( \langle N_n(S) \rangle \) (which immediately gives \( P_n(S) \)) instead of just our single instance of \( N_n(S) \) (which has fluctuations and error), we could write

\[
W(S) = \langle N_n(S) \rangle / n_n e^{-\beta F_n + \beta K_n S}
\]  
(8.31)

Without access to \( \langle N_n(S) \rangle \), we are motivated to estimate \( W(S) \) with some sort of weighted average of the histograms we do have:

\[
W(S) \approx \sum_n p_n N_n(S) / n_n e^{-\beta F_n + \beta K_n S}
\]  
(8.32)

in which \( \sum_n p_n = 1 \). The above equation estimates \( W(S) \) as a weighted average of the approximate histograms, times the appropriate reweighting factor for the exact histograms.

How should we choose the weighting factors \( \{p_n\} \)? Intuitively, we should weight the histograms with more counts more heavily, because these histograms are known with more confidence. Ferrenberg and Swendsen provided a systematic way to determine the \( \{p_n\} \) values, by minimizing the variance in our estimate for the density of states \( W(S) \).

Our estimate for \( W(S) \) is stochastic, because it is a linear combination of stochastic quantities \( N_n(S) \). Each \( N_n(S) \) has an average \( \langle N_n(S) \rangle \) and a variance \( \langle \Delta^2 N_n(S) \rangle \), which we argued to be \( \langle N_n(S) \rangle \) itself. Simulations with different values of \( K_n \) are uncorrelated in their fluctuations by definition, because the fluctuations are a property of the (notional) ensemble of multiple simulations at the same \( K_n \). Thus we have

\[
\langle N_n(S) N_m(S) \rangle = \langle N_n(S) \rangle \langle N_m(S) \rangle + \delta_{nm} \langle N_n(S) \rangle
\]  
(8.33)

From these considerations, it is straightforward to show that the variance of \( W(S) \) can be written

\[
\langle \Delta^2 W(S) \rangle = \langle (W(S))^2 \rangle - \langle W(S) \rangle^2
\]  
\[
= \sum_n p_n^2 \langle N_n(S) \rangle f_n^2
\]  
(8.34)

in which we have defined \( f_n = e^{-\beta F_n + \beta K_n S} / n_n \) for convenience.

Now we minimize this expression with respect to the set of \( p_n \), subject to the requirement that \( \sum_n p_n = 1 \). We carry out this constrained minimization using Lagrange multipliers, with the result that

\[
p_n \propto \frac{1}{\langle N_n(S) \rangle f_n^2}
\]  
(8.35)

Now we use our formal result for \( P_n(S) \), in the form

\[
\langle N_n(S) \rangle = n_n W(S) e^{-\beta K_n S + \beta F_n} = W(S) / f_n
\]  
(8.36)

We see that the factor of \( W(S) \), being independent of \( n \), will drop out of the result for \( p(n) \) once we normalize (i.e., enforce \( \sum_n p_n = 1 \)), so we can write

\[
p_n = \frac{(1/f_n)}{\sum_n (1/f_n)}
\]  
(8.37)

Using this result in our estimate for \( W(S) \), we have after a bit of rearrangement

\[
W(S) \approx \frac{\sum_n N_n(S)}{\sum_n n_n e^{-\beta K_n S + \beta F_n}}
\]  
(8.38)
and correspondingly

\[ Z_n = e^{-\beta F_n} \approx \sum_S W(S)e^{-\beta K_n S} \]  \hspace{1cm} (8.39)

These are the WHAM equations. They must be solved iteratively, since \( F_n \) appears in the otherwise explicit expression for \( W(S) \), and \( W(S) \) appears in the companion expression for \( F_n \). Once the WHAM equations are solved, any other quantity \( A(S) \) that depends on \( S \) alone may be averaged over \( W(S) \), according to

\[ \langle A(S) \rangle(K) = \frac{\sum_S A(S)W(S)e^{-\beta KS}}{\sum_S W(S)e^{-\beta KS}} \]  \hspace{1cm} (8.40)
8.5 Umbrella sampling

In many circumstances, microscopic simulations can be used to determine the effective potential acting on some particle positions or other degrees of freedom, which we hope to retain in a more “coarse-grained” description of the system. For example, consider a fluid consisting of solvent molecules, a dilute suspension of colloidal particles of diameter 1000Å, and some concentration of small surfactant spherical micelles of diameter 100Å.

We may simulate all three species in some molecular dynamics or Monte Carlo simulation, but we may be more interested in the behavior of the colloidal particles as a suspension. For example, do they tend to aggregate in the micelle-bearing solution? More generally, what is the apparent “effective potential” \( U(r) \) acting between pairs of colloidal particles?

One simple way to define what we mean by an effective pair potential \( U(r) \), is to select it so that if we performed a Monte Carlo simulation with only the colloidal particles present, they would have the same pair distribution function \( g(r) \) in dilute solution as they have in the full simulation.

Imagine that we had just two colloidal particles in the simulation, with one particle located at the origin, in a spherical volume of radius \( R \). In equilibrium, the second particle would be located at position \( r \) with probability \( P(r) \) given by

\[
P(r) \propto e^{-\beta U(r)} \tag{8.41}
\]

Inverting this relation, we have

\[
U(r) = -\frac{1}{\beta} \log P(r) + \text{const} \tag{8.42}
\]

(In dilute solution, the pair distribution function \( g(r) \) would be proportional to this \( P(r) \); see Section 3.5.3).

An example of this approach is shown in Fig. 8.8, in which a typical histogram result for \( P(r) \) obtained for a model potential \( U(r) \) (with \( \beta = 3 \)) is displayed alongside the inferred \( U(r) \) from Eqn. (8.42). We note that the points so obtained represent \( U(r) \) very well, except for the most repulsive part of the potential at small separations, where we have very few counts in the \( P(r) \) histogram, and hence poor knowledge of exactly how likely it is that the particles will be found at such small separations.

![Figure 8.8: (a) Probability distribution \( P(r) \) for the separation distance \( r \) between a pair of particles interacting with an effective potential of mean force \( U(r) \). (b) \( U(r) = -\frac{1}{\beta} \log P(r) \), shifted to zero at large separations. [Solid curve is actual potential used to generate \( P(r) \) histogram in (a).]
To obtain better information about how likely are these unlikely small separations, we could of course try to simulate for longer times. But this will not be effective for large repulsive potential barriers, for which it will be exponentially unlikely to find particles at such distances. Instead, we employ a technique called “umbrella sampling”: we “encourage” unlikely configurations to appear more frequently by adding a fictitious extra “restraining potential” to the Hamiltonian. Typically, the restraining potential is harmonic, coupling to the degree of freedom of interest.

In the present case, we could introduce an extra term
\[ \delta H = a(|r| - r_0)^2 \] (8.43)
in which the separation \( r_0 \) is chosen to be near the region of interest, and the coefficient \( a \) is chosen strong enough to “pull” the particles into the vicinity of \( |r| = r_0 \). (The technique is called “umbrella sampling” because of the resemblance of the harmonic restraining potential to an upturned umbrella.)

Then the effective potential can be determined from Eqn. (8.42), with the extra interaction subtracted at the end:
\[ U(r) = -1/\beta \log P(r) - a(|r| - r_0)^2 + \text{const} \] (8.44)
Typically, histograms \( P(r) \) are gathered for a series of simulations in which \( r_0 \) takes on a sequence of regularly spaced values, chosen so that the resulting histograms overlap significantly. Each simulation will give reasonable results for the shape of \( U(r) \) over a limited range of \( r \), over which the corresponding \( P(r) \) has appreciable weight. The arbitrary constants for the \( U(r) \) inferred from each restrained simulation are chosen so that the \( U(r) \) collapse to a common curve. A simple example of umbrella sampling is shown in Fig. 8.9.
Figure 8.9: (a) Effective potential $U(r)$ inferred from separation histograms, with different umbrella potentials applied (red, green, blue points). (b) Umbrella potentials (dashed curves, colors match figure (a)) and actual potential $U(r)$. (c) Resulting separation histograms $P(r)$ for each umbrella potential.
8.6 Parallel tempering