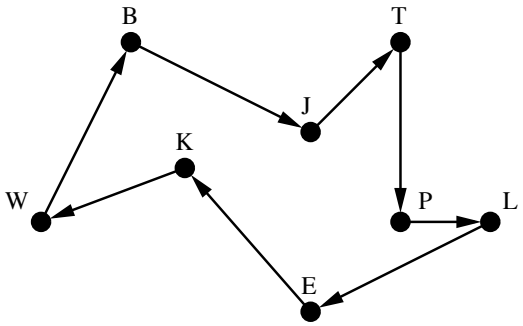

Figure 15.7: What is the optimum route for this random arrangement of $N = 8$ cities? The route begins and ends at city W. A possible route is shown.

15.13 More Applications

You probably are convinced that Monte Carlo methods are powerful, flexible, and applicable to a wide variety of systems. Extensions to the Monte Carlo methods that we have not discussed include multiparticle moves, biased moves where particles tend to move in the direction of the force on them, bit manipulation for Ising-like models, and the use of multiple processors to update different parts of a large system simultaneously. We also have not described the simulation of systems with long-range potentials such as Coulombic systems and dipole-dipole interactions. For these potentials, it is necessary to include the interactions of the particles in the center cell with the infinite set of periodic images.

We conclude this chapter with a discussion of Monte Carlo methods in a context that might seem to have little in common with the types of problems we have discussed. This context is called *multivariate* or *combinatorial optimization,* a fancy way of saying, "How do you find the global minimum of a function that depends on many parameters?" Problems of this type arise in many areas of scheduling and design as well as in physics, biology, and chemistry. We explain the nature of this type of problem for the *traveling salesman problem*, although we would prefer to call it the traveling peddler or traveling salesperson, problem.

Suppose that a salesman wishes to visit N cities and follow a route such that no city is visited more than once and the end of the trip coincides with the beginning. Given these constraints, the problem is to find the optimum route such that the total distance traveled is a minimum. An example of $N = 8$ cities and a possible route is shown in Figure 15.7. All known exact methods for determining the optimal route require a computing time that increases as e^N, and hence, in practice, an exact solution can be found only for a small number of cities. (The traveling salesman problem belongs to a large class of problems known as NP-complete. The NP refers to nondeterministic-polynomial. Such problems cannot be done in a time proportional to a finite polynomial in N on standard computers, though polynomial time algorithms are known for hypothetical nondeterministic (quantum) computers.) What is a reasonable estimate for the maximum number of cities that you can consider without the use of a computer?

To understand the nature of the different approaches to the traveling salesman problem,

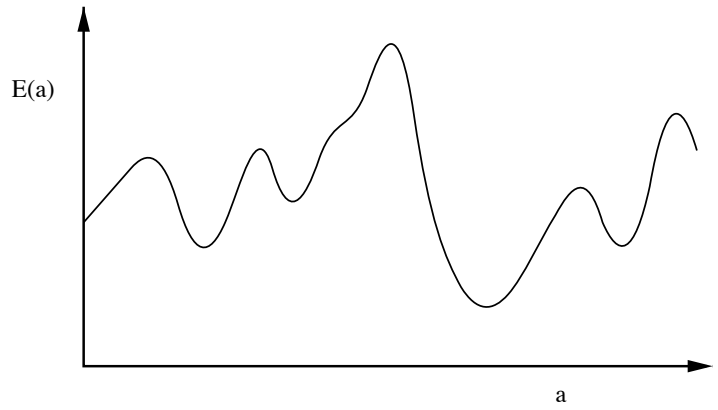

Figure 15.8: Plot of the function $E(a)$ as a function of the parameter a.

consider the plot in Figure 15.8 of the "energy" or "cost" function $E(a)$. We can associate $E(a)$ with the length of the route and interpret a as a parameter that represents the order in which the cities are visited. If $E(a)$ has several local minima, what is a good strategy for finding the global (absolute) minimum of $E(a)$? One way is to vary a systematically and find the value of E everywhere. This way corresponds to an exact enumeration method and would mean knowing the length of each possible route, an impossible task if the number of cities is too large. Another way is to use a *heuristic method*, that is, an approximate method for finding a route that is close to the absolute minimum. One strategy is to choose a value of a, generate a small random change δa, and accept this change if $E(a+\delta a)$ is less than or equal to $E(a)$. This iterative improvement strategy corresponds to a search for steps that lead downhill (see Figure 15.8). Because this strategy usually leads to a local and not a global minimum, it is useful to begin from several initial choices of a and to keep the best result. What would be the application of this type of strategy to the salesman problem?

Because we cannot optimize the path exactly when N becomes large, we have to be satisfied with solving the optimization problem approximately and finding a relatively good local minimum. To understand the motivation for the *simulated annealing* algorithm, consider a seemingly unrelated problem. Suppose we wish to make a perfect single crystal. You might know that we should start with the material at a high temperature at which the material is a liquid melt and then gradually lower the temperature. If we lower the temperature too quickly (a rapid quench), the resulting crystal would have many defects or not become a crystal at all. The gradual lowering of the temperature is known as *annealing*.

The method of annealing can be used to estimate the minimum of $E(a)$. We choose a value of a, generate a small random change δa, and calculate $E(a+\delta a)$. If $E(a+\delta a)$ is less than or equal to $E(a)$, we accept the change. However, if $\Delta E = E(a+\delta a) - E(a) > 0$, we accept the change with a probability $p = e^{-\Delta E/T}$, where T is an effective temperature. This procedure is the familiar Metropolis algorithm with the temperature playing the role of a control parameter. The *simulated annealing* process consists of first choosing a value for T for which most moves are accepted, and then gradually lowering the temperature. At each temperature, the simulation

should last long enough for the system to reach quasiequilibrium. The annealing schedule, that is, the rate of temperature decrease, determines the quality of the solution.

The idea is to allow moves that result in solutions of worse quality than the current solution (uphill moves) in order to escape from local minima. The probability of doing such a move is decreased during the search. The slower the temperature is lowered, the higher the chance of finding the optimum solution, but the longer the run time. The effective use of simulated annealing depends on finding a annealing schedule that yields good solutions without taking too much time. It has been proven that if the cooling rate is sufficiently slow, the absolute (global) minimum will eventually be reached. The bounds for "sufficiently slow" depend on the properties of the search landscape (the nature of $E(a)$) and are exceeded for most problems of interest. However, simulated annealing is usually superior to conventional heuristic algorithms.

The moral of the simulated annealing method is that sometimes it is necessary to climb a hill to reach a valley. The first application of the method of simulated annealing was to the optimal design of computers. In Problem 15.31 we apply this method to the traveling salesman problem.

Problem 15.31. Simulated annealing and the traveling salesman problem

a. Generate a random arrangement of $N = 8$ cities in a square of linear dimension $L = \sqrt{N}$ and calculate the optimum route by hand. Then write a Monte Carlo program and apply the method of simulated annealing to this problem. For example, use two arrays to store the x and y coordinate of each city and an array to store the distances between them. The state of the system, that is, the route represented by a sequence of cities, can be stored in another array. The length of this route is associated with the energy of an imaginary thermal system. A reasonable choice for the initial temperature is one that is the same order as the initial energy. One way to generate a random rearrangement of the route is to choose two cities at random and to interchange the order of visit. Choose this method or one that you devise and find a reasonable annealing schedule. Compare your annealing results to exact results whenever possible. Extend your results to larger N, for example, $N = 12$, 24, and 48. For a given annealing schedule, determine the probability of finding a route of a given length. More suggestions can be found in the references.

b. The microcanonical Monte Carlo algorithm (demon) discussed in Section 15.3 also can be used to do simulated annealing. The advantages of the demon algorithm are that it is deterministic and allows large temperature fluctuations. One way to implement the analog of simulated annealing is to impose a maximum value on the energy of the demon, $E_{d,\max}$, which is gradually decreased. Guo et al. choose $E_{d,\max}$ to be initially equal to $\sqrt{N}/4$. Their results are comparable to the usual simulated annealing method, but require approximately half the CPU time. Apply this method to the same city positions that you considered in part (a) and compare your results.

15.14 Projects

Many of the original applications of Monte Carlo methods were done for systems of approximately one hundred particles and lattices of order 32^2 spins. It would be instructive to redo many of these applications with much better statistics and with larger system sizes. In the following, we

discuss some additional recent developments, but we have omitted other important topics such as Brownian dynamics and umbrella sampling. More ideas for projects can be found in the references.

Project 15.32. Overcoming critical slowing down

The usual limiting factor of most simulations is the speed of the computer. Of course, one way to overcome this problem is to use a faster computer. Near a continuous phase transition, the most important limiting factor on even the fastest available computers is the existence of critical slowing down (see Problem 15.19). In this project we discuss the nature of critical slowing down and ways of overcoming it in the context of the Ising model.

As we have mentioned, the existence of critical slowing down is related to the fact that the size of the correlated regions of spins becomes very large near the critical point. The large size of the correlated regions and the corresponding divergent behavior of the correlation length ξ near T_c implies that the time τ required for a region to lose its coherence becomes very long if a *local* dynamics is used. At $T = T_c$, $\tau \sim L^z$ for $L \gg 1$. For single spin flip algorithms, $z \approx 2$, and τ becomes very large for $L \gg 1$. On a serial computer, the CPU time needed to obtain n configurations increases as L^2, the time needed to visit L^2 spins. This factor of L^2 is expected and not a problem because a larger system contains proportionally more information. However, the time needed to obtain n approximately *independent* configurations is of order $\tau L^2 \sim L^{2+z} \approx L^4$ for the Metropolis algorithm. We conclude that an increase of L by a factor of 10 requires 10^4 more computing time. Hence, the existence of critical slowing down limits the maximum value of L that can be considered.

If we are interested only in the static properties of the Ising model, the choice of dynamics is irrelevant as long as the transition probability satisfies the detailed balance condition (15.18). It is reasonable to look for a *global* algorithm for which groups or *clusters* of spins are flipped simultaneously. We already are familiar with cluster properties in the context of percolation (see Chapter 12). A naive definition of a cluster of spins might be a domain of parallel nearest neighbor spins. We can make this definition explicit by introducing a bond between any two nearest neighbor spins that are parallel. The introduction of a bond between parallel spins defines a site-bond percolation problem. More generally, we may assume that such a bond exists with probability p and that this bond probability depends on the temperature T.

The dependence of p on T can be determined by requiring that the percolation transition of the clusters occurs at the Ising critical point, and by requiring that the critical exponents associated with the clusters be identical to the analogous thermal exponents. For example, we can define a critical exponent ν_p to characterize the divergence of the connectedness length of the clusters near p_c. The analogous thermal exponent ν quantifies the divergence of the thermal correlation length ξ near T_c. We will argue in the following that these (and other) critical exponents are identical if we define the bond probability as

$$p = 1 - e^{-2J/kT}. \qquad \text{(bond probability)} \tag{15.79}$$

The relation (15.79) holds for any spatial dimension. What is the value of p at $T = T_c$ for the two-dimensional Ising model on the square lattice?

A simple argument for the temperature-dependence of p in (15.79) is as follows. Consider the two configurations in Figure 15.9 which differ from one another by the flip of the cluster of two spins. In Figure 15.9(a) the six nearest neighbor spins of the cluster are in the opposite

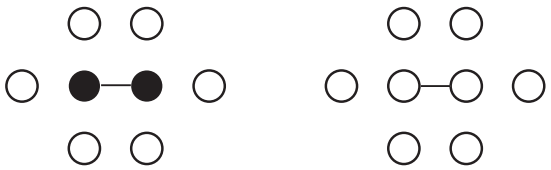

Figure 15.9: (a) A cluster of two up spins. (b) A cluster of two down spins, respectively. The filled and open circles represent the up and down spins. Note the bond between the two spins in the cluster. Adapted from Newman and Barkema.

direction and hence are not part of the cluster. Thus the probability of this configuration with a cluster of two spins is $p\,e^{-\beta J}e^{6\beta J}$, where p is the probability of a bond between the two up spins, $e^{-\beta J}$ is proportional to the probability that these two spins are parallel, and $e^{6\beta J}$ is proportional to the probability that the six nearest neighbors are antiparallel. In Figure 15.9(b) the cluster spins have been flipped and the possible bonds between the cluster spins and its nearest neighbors have to be "broken." The probability of this configuration with a cluster of two (down) spins is $p(1-p)^6 e^{-\beta J}e^{-6\beta J}$, where the factor of $1-p)^6$ is the probability that the six nearest neighbor spins are not part of the cluster. Because we want the probability that a cluster is flipped to be unity, we need to have the probability of the two configurations and their corresponding clusters be the same. Hence, we must have

$$pe^{-\beta J}e^{6\beta J} = p(1-p)^6 e^{-\beta J}e^{-6\beta J}, \tag{15.80}$$

or $(1-p)^6 = e^{-12\beta J}$. It is straightforward to solve for p and obtain the relation (15.79).

Now that we know how to generate clusters of spins, we can use these clusters to construct a global dynamics instead of only flipping one spin at a time as in the Metropolis algorithm. The idea is to grow a single (site-bond) percolation cluster in a way that is analogous to the single (site) percolation cluster algorithm discussed in Section 13.1. The algorithm can be implemented by the following steps:

i. Choose a seed spin at random. Its four nearest neighbor sites (on the square lattice) are the perimeter sites. Form an ordered array corresponding to the perimeter spins that are parallel to the seed spin and define a counter for the total number of perimeter spins.

ii. Choose the first spin in the ordered perimeter array. Remove it from the array and replace it by the last spin in the array. Generate a random number r. If $r \leq p$, the bond exists between the two spins, and the perimeter spin is added to the cluster.

iii. If the spin is added to the cluster, inspect its parallel perimeter spins. If any of these spins are not already part of the cluster, add them to the end of the array of perimeter spins.

iv. Repeat steps (ii) and (iii) until no perimeter spins remain.

v. Flip all the spins in the single cluster.

This algorithm is known as single cluster flip or *Wolff* dynamics. Note that bonds rather than sites are tested so that a spin might have more than one chance to join a cluster. In the following, we consider both the static and dynamical properties of the two-dimensional Ising model using the Wolff algorithm to generate the configurations.

a. Modify your program for the Ising model on a square lattice so that single cluster flip dynamics (the Wolff algorithm) is used. Compute the mean energy and magnetization for $L = 16$ as a function of T for $T = 2.0$ to 2.7 in steps of 0.1. Compare your results to those obtained using the Metropolis algorithm. How many cluster flips do you need to obtain comparable accuracy at each temperature? Is the Wolff algorithm more efficient at every temperature near T_c?

b. Fix T at the critical temperature of the infinite lattice ($T_c = 2/\ln(1+\sqrt{2})$) and use finite size scaling to estimate the values of the various static critical exponents, for example, γ and α. Compare your results to those obtained using the Metropolis algorithm.

c. Because we are generating site-bond percolation clusters, we can study their geometrical properties as we did for site percolation. For example, measure the distribution sn_s of cluster sizes at $p = p_c$ (see Problem 13.3). How does n_s depend on s for large s (see Project 13.15)? What is the fractal dimension of the clusters in the Ising model at $T = T_c$?

d. The natural unit of time for single cluster flip dynamics is the number of cluster flips $t_{\rm cf}$. Measure $C_M(t_{\rm cf})$ and/or $C_E(t_{\rm cf})$ and estimate the corresponding correlation time $\tau_{\rm cf}$ for $T = 2.5$, 2.4, 2.3, and T_c for $L = 16$. As discussed in Problem 15.19, $\tau_{\rm cf}$ can be found from the relation, $\tau_{\rm cf} = \sum_{t_{\rm cf}=1} C(t_{\rm cf})$. The sum is cut-off at the first negative value of $C(t_{\rm cf})$. Estimate the value of $z_{\rm cf}$ from the relation $\tau_{\rm cf} = L^{z_{\rm cf}}$.

e. To compare our results for the Wolff algorithm to our results for the Metropolis algorithm, we should use the same unit of time. Because only a fraction of the spins are updated at each cluster flip, the time $t_{\rm cf}$ is not equal to the usual unit of time, which corresponds to an update of the entire lattice or one Monte Carlo step per spin. We have that τ measured in Monte Carlo steps per spin is related to $\tau_{\rm cf}$ by $\tau = \tau_{\rm cf}\langle c\rangle/L^2$, where $\langle c\rangle$ is the mean number of spins in the single clusters, and L^2 is the number of spins in the entire lattice. Verify that the mean cluster size scales as $\langle c\rangle \sim L^{\gamma/\nu}$ with $\gamma = 7/4$ and $\nu = 1$. (The quantity $\langle c\rangle$ is the same quantity as the mean cluster size S defined in Chapter 12. The exponents characterizing the divergence of the various properties of the clusters are identical to the analogous thermal exponents.)

f. To obtain the value of z that is directly comparable to the value found for the Metropolis algorithm, we need to rescale the time as in part (e). We have that $\tau \sim L^z \propto L^{z_{\rm cf}} L^{\gamma/\nu} L^{-d}$. Hence, z is related to the measured value of $z_{\rm cf}$ by $z = z_{\rm cf} - (d - \gamma/\nu)$. What is your estimated value of z? (It has been estimated that $z_{\rm cf} \approx 0.50$ for the $d = 2$ Ising model, which would imply that $z \approx 0.25$.)

g. One of the limitations of the usual implementation of the Metropolis algorithm is that only one spin is flipped at a time. However, there is no reason why we could not choose f spins at random, compute the change in energy ΔE for flipping these f spins, and accepting or rejecting the trial move in the usual way according to the Boltzmann probability. Explain why this generalization of the Metropolis algorithm would be very inefficient, especially if $f >> 1$. We conclude that the groups of spins to be flipped must be chosen with the physics of the system in mind and not simply at random.

Another cluster algorithm is to assign all bonds between parallel spins with probability p. As usual, no bonds are included between sites that have different spin orientations. From this configuration of bonds, we can form clusters of spins using one of the cluster identification algorithms we discussed in Chapter 12. The smallest cluster contains a single spin. After the clusters have been identified, all the spins in each cluster are flipped with probability 1/2. This algorithm is known as the *Swendsen-Wang* algorithm and preceded the Wolff algorithm. Because the Wolff algorithm is easier to program and gives a smaller value of z than the Swendsen-Wang algorithm for the $d = 3$ and $d = 4$ Ising models, the Wolff algorithm is more commonly used.

Project 15.33. Invaded cluster algorithm

In Problem 13.7 we found that invasion percolation is an example of a self-organized critical phenomenon. In this cluster growth algorithm, random numbers are independently assigned to the bonds of a lattice. The growth starts from the seed sites of the left-most column. At each step the cluster grows by the occupation of the perimeter bond with the smallest random number. The growth continues until the cluster satisfies a stopping condition. We found that if we stop adding sites when the cluster is comparable in extent to the linear dimension L, then the fraction of bonds that are occupied approaches the percolation threshold p_c as $L \to \infty$. The invaded percolation algorithm automatically finds the percolation threshold!

Machta and co-workers have used this idea to find the critical temperature of a spin system without knowing its value in advance. For simplicity, we will discuss their algorithm in the context of the Ising model, although it can be easily generalized to the q-state Potts model (see the references). Consider a lattice on which there is a spin configuration $\{s_i\}$. The bonds of the lattice are assigned a random order. Bonds (i, j) are tested in this assigned order to see if s_i is parallel to s_j. If so, the bond is occupied and spins i and j are part of the same cluster. Otherwise, the bond is not occupied and is not considered for the remainder of the current Monte Carlo step. The set of occupied bonds partitions the lattice into clusters of connected sites. The clusters can be found using the Newman-Ziff algorithm (see Section 12.3). The cluster structure evolves until a stopping condition is satisfied. Then a new spin configuration is obtained by flipping each cluster with probability 1/2, thus completing one Monte Carlo step. The fraction f of bonds that were occupied during the growth process and the energy of the system are measured. The bonds are than randomly reordered and the process begins again. Note that the temperature is not an input parameter.

If open boundary conditions are used, the appropriate stopping rule is that a cluster spans the lattice (see Chapter 12, page **??**). For periodic boundary conditions, the spanning rule discussed in Project 12.17 is appropriate.

Write a program to simulate the invaded cluster algorithm for the Ising model on the square lattice. Start with all spins up and determine how many Monte Carlo steps are needed for equilibration. How does this number compare to that required by the Metropolis algorithm at the critical temperature for the same value of L? An estimate for the critical temperature can be found from the relation (15.79) with f corresponding to p.

After you are satisfied that your program is working properly, determine the dependence of the critical temperature on the concentration c of non-magnetic impurities. That is, randomly place non-magnetic impurities on a fraction c of the sites.

Project 15.34. Physical test of random number generators

In Section 7.9 we discussed various statistical tests for the quality of random number generators. In this project we will find that the usual statistical tests might not be sufficient for determining the quality of a random number generator for a particular application. The difficulty is that the quality of a random number generator for a specific application depends in part on how the subtle correlations that are intrinsic to all deterministic random number generators couple to the way that the random number sequences are used. In this project we explore the quality of two random number generators when they are used to implement single spin flip dynamics (the Metropolis algorithm) and single cluster flip dynamics (Wolff algorithm) for the two-dimensional Ising model.

a. Write methods to generate sequences of random numbers based on the linear congruential algorithm

$$x_n = 16807\, x_{n-1} \bmod (2^{31} - 1), \tag{15.81}$$

and the generalized feedback shift register (GFSR) algorithm

$$x_n = x_{n-103} \oplus x_{n-250}. \tag{15.82}$$

In both cases x_n is the nth random number. Both algorithms require that x_n be divided by the largest possible value of x_n to obtain numbers in the range $0 \leq x_n < 1$. The GFSR algorithm requires bit manipulation. Which random number generator does a better job of passing the various statistical tests discussed in Problem 7.35?

b. Use the Metropolis algorithm and the linear congruential random number generator to determine the mean energy per spin E/N and the specific heat (per spin) C for the $L = 16$ Ising model at $T = T_c = 2/\ln(1+\sqrt{2})$. Make ten independent runs (that is, ten runs that use different random number seeds), and compute the standard deviation of the means σ_m from the ten values of E/N and C, respectively. Published results by Ferrenberg, Landau, and Wong are for 10^6 Monte Carlo steps per spin for each run. Calculate the differences δ_e and δ_c between the average of E/N and C over the ten runs and the exact values (to five decimal places) $E/N = -1.45306$ and $C = 1.49871$. If the ratio δ/σ_m for the two quantities is order unity, then the random number generator does not appear to be biased. Repeat your runs using the GFSR algorithm to generate the random number sequences. Do you find any evidence of statistical bias?

c. Repeat part (b) using Wolff dynamics. Do you find any evidence of statistical bias?

d. Repeat the computations in parts (b) and (c) using the random number generator supplied with your programming language.

Project 15.35. Nucleation and the Ising model

a. Equilibrate the two-dimensional Ising model at $T = 4T_c/9$ and $B = 0.3$ for a system with $L \geq 50$. What is the equilibrium value of m? Then flip the magnetic field so that it points down, that is, $B = -0.3$. Use the Metropolis algorithm and plot m as a function of the time t (the number of Monte Carlo steps per spin). What is the qualitative behavior of $m(t)$? Does it fluctuate about a positive value for a time long enough to determine various averages? If so, the system can be considered to have been in a *metastable state*. Watch the spins evolve for a time

before m changes sign. Visually determine a place in the lattice where a "droplet" of the stable phase (down spins) first appears and then grows. Change the random number seed and rerun the simulation. Does the droplet appear in the same spot at the same time? Can the magnitude of the field be increased further or is there an upper bound above which a metastable state is not well defined?

b. As discussed in Project 15.32, we can define clusters of spins by placing a bond with probability p between parallel spins. In this case there is an external field and the proper definition of the clusters is more difficult. For simplicity, assume that there is a bond between all nearest-neighbor down spins and find all the clusters of down spins. One way to identify the droplet that initiates the decay of the metastable state is to monitor the number of spins in the largest cluster as a function of time after the quench. At what time does the number of spins in the largest cluster begin to grow quickly? This time is an estimate of the *nucleation time*. Another way of estimating the nucleation time is to follow the evolution of the center of mass of the largest cluster. For early times after the quench, the center of mass position has large fluctuations. However, at a certain time, these fluctuations decrease considerably, which is another criterion for the nucleation time. What is the order of magnitude of the nucleation time?

c. While the system is in a metastable state, clusters of down spins grow and shrink randomly until eventually one of the clusters becomes large enough to grow, nucleation occurs, and the system decays to its stable macroscopic state. The cluster that initiates this decay is called the nucleating droplet. This type of nucleation is due to spontaneous thermal fluctuations and is called *homogeneous nucleation*. Although the criteria for the nucleation time that we used in part (b) are plausible, they are not based on fundamental considerations. From theoretical considerations the nucleating droplet can be thought of as a cluster that just makes it to the top of the saddle point of the free energy that separates the metastable and stable states. We can identify the nucleating droplet by using the fact that a saddle point structure should initiate the decay of the metastable state 50% of the time. The idea is to save the spin configurations at regular intervals at about the time that nucleation is thought to have occurred. We then restart the simulation using a saved configuration at a certain time and use a different random number sequence to flip the spins. If we have intervened at a time such that the largest cluster decays in more than 50% of the trials, then the intervention time (the time at which we changed the random number seed) is before nucleation. Similarly, if less than 50% of the clusters decay, the intervention is after the nucleation time. The nucleating droplet is the cluster that decays in approximately half of the trial interventions. Because we need to do a number of interventions (usually in the range 20–100) at different times, the intervention method is much more CPU intensive than the other criteria. However, it has the advantage that it has a sound theoretical basis. Redo some of the simulations that you did in part (b) and compare the different estimates of the nucleation time. What is the nature and size of the nucleating droplet? If time permits, determine the probability that the system nucleates at time t for a given quench depth. (Measure the time t after the flip of the field.)

d. *Heterogeneous nucleation* occurs in nature because of the presence of impurities, defects, or walls. One way of simulating heterogeneous nucleation in the Ising model is to fix a certain number of spins in the direction of the stable phase (down). For simplicity, choose the impurity to be five spins in the shape of a + sign. What is the effect of the impurity on the lifetime of

the metastable state? What is the probability of droplet growth on and off the impurity as a function of quench depth B?

e. The questions raised in parts (b)–(d) become even more interesting when the interaction between the spins extends beyond nearest neighbors. Assume that a given spin interacts with all spins that are within a distance R with an interaction strength of $4J/q$, where q is the number of spins within the interaction range R. (Note that $q = 4$ for nearest neighbor interactions on the square lattice.) A good choice is $R = 10$, although your preliminary simulations should be for smaller R. How does the value of T_c change as R is increased?

Project 15.36. The n-fold way: Simulations at low temperature

Monte Carlo simulations become very inefficient at low temperatures because almost all trial configurations will be rejected. For example, consider an Ising model for which all spins are up, but a small magnetic field is applied in the negative direction. The equilibrium state will have most spins pointing down. Nevertheless, if the magnetic field is small and the temperature is low enough, equilibrium will take a very long time to occur.

What we need is a more efficient way of sampling configurations if the acceptance probability is low. The *n-fold way* algorithm is one such method. The idea is to accept more low probability configurations, but to weight them appropriately. If we use the usual Metropolis rule, then the probability of flipping the ith spin is

$$p_i = \min\bigl[1, e^{-\Delta E/kT}\bigr]. \tag{15.83}$$

One limitation of the Metropolis algorithm is that it becomes very inefficeint if the probabilities p_i are very small. If we sum over all the spins, then we can define the total weight

$$Q = \sum_i p_i. \tag{15.84}$$

The idea is to choose a spin to flip (with probability one) by computing a random number, r_Q, between 0 and Q and finding spin i that satisfies the condition:

$$\sum_{k=0}^{i-1} p_k \le r_Q < \sum_{k=0}^{i} p_k. \tag{15.85}$$

There are two more ingredients we need to make this algorithm practical. We need to determine how long a configuration would remain unchanged if we had used the Metropolis algorithm. Also the algorithm would be very inefficient because on average the computation of which spin to flip from (15.85) would take $O(N)$ computations. This second problem can be easily overcome by realizing that there are only a few possible values of p_i. For example, for the Ising model on a square lattice in a magnetic field, there are only $n = 10$ possible values of p_i. Thus, instead of (15.85), we have

$$\sum_{\alpha=0}^{i-1} n_\alpha p_\alpha \le r_Q < \sum_{\alpha=0}^{i} n_\alpha p_\alpha, \tag{15.86}$$

where α labels one of the n possible values of p_i or classes, and n_α is the number of spins in class α. Hence, instead of $O(N)$ calculations, we need to perform only $O(n)$ calculations. Once we know which class we have chosen, we can randomly flip one of the spins in that class.

Next we need to determine the time spent in a configuration. The probability in one Metropolis Monte Carlo step of choosing a spin at random is $1/N$, and the probability of actually flipping that spin is p_i, which is given by (15.83). Thus, the probability of flipping any spin is

$$\frac{1}{N}\sum_{i=0}^{N-1} p_i = \frac{1}{N}\sum_{\alpha=0}^{n-1} n_\alpha p_\alpha = \frac{Q}{N}, \tag{15.87}$$

The probability of not flipping any spin is $q \equiv 1 - Q/N$, and the probability of not flipping after s steps is q^s. Thus, if we generate a random number r between 0 and 1, the time s in Monte Carlo steps per spin to remain in the current configuration will be determined by solving

$$q^{s-1} \le r < q^s. \tag{15.88}$$

If $Q/N << 1$, then both sides of (15.88) are approximately equal, and we can approximate s by

$$s \approx \frac{\ln r}{\ln q} = \frac{\ln r}{\ln(1 - Q/N)} \approx -\frac{N}{Q}\ln r. \tag{15.89}$$

That is, we would have to wait s Monte Carlo steps per spin on the average before we would flip a spin using the Metropolis algorithm. Note that the random number r in (15.88) and (15.89) should not be confused with the random number r_Q in (15.86).

The n-fold algorithm can be summarized by the following steps:

i. Start with an initial configuration and determine the class to which each spin belongs. Store all the possible values of p_i in an array. Compute Q. Store in an array the number of spins in class α, n_α.

ii. Determine s from (15.89). Accumulate any averages such as the energy and magnetization weighted by s. Also, accumulate the total time `tTotal += s`.

iii. Choose a class of spin using (15.86) and randomly choose which spin in the chosen class to flip.

iv. Update the classes of the chosen spin and its four neighbors.

v. Repeat steps (ii)–(iv).

To conveniently carry out step (iv) set up the following arrays: `spinClass[i]` returns the class of the ith spin, `spinInClass[k][alpha]` returns the kth spin in class α, and `spinIndex[i]` returns the value of k for the ith spin to use in the array `spinInClass[k][alpha]`. If we define the local field of a spin by the sum of the fields of its four neighbors, then this local field can take on the values $\{-4, -2, 0, 2, 4\}$. The ten classes correspond to these five local field values and the center spin equal to -1 plus these five local field values and the center spin equal to $+1$. If we order these ten classes from 0 to 9, then the class of a spin that is flipped changes by $+5$ mod 10 and the class of a neighbor changes by the new spin value equal to ± 1.

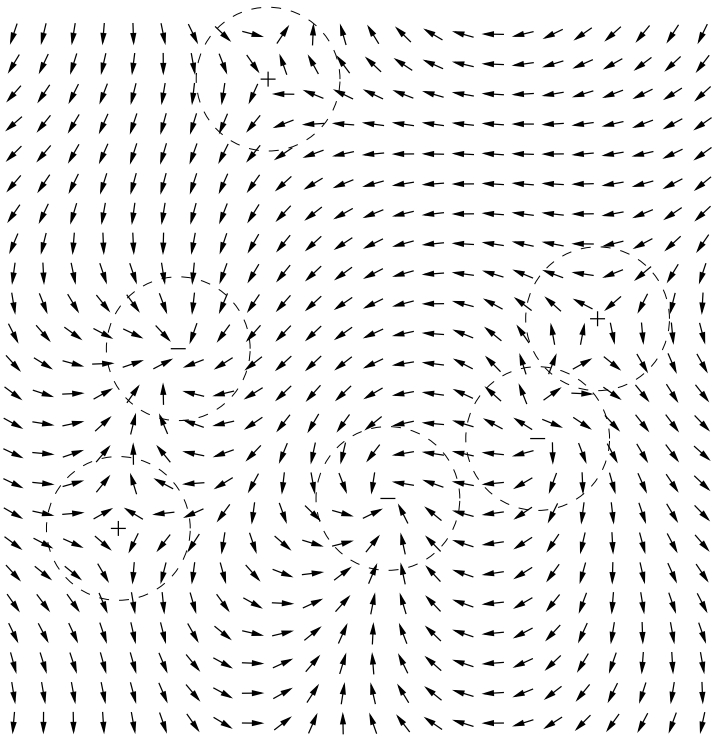

Figure 15.10: A typical configuration of the planar model on a 24×24 square lattice that has been quenched from $T = \infty$ to $T = 0$ and equilibrated for 200 Monte Carlo steps per spin after the quench. Note that there are six vortices. The circle around each vortex is a guide to the eye and is not meant to indicate the size of the vortex.

a. Write a program to implement the n-fold way algorithm for the Ising model on a square lattice with an applied magnetic field. Check your program by comparing various averages at a few temperatures with the results from your program using the Metropolis algorithm.

b. Choose the magnetic field $B = -0.5$ at the temperature $T = 1$. Begin with an initial configuration of all spins up, and use the n-fold way to estimate how long it takes before the majority of the spins flip. Do the same simulation using the Metropolis algorithm. Which algorithm is more efficient?

c. Repeat part (b) for other temperature and field values. For what conditions is the n-fold way algorithm more efficient than the standard Metropolis algorithm?

d. Repeat part (b) for different values of the magnetic field and plot the number of Monte Carlo steps needed to flip the spins as a function of $1/|B|$, for values of B from 0 to ≈ 3. Average over at least 10 starting configurations for each field value.

Project 15.37. The Kosterlitz-Thouless transition

The planar model (also called the x-y model) consists of spins of unit magnitude that can point in any direction in the x-y plane. The energy or Hamiltonian function of the planar model in zero magnetic field can be written as

$$E = -J \sum_{i,j=nn(i)} [s_{i,x}s_{j,x} + s_{i,y}s_{j,y}], \tag{15.90}$$

where $s_{i,x}$ represents the x-component of the spin at the ith site, J measures the strength of the interaction, and the sum is over all nearest neighbors. We can rewrite (15.90) in a simpler form by substituting $s_{i,x} = \cos\theta_i$ and $s_{i,y} = \sin\theta_i$. The result is

$$E = -J \sum_{i,j=nn(i)} \cos(\theta_i - \theta_j), \tag{15.91}$$

where θ_i is the angle that the ith spin makes with the x axis. The most studied case is the two-dimensional model on a square lattice. In this case the mean magnetization $\langle \mathbf{M} \rangle = 0$ for all temperatures $T > 0$, but nevertheless, there is a phase transition at a nonzero temperature, $T_{\rm KT}$, which is known as the Kosterlitz-Thouless (KT) transition. For $T \leq T_{\rm KT}$, the spin-spin correlation function $C(r)$ decreases as a power law; for $T > T_{\rm KT}$, $C(r)$ decreases exponentially. The power law decay of $C(r)$ for $T \leq T_{\rm KT}$ implies that every temperature below $T_{\rm KT}$ acts as if it were a critical point. We say that the planar model has a line of critical points. In the following, we explore some of the properties of the planar model and the mechanism that causes the transition.

a. Write a program that uses the Metropolis algorithm to simulate the planar model on a square lattice using periodic boundary conditions. Because θ and hence the energy of the system is a continuous variable, it is not possible to store the previously computed values of the Boltzmann factor for each possible value of ΔE. Instead, of computing $e^{-\beta\Delta E}$ for each trial change, it is faster to set up an array `w` such that the array element `w(j)` $= e^{-\beta\Delta E}$, where `j` is the integer part of $1000\Delta E$. This procedure leads to an energy resolution of 0.001, which should be sufficient for most purposes.

b. One way to show that the magnetization $\langle \mathbf{M} \rangle$ vanishes for all T is to compute $\langle \theta^2 \rangle$, where θ is the angle that a spin makes with the magnetization $\mathbf{M}$ for any given configuration. (Although the mean magnetization vanishes, $\mathbf{M} \neq 0$ at any given instant.) Compute $\langle \theta^2 \rangle$ as a function of the number of spins N at $T = 0.1$, and show that $\langle \theta^2 \rangle$ diverges as $\ln N$. Begin with a 4×4 lattice and choose the maximum change in θ_i to be $\Delta\theta_{\rm max} = 1.0$. If necessary, change $\theta_{\rm max}$ so that the acceptance probability is about 40%. If $\langle \theta^2 \rangle$ diverges, then the fluctuations in the direction of the spins diverges, which implies that there is no preferred direction for the spins, and hence the mean magnetization vanishes.

c. Modify your program so that an arrow is drawn at each site to show the orientation of each spin. You can use the `Vector2DFrame` to draw a lattice of arrows. Look at a typical configuration and analyze it visually. Begin with a 32×32 lattice with spins pointing in random directions and do a temperature quench to $T = 0.5$. (Simply change the value of β in the Boltzmann probability.) Such a quench should lock-in some long lived, but metastable vortices. A vortex is a region of the lattice where the spins rotate by at least 2π as your eye moves around a

closed path (see Figure 15.10). To determine the center of a vortex, choose a group of four spins that are at the corners of a unit square, and determine whether the spins rotate by $\pm 2\pi$ as your eye goes from one spin to the next in a counterclockwise direction around the square. Assume that the difference between the direction of two neighboring spins, $\delta\theta$, is in the range $-\pi < \delta\theta < \pi$. A total rotation of $+2\pi$ indicates the existence of a positive vortex, and a change of -2π indicates a negative vortex. Count the number of positive and negative vortices. Repeat these observations for several configurations. What can you say about the number of vortices of each sign?

d. Write a method to determine the existence of a vortex for each 1×1 square of the lattice. Represent the center of the vortices using a different symbol to distinguish between a positive and a negative vortex. Do a Monte Carlo simulation to compute the mean energy, the specific heat, and number of vortices in the range from $T = 0.5$ to $T = 1.5$ in steps of 0.1. Use the last configuration at the previous temperature as the first configuration for the next temperature. Begin at $T = 0.5$ with all $\theta_i = 0$. Draw the vortex locations for the last configuration at each temperature. Use at least 1000 Monte Carlo steps per spin at each temperature to equilibrate, and at least 5000 Monte Carlo steps per spin for computing the averages. Use an 8×8 or 16×16 lattice if your computer resources are limited, and larger lattices if you have sufficient resources. Describe the T dependence of the energy, the specific heat, and the vorticity (equal to the number of vortices per unit area). Plot the logarithm of the vorticity versus T for $T < 1.1$. What can you conclude about the T-dependence of the vorticity? Explain why this form is reasonable. Describe the vortex configurations. At what temperature do you find a vortex that appears to be free, that is, a vortex that is not obviously paired with another vortex of opposite sign?

e. The Kosterlitz-Thouless theory predicts that the susceptibility χ diverges above the transition as

$$\chi \sim A\, e^{b/\epsilon^{\nu}}, \tag{15.92}$$

where ϵ is the reduced temperature $\epsilon = (T - T_{\rm KT})/T_{\rm KT}$, $\nu = 0.5$, and A and b are nonuniversal constants. Compute χ from the relation (15.21) with $\mathbf{M} = 0$. Assume the exponential form (15.92) for χ in the range $T = 1$ and $T = 1.2$ with $\nu = 0.7$, and find the best values of $T_{\rm KT}$, A, and b. (Although theory predicts $\nu = 0.5$, simulations for small systems indicate that $\nu = 0.7$ gives a better fit.) One way to determine $T_{\rm KT}$, A, and b is to assume a value of $T_{\rm KT}$ and then do a least squares fit of $\ln \chi$ to determine A and b. Choose the set of parameters that minimizes the variance of $\ln \chi$. How does your estimated value of $T_{\rm KT}$ compare with the temperature at which free vortices first appear? At what temperature does the specific heat have a peak? The Kosterlitz-Thouless theory predicts that the specific heat peak does not occur at $T_{\rm KT}$. This prediction has been confirmed by simulations (see Tobochnik and Chester). To obtain quantitative results, you will need lattices larger than 32×32.

Project 15.38. The classical Heisenberg model in two dimensions

The energy or Hamiltonian of the classical Heisenberg model is similar to the Ising model and the planar model, except that the spins can point in any direction in three dimensions. The energy in

zero external magnetic field is

$$E = -J \sum_{i,j=\mathrm{nn}(i)}^{N} \mathbf{s}_i \cdot \mathbf{s}_j = -J \sum_{i,j=\mathrm{nn}(i)}^{N} [s_{i,x}s_{j,x} + s_{i,y}s_{j,y} + s_{i,z}s_{j,z}], \tag{15.93}$$

where $\mathbf{s}$ is a classical vector of unit length. The spins have three components, in contrast to the spins in the Ising model which only have one component, and the spins in the planar model which have two components.

We will consider the two-dimensional Heisenberg model for which the spins are located on a two-dimensional lattice. Early simulations and approximate theories led researchers to believe that there was a continuous phase transition, similar to that found in the Ising model. The Heisenberg model received more interest after it was related to quark confinement. Lattice models of the interaction between quarks, called lattice gauge theories, predict that the confinement of quarks could be explained if there are no phase transitions in these models. (The lack of a phase transition in these models implies that the attraction between quarks grows with distance.) The two-dimensional Heisenberg model is an analog of the four-dimensional models used to model quark-quark interactions. Shenker and Tobochnik used a combination of Monte Carlo and renormalization group methods to show that this model does not have a phase transition. Subsequent work on lattice gauge theories showed similar behavior.

a. Modify your Ising model program to simulate the Heisenberg model in two dimensions. One way to do so is to define three arrays, one for each of the three components of the unit spin vectors. A trial Monte Carlo move consists of randomly changing the direction of a spin, $\mathbf{s}_i$. First compute a small vector $\Delta\mathbf{s} = \Delta s_{\max}(q_1, q_2, q_3)$, where $-1 \leq q_n \leq 1$ is a uniform random number, and $\Delta s_{\max}$ is the maximum change of any spin component. If $|\Delta\mathbf{s}| > \Delta s_{\max}$, than compute another $\Delta\mathbf{s}$. This latter step is necessary to insure that the change in a spin direction is symmetrically distributed around the current spin direction. Then let the trial spin equal $\mathbf{s}_i + \Delta\mathbf{s}$ normalized to a unit vector. The standard Metropolis algorithm can now be used to determine if the trial spin is accepted. Compute the mean energy, the specific heat, and the susceptibility as a function of T. Choose lattice sizes of $L = 8$, 16, 32 and larger if possible and average over at least 2000 Monte Carlo steps per spin at each temperature. Is there any evidence of a phase transition? Does the susceptibility appear to diverge at a nonzero temperature? Plot the logarithm of the susceptibility versus the inverse temperature, and determine the temperature dependence of the susceptibility in the limit of low temperatures.

b. Use the Lee-Kosterlitz analysis at the specific heat peak to determine if there is a phase transition.

Project 15.39. Domain growth kinetics

When the Ising model is quenched from a high temperature to very low temperatures, domains of the ordered low temperature phase typically grow with time as a power law, $R \sim t^{\alpha}$, where R is a measure of the average linear dimension of the domains. A simple measure of the domain size is the perimeter length of a domain which can be computed from the energy per spin, ϵ, and is given by

$$R = \frac{2}{2 + \epsilon}. \tag{15.94}$$

Equation (15.94) can be motivated by the following argument. Imagine a region of N spins made up of a domain of up spins with a perimeter size, R, embedded in a sea of down spins. The total energy of this region is $-2N + 2R$, where for each spin on the perimeter, the energy is increased by 2 because one of the neighbors of a perimeter spin will be of opposite sign. The energy per spin is $\epsilon = -2 + 2R/N$. Because N is of order R^2, we arrive at the result given in (15.94).

a. Modify your Ising model program so that the initial configuration is random, that is, a typical high temperature configuration. Write a target class to simulate a quench of the system. The input parameters should be the lattice size, the quench temperature (use 0.5 initially), the maximum time (measured in Monte Carlo steps per spin) for each quench, and the number of Monte Carlo steps between drawing the lattice. Plot $\ln\langle R\rangle$ versus $\ln t$ after each quench is finished, where t is measured from the time of the quench.

b. Choose $L = 64$ and a maximum time of 128 mcs. Averages over 10 quenches will give acceptable results. What value do you obtain for α? Repeat for other temperatures and system sizes. Does the exponent change? Run for a longer maximum time to check your results.

c. Modify your program to simulate the q-state Potts model. Consider various values of q. Do your results change? Results for large q and large system sizes are given in Grest et al.

d.* Modify your program to simulate a three-dimensional system. How should you modify (15.94)? Are your results similar?

Project 15.40. Ground state energy of the Ising spin glass

A spin glass is a magnetic system with frozen-in disorder. An example of such a system is the Ising model with the exchange constant J_{ij} between nearest neighbor spins randomly chosen to be ± 1. The disorder is said to be "frozen-in" because the set of interactions $\{J_{ij}\}$ does not change with time. Because the spins cannot arrange themselves so that every pair of spins is in its lowest energy state, the system exhibits frustration similar to the antiferromagnetic Ising model on a triangular lattice (see Problem 15.22). Is there a phase transition in the spin glass model, and if so, what is its nature? The answers to these questions are very difficult to obtain by doing simulations. One of the difficulties is that we need to do not only an average over the possible configurations of spins for a given set of $\{J_{ij}\}$, but we also need to average over different realizations of the interactions. Another difficulty is that there are many local minima in the energy (free energy at finite temperature) as a function of the configurations of spins, and it is very difficult to find the global minimum. As a result, Monte Carlo simulations typically become stuck in these local minima or metastable states. Detailed finite size scaling analyses of simulations indicate that there might be a transition in three dimensions. It is generally accepted that the transition in two dimensions is at zero temperature. In the following, we will look at some of the properties of an Ising spin glass on a square lattice at low temperatures.

a. Write a program to apply simulated annealing to an Ising spin glass using the Metropolis algorithm with the temperature fixed at each stage of the annealing schedule (see Problem 15.31a). Search for the lowest energy configuration for a fixed set of $\{J_{ij}\}$. Use at least one other annealing schedule for the same $\{J_{ij}\}$ and compare your results. Then find the ground state energy for at least ten other sets of $\{J_{ij}\}$. Use lattice sizes of $L = 5$ and $L = 10$. Discuss the nature

of the ground states you are able to find. Is there much variation in the ground state energy E_0 from one set of $\{J_{ij}\}$ to another? Theoretical calculations give an average over realizations of $\overline{E_0}/N \approx -1.4$. If you have sufficient computer resources, repeat your computations for the three-dimensional spin glass.

b. Modify your program to do simulated annealing using the demon algorithm (see Problem 15.31b). How do your results compare to those that you found in part (a)?

Project 15.41. Zero temperature dynamics of the Ising model

We have seen that various kinetic growth models (Section 13.3) and reaction-diffusion models (Section 7.8) lead to interesting and nontrivial behavior. Similar behavior can be seen in the zero temperature dynamics of the Ising model. Consider the one-dimensional Ising model with $J > 0$ and periodic boundary conditions. The initial orientation of the spins is chosen at random. We update the configurations by choosing a spin at random and computing the change in energy ΔE. If $\Delta E < 0$, then flip the spin; else if $\Delta E = 0$, flip the spin with 50% probability. The spin is not flipped if $\Delta E > 0$. This type of Monte Carlo update is known as Glauber dynamics. How does this algorithm differ from the Metropolis algorithm at $T = 0$?

a. A quantity of interest is $f(t)$, the fraction of spins that have not yet flipped at time t. As usual, the time is measured in terms of Monte Carlo steps per spin. Published results (Derrida et al.) for $N = 10^5$ indicate that $f(t)$ behaves as

$$f(t) \sim t^{-\theta}, \tag{15.95}$$

for $t \approx 3$ to $t \approx 10,000$. The exact value of θ is 0.375. Verify this result and extend your results to the one-dimensional q-state Potts model. In the latter model each site is initially given a random integer between 1 and q. A site is chosen at random and set equal to either of its two neighbors with equal probability.

b. Another interesting quantity is the probability distribution, $P_n(t)$, that n sites have not yet flipped as a function of the time t (see Das and Sen). Plot P_n versus n for two times on the same graph. Discuss the shape of the curves and their differences. Choose $L \geq 100$ and $t = 50$ and 100. Try to fit the curves to a Gaussian distribution. Because the possible values of n are bounded, fit each side of the maximum of P_n to a Gaussian with different widths. There are a number of scaling properties that can be investigated. Show that $P_{n=0}(t)$ scales approximately as t/L^2. Thus, if you compute $P_{n=0}(t)$ for a number of different times and lengths such that t/L^2 has the same value, you should obtain the same value of $P_{n=0}$.

Project 15.42. The inverse power law potential

Consider the inverse power law potential

$$V(r) = V_0 \left(\frac{\sigma}{r}\right)^n, \tag{15.96}$$

with $V_0 > 0$. One reason for the interest in potentials of this form is that thermodynamic quantities such as the mean energy E do not depend on V_0 and σ separately, but depend on a single dimensionless parameter, which is defined as (see Project 8.25)

$$\Gamma = \frac{V_0}{kT}\frac{\sigma}{a}, \tag{15.97}$$

where a is defined in three and two dimensions by $4\pi a^3\rho/3 = 1$ and $\pi a^2\rho = 1$, respectively. The length a is proportional to the mean distance between particles. A Coulomb interaction corresponds to $n = 1$, and a hard sphere system corresponds to $n \to \infty$. What phases do you expect to occur for arbitrary n?

a. Compare the qualitative features of $g(r)$ for a "soft" potential with $n = 4$ to a system of hard disks at the same density.

b. Let $n = 12$ and compute the mean energy E as a function of Γ for a three-dimensional system with $N = 16$, 32, 64, and 128. Does E depend on N? Can you extrapolate your results for the N-dependence of E to $N \to \infty$? Do you see any evidence of a fluid-solid phase transition? If so, estimate the value of Γ at which it occurs. What is the nature of the transition if it exists? What is the symmetry of the ground state?

c. Let $n = 4$ and determine the symmetry of the ground state. For this value of n, there is a solid-to-solid phase transition at which the solid changes symmetry. To determine the value of Γ at which this phase transition exists and the symmetry of the smaller Γ solid phase (see Dubin and Dewitt), it is necessary to use a Monte Carlo method in which the shape of the simulation cell changes to accomodate the different symmetry (the Rahman-Parrinello method), an interesting project. An alternative is to prepare a bcc lattice at $\Gamma =\approx 105$ (for example, $T = 0.06$ and $\rho = 0.95$. Then instantaneously change the potential from $n = 4$ to $n = 12$; the new value of Γ is ≈ 4180, and the new stable phase is fcc. The transition can be observed by watching the evolution of $g(r)$.

Project 15.43. Rare gas clusters

There has been much recent interest in structures that contain many particles, but that are not macroscopic. An example is the unusual structure of sixty carbon atoms known as a "buckeyball." A less unusual structure is a cluster of argon atoms. Questions of interest include the structure of the clusters, the existence of "magic" numbers of particles for which the cluster is particularly stable, the temperature dependence of the quantities, and the possibility of different phases. This latter question has been subject to some controversy, because transitions between different kinds of behavior in finite systems are not well defined as they are for infinite systems.

a. Write a Monte Carlo program to simulate a three-dimensional system of particles interacting via the Lennard-Jones potential. Use open boundary conditions, that is, do not enclose the system in a box. The number of particles N and the temperature T should be input parameters.

b. Find the ground state energy E_0 as a function of N. For each value of N begin with a random initial configuration and accept any trial displacement that lowers the energy. Repeat for at least ten different initial configurations. Plot E_0/N versus N for $N = 2$ to 20 and describe the qualitative dependence of E_0/N on N. Is there any evidence of magic numbers, that is, value(s) of N for which E_0/N is a minimum? For each value of N save the final configuration. Plot the positions of the atoms. Does the cluster look like a part of a crystalline solid?

c. Repeat part (b) using simulated annealing. The initial temperature should be sufficiently low so that the particles do not move far away from each other. Slowly lower the temperature according to some annealing schedule. Are your results for E_0/N lower than those you obtained in part (b)?

d. To gain more insight into the structure of the clusters, compute the mean number of neighbors per particle for each value of N. What is a reasonable criteria for two particles to be neighbors? Also compute the mean distance between each pair of particles. Plot both quantities as a function of N, and compare their dependence on N with your plot of E_0/N.

e. Do you find any evidence for a "melting" transition? Begin with the configuration that has the minimum value of E_0/N and slowly increase the temperature T. Compute the energy per particle and the mean square displacement of the particles from their initial positions. Plot your results for these quantities versus T.

Project 15.44. The hard disks fluid-solid transition

Although we have mentioned (see Section 15.10) that there is much evidence for a fluid-solid transition in a hard disk system, the nature of the transition still is a problem of current research. In this project we follow the work of Lee and Strandburg and apply the constant pressure Monte Carlo method (see Section 15.12) and the Lee-Kosterlitz method (see Section 15.11) to investigate the nature of the transition. Consider $N = L^2$ hard disks of diameter $\sigma = 1$ in a two-dimensional box of volume $V = \sqrt{3}L^2v/2$ with periodic boundary conditions. The quantity $v \geq 1$ is the reduced volume and is related to the density ρ by $\rho = N/V = 2/(\sqrt{3}v)$; $v = 1$ corresponds to maximum packing. The aspect ratio of $2/\sqrt{3}$ is used to match the perfect triangular lattice. Do a constant pressure (actually constant $p^* = P/kT$) Monte Carlo simulation. The trial displacement of each disk is implemented as discussed in Section 15.10. Lee and Strandburg find that a maximum displacement of 0.09 gives a 45% acceptance probability. The other type of move is a random isotropic change of the volume of the system. If the change of the volume leads to an overlap of the disks, the change is rejected. Otherwise, if the trial volume $\tilde{V}$ is less than the current volume V, the change is accepted. A larger trial volume is accepted with probability

$$e^{-p^*(\tilde{V}-V)+N\ln\tilde{V}/V)}. \tag{15.98}$$

Volume changes are attempted 40–200 times for each set of individual disk moves. The quantity of interest is $N(v)$, the distribution of the reduced volume v. Because we need to store information about $N(v)$ in an array, it is convenient to discretize the volume in advance and choose the mesh size so that the acceptance probability for changing the volume by one unit is 40–50%. Do a Monte Carlo simulation of the hard disk system for $L = 10$ ($N = 100$) and $p^* = 7.30$. Published results are for 10^7 Monte Carlo steps. To apply the Lee-Kosterlitz method, smooth $\ln N(v)$ by fitting it to an eighth-order polynomial. Then extrapolate $\ln N(v)$ using the histogram method to determine $p_c^*(L = 10)$, the pressure at which the two peaks of $N(v)$ are of equal height. What is the value of the free energy barrier ΔF? If sufficient computer resources are available, compute ΔF for larger L (published results are for $L = 10$, 12, 14, 16, and 20) and determine if ΔF depends on L. Can you reach any conclusions about the nature of the transition?

Project 15.45. Vacancy mediated dynamics in binary alloys

When a binary alloy is rapidly quenched from a high temperature to a low temperature unstable state, a pattern of domain formation called *spinodal decomposition* takes place as the two metals in the alloy separate. This process is of much interest experimentally. Lifshitz and Slyozov have predicted that at long times, the linear domain size increases with time as $R \sim t^{1/3}$. This result

is independent of the dimension for $d \geq 2$, and has been verified experimentally and in computer simulations. The behavior is modified for binary fluids due to hydrodynamic effects.

Most of the computer simulations of this growth process have been based on the Ising model with spin exchange dynamics. In this model there is an A or B atom (spin up or spin down) at each site, where A and B represent different metals. The energy of interaction between atoms on two neighboring sites is $-J$ if the two atoms are the same type and $+J$ if they are different. Monte Carlo moves are made by exchanging unlike atoms. (The number of A and B atoms must be conserved.) A typical simulation begins with an equilibrated system at high temperatures. Then the temperature is changed instantaneously to a low temperature below the critical temperature T_c. If there are equal numbers of A and B atoms on the lattice, then spinodal decomposition occurs. If you watch a visualization of the evolution of the system, you will see wavy-like domains of each type of atom thickening with time.

The growth of the domains is very slow if we use spin exchange dynamics. We will see that if simulations are performed with vacancy mediated dynamics, the scaling behavior begins at much earlier times. Because of the large energy barriers that prevent real metallic atoms from exchanging position, it is likely that spinodal decomposition in real alloys also occurs with vacancy mediated dynamics. We can do a realistic simulation by including just one vacancy because the number of vacancies in a real alloy also is very small. In this case the only possible Monte Carlo move on a square lattice is to exchange the vacancy with one of its four neighboring atoms. To implement this algorithm, you will need an array to keep track of which type of atom is at each lattice site and variables to keep track of the location of the single vacancy. The simulation will run very fast because there is little bookkeeping and all the possible trial moves are potentially good ones. In contrast, in standard spin exchange dynamics, it is necessary to either waste computer time checking for unlike nearest-neighbor atoms or keep track of where they are.

The major quantity of interest is the growth of the domain size R. One way to determine R is to measure the pair correlation function, $C(r) = \langle s_i s_j \rangle$, where $r = |\mathbf{r}_i - \mathbf{r}_j|$, and $s_i = 1$ for an A atom and $s_i = -1$ for a B atom. The first zero in $C(r)$ is a measure of the domain size. An alternative measure of the domain size is the quantity $R = 2/(\langle E \rangle/N + 2)$, where $\langle E \rangle/N$ is the average energy per site and N is the number of sites (see Project 15.39). The quantity R is a rough measure of the length of the perimeter of a domain and is proportional to the domain size.

a. Write a program to simulate vacancy mediated dynamics. The initial state consists of the random placement of A and B atoms (half of the sites have A and half B atoms); one vacancy replaces one of the atoms. Explain why this configuration corresponds to infinite temperature. Choose a square lattice with $L \geq 50$.

b. Instantaneously quench the system by running the Metropolis algorithm at a temperature of $T = T_c/2 \approx 1.13$. You should first look at the lattice after every attempted move of the vacancy to see the effect of vacancy dynamics. After you are satisfied that your program is working correctly and that you understand the algorithm, speed up the simulation by only collecting data and showing the lattice at times equal to $t = 2^n$ where $n = 1, 2, 3 \ldots$ Measure the domain size using either the energy or $C(r)$ as a function of time averaged over many different initial configurations quenches.

c. At what time does the $\log R$ versus $\log t$ plot become linear? Do both measures of the domain size give the same results? Does the behavior change for different quench temperatures? Try

$0.2T_c$ and $0.7T_c$. A log-log plot of the domain size versus time should give the exponent 1/3.

d. Repeat the measurements in three dimensions. Do you obtain the same exponent?

Project 15.46. Heat flow using the demon algorithm

In our applications of the demon algorithm one demon shared its energy equally with all the spins. As a result the spins all attained the same mean energy of interaction. Many interesting questions arise when the system is not spatially uniform and is in a nonequilibrium but time-independent (steady) state.

Let us consider heat flow in a one-dimensional Ising model. Suppose that instead of all the sites sharing energy with one demon, each site has its own demon. We can study the flow of heat by requiring the demons at the boundary spins to satisfy different conditions than the demons at the other spins. The demon at spin 0 adds energy to the system by flipping this spin so that it is in its highest energy state, that is, in the opposite direction of spin 1. The demon at spin $N-1$ removes energy from the system by flipping spin $N-1$ so that it is in its lowest energy state, that is, in the same direction as spin $N-2$. As a result, energy flows from site 0 to site $N-1$ via the demons associated with the intermediate sites. In order that energy not build up at the "hot" end of the Ising chain, we require that spin 0 can only add energy to the system if spin $N-1$ simultaneously removes energy from the system. Because the demons at the two ends of the lattice satisfy different conditions than the other demons, we do not use periodic boundary conditions.

The temperature is determined by the generalization of the relation (15.10), that is, the temperature at site i is related to the mean energy of the demon at site i. To control the temperature gradient, we can update the end spins at a different rate than the other spins. The maximum temperature gradient occurs if we update the end spins after every update of an internal spin. A smaller temperature gradient occurs if we update the end spins less frequently. The temperature gradient between any two spins can be determined from the temperature profile, the spatial dependence of the temperature. The energy flow can be determined by computing the magnitude of the energy per unit time that enters the lattice at site 0.

To implement this procedure we modify `IsingDemon` by converting the variables `demonEnergy` and `demonEnergyAccumulator` to arrays. We do the usual updating procedure for spins 1 through $N-2$ and visit spins 0 and $N-1$ at regular intervals denoted by `timeToAddEnergy`. The class `ManyDemons` can be downloaded from the ch15 directory.

a. Write a target class that inputs the number of spins, N, and the initial energy of the system, outputs the number of Monte Carlo steps per spin and the energy added to the system at the high temperature boundary, and plots the temperature as a function of position.

b. As a check on `ManyDemons`, modify the class so that all the demons are equivalent, that is, impose periodic boundary conditions and do not use method `boundarySpins`. Compute the mean energy of the demon at each site and use (15.10) to define a local site temperature. Use $N \geq 52$ and run for about 10000 mcs. Is the local temperature approximately uniform? How do your results compare with the single demon case?

c. In `ManyDemons` the energy is added to the system at site 0 and is removed at site $N-1$. Determine the mean demon energy for each site and obtain the corresponding local temperature and the

mean energy of the system. Draw the temperature profile by plotting the temperature as a function of site number. The temperature gradient is the difference in temperature from site $N-2$ to site 1 divided by the distance between them. (The distance between neighboring sites is unity.) Because of local temperature fluctuations and edge effects, the temperature gradient should be estimated by fitting the temperature profile in the middle of the lattice to a straight line. Reasonable choices for the parameters are $N = 52$ and `timeToAddEnergy` $= 1$. Run for at least 10000 mcs.

d. The heat flux Q is the energy flow per unit length per unit time. The energy flow is the amount of energy that demon 0 adds to the system at site 0. The time is conveniently measured in terms of Monte Carlo steps per spin. Determine Q for the parameters used in part (c).

e. If the temperature gradient $\partial T/\partial x$ is not too large, the heat flux Q is proportional to $\partial T/\partial x$. We can determine the *thermal conductivity* κ by the relation

$$Q = -\kappa \frac{\partial T}{\partial x}. \tag{15.99}$$

Use your results for $\partial T/\partial x$ and Q to estimate κ.

f. Determine Q, the temperature profile, and the mean temperature for different values of `timeToAddEnergy`. Is the temperature profile linear for all values of `timeToAddEnergy`? If the temperature profile is linear, estimate $\partial T/\partial x$ and determine κ. Does κ depend on the mean temperature?

Note that by using many demons we were able to compute a temperature profile by using an algorithm that manipulates only integer numbers. The conventional approach is to solve a heat equation similar in form to the diffusion equation. Now we use the same idea to compute the magnetization profile when the end spins of the lattice are fixed.

g. Modify `ManyDemons` by not calling `method boundarySpins`. Also, constrain spins 0 and $N-1$ to be $+1$ and -1 respectively. Estimate the magnetization profile by plotting the mean value of the spin at each site versus the site number. Choose $N = 22$ and `mcs` ≥ 1000. How do your results vary as you increase N?

h. Compute the mean demon energy and hence the local temperature at each site. Does the system have a uniform temperature even though the magnetization is not uniform? Is the system in thermal equilibrium?

i. The effect of the constraint on the end spins is easier to observe in two and three dimensions than in one dimension. Write a program for a two-dimensional Ising model on a $L \times L$ square lattice. Constrain the spins at site (i, j) to be $+1$ and -1 for $i = 0$ and $i = L-1$ respectively. Use periodic boundary conditions in the y direction. How do your results compare with the one-dimensional case?

j. Remove the periodic boundary condition in the y direction and constrain all the boundary spins from $i = 0$ to $(L/2) - 1$ to be $+1$ and the other boundary spins to be -1. Choose an initial configuration where all the spins on the left half of the system are $+1$ and the others are -1. Do

the simulation and draw a configuration of the spins once the system has reached equilibrium. Draw a line between each pair of spins of opposite sign. Describe the curve separating the $+1$ spins from the -1 spins. Begin with $L = 20$ and determine what happens as L is increased.

Appendix 15A: Relation of the Mean Demon Energy to the Temperature

We know that the energy of the demon, E_d, is constrained to be positive and that the probability for the demon to have energy E_d is proportional to $e^{-E_d/kT}$. Hence in general, $\langle E_d \rangle$ is given by

$$\langle E_d \rangle = \frac{\sum_{E_d} E_d \, e^{-E_d/kT}}{\sum_{E_d} e^{-E_d/kT}}, \tag{15.100}$$

where the summations in (15.100) are over the possible values of E_d. If an Ising spin is flipped in zero magnetic field, the minimum nonzero decrease in energy of the system is $4J$ (see Figure 15.11). Hence the possible energies of the demon are 0, $4J$, $8J$, $12J$, ... We write $x = 4J/kT$ and perform the summations in (15.100). The result is

$$\langle E_d/kT \rangle = \frac{0 + xe^{-x} + 2xe^{-2x} + \ldots}{1 + e^{-x} + e^{-2x} + \ldots} = \frac{x}{e^x - 1}. \tag{15.101}$$

The form (15.10) can be obtained by solving (15.101) for T in terms of E_d. Convince yourself that the relation (15.101) is independent of dimension for lattices with an even number of nearest neighbors.

If the magnetic field is nonzero, the possible values of the demon energy are $0, 2H, 4J - 2H, 4J + 2H, \cdots$. If J is a multiple of H, then the result is the same as before with $4J$ replaced by $2H$, because the possible energy values for the demon are multiples of $2H$. If the ratio $4J/2H$ is irrational, then the demon can take on a continuum of values, and thus $\langle E_d \rangle = kT$. The other possibility is that $4J/2H = m/n$, where m and n are prime positive integers that have no common factors (other than 1). In this case it can be shown that (see Mak)

$$kT/J = \frac{4/m}{\ln(1 + 4J/m\langle E_d \rangle)}. \tag{15.102}$$

Surprisingly, (15.102) does not depend on n. Test these relations for $H \neq 0$ by choosing values of J and H and computing the sums in (15.100) directly.

Appendix 15B: Fluctuations in the Canonical Ensemble

We first obtain the relation of the constant volume heat capacity C_V to the energy fluctuations in the canonical ensemble. We write C_V as

$$C_V = \frac{\partial \langle E \rangle}{\partial T} = -\frac{1}{kT^2} \frac{\partial \langle E \rangle}{\partial \beta}. \tag{15.103}$$

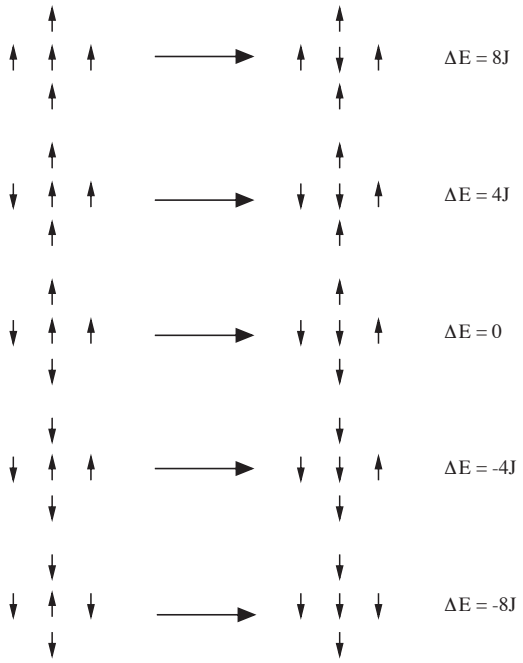

Figure 15.11: The five possible transitions of the Ising model on the square lattice with spin flip dynamics.

From (15.11) we have

$$\langle E\rangle = -\frac{\partial}{\partial\beta}\ln Z, \tag{15.104}$$

and

$$\frac{\partial\langle E\rangle}{\partial\beta} = -\frac{1}{Z^2}\frac{\partial Z}{\partial\beta}\sum_s E_s\, e^{-\beta E_s} - \frac{1}{Z}\sum_s E_s^2\, e^{-\beta E_s} \tag{15.105}$$

$$= \langle E\rangle^2 - \langle E^2\rangle. \tag{15.106}$$

The relation (15.19) follows from (15.103) and (15.106). Note that the heat capacity is at constant volume because the partial derivatives were performed with the energy levels E_s kept constant. The corresponding quantity for a magnetic system is the heat capacity at constant external magnetic field.

The relation of the magnetic susceptibility χ to the fluctuations of the magnetization M can be obtained in a similar way. We assume that the energy can be written as

$$E_s = E_{0,s} - HM_s, \tag{15.107}$$

where $E_{0,s}$ is the energy of interaction of the spins in the absence of a magnetic field, H is the external applied field, and M_s is the magnetization in the s state. The mean magnetization is

# spins up	$g(E,M)$	Energy	Magnetization
4	1	−8	4
3	4	0	2
2	4	0	0
2	2	8	0
1	4	0	−2
0	1	−8	−4

Table 15.2: The energy and magnetization of the 2^4 states of the zero field Ising model on the 2×2 square lattice. The quantity $g(E,M)$ is the number of microstates with the same energy.

given by

$$\langle M\rangle = \frac{1}{Z}\sum M_s\, e^{-\beta E_s}. \tag{15.108}$$

Because $\partial E_s/\partial H = -M_s$, we have

$$\frac{\partial Z}{\partial H} = \sum_s \beta M_s\, e^{-\beta E_s}. \tag{15.109}$$

Hence we obtain

$$\langle M\rangle = \frac{1}{\beta}\frac{\partial}{\partial H}\ln Z. \tag{15.110}$$

If we use (15.108) and (15.110), we find

$$\frac{\partial\langle M\rangle}{\partial H} = -\frac{1}{Z^2}\frac{\partial Z}{\partial H}\sum_s M_s\, e^{-\beta E_s} + \frac{1}{Z}\sum_s \beta M_s^2\, e^{-\beta E_s} \tag{15.111}$$

$$= -\beta\langle M\rangle^2 + \beta\langle M^2\rangle. \tag{15.112}$$

The relation (15.21) for the zero field susceptibility follows from (15.112) and the definition (15.20).

Appendix 15C: Exact Enumeration of the 2×2 Ising Model

Because the number of possible states or configurations of the Ising model increases as 2^N, we can enumerate the possible configurations only for small N. As an example, we calculate the various quantities of interest for a 2×2 Ising model on the square lattice with periodic boundary conditions. In Table 15.2 we group the sixteen states according to their total energy and magnetization.

We can compute all the quantities of interest using Table 15.2. The partition function is given by

$$Z = 2\,e^{8\beta J} + 12 + 2\,e^{-8\beta J}. \tag{15.113}$$

If we use (15.104) and (15.113), we find

$$\langle E\rangle = -\frac{\partial}{\partial\beta}\ln Z = -\frac{1}{Z}\big[2(8)e^{8\beta J} + 2(-8)e^{-8\beta J}\big]. \tag{15.114}$$

Because the other quantities of interest can be found in a similar manner, we only give the results:

$$\langle E^2 \rangle = \frac{1}{Z}\left[(2 \times 64)\, e^{8\beta J} + (2 \times 64)\, e^{-8\beta J}\right] \tag{15.115}$$

$$\langle M \rangle = \frac{1}{Z}(0) = 0 \tag{15.116}$$

$$\langle |M| \rangle = \frac{1}{Z}\left[(2 \times 4)\, e^{8\beta J} + 8 \times 2\right] \tag{15.117}$$

$$\langle M^2 \rangle = \frac{1}{Z}\left[(2 \times 16)\, e^{8\beta J} + 8 \times 4\right]. \tag{15.118}$$

The dependence of C and χ on βJ can be found by using (15.114) and (15.115) and (15.116) and (15.118) respectively.

References and Suggestions for Further Reading

M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, Clarendon Press (1987). See Chapter 4 for a discussion of Monte Carlo methods.

Paul D. Beale, "Exact distribution of energies in the two-dimensional Ising model," Phys. Rev. Lett. **76**, 78 (1996). The author discusses a Mathematica program that can compute the exact density of states for the two-dimensional Ising model.

K. Binder, editor, *Monte Carlo Methods in Statistical Physics*, second edition, Springer-Verlag (1986). Also see K. Binder, editor, *Applications of the Monte Carlo Method in Statistical Physics*, Springer-Verlag (1984) and K. Binder, editor, *The Monte Carlo Method in Condensed Matter Physics*, Springer-Verlag (1992). The latter book discusses the Binder cumulant method in the introductory chapter.

Marvin Bishop and C. Bruin, "The pair correlation function: a probe of molecular order," Am. J. Phys. **52**, 1106–1108 (1984). The authors compute the pair correlation function for a two-dimensional Lennard-Jones model.

A. B. Bortz, M. H. Kalos and J. L. Lebowitz, "A new algorithm for Monte Carlo simulation of Ising spin systems," J. Comput. Phys. **17**, 10–18 (1975). This paper first introduced the n-fold way algorithm, which was rediscovered independently by many workers in the 1970s and 80s.

S. G. Brush, "History of the Lenz-Ising model," Rev. Mod. Phys. **39**, 883–893 (1967).

James B. Cole, "The statistical mechanics of image recovery and pattern recognition,"Am. J. Phys. **59**, 839–842 (1991). A discussion of the application of simulated annealing to the recovery of images from noisy data.

R. Cordery, S. Sarker, and J. Tobochnik, "Physics of the dynamical critical exponent in one dimension," Phys. Rev. B **24**, 5402–5403 (1981).

Michael Creutz, "Microcanonical Monte Carlo simulation," Phys. Rev. Lett. **50**, 1411 (1983). See also Gyan Bhanot, Michael Creutz, and Herbert Neuberger, "Microcanonical simulation of Ising systems," Nuc. Phys. B **235**, 417–434 (1984).

Pratap Kumar Das and Parongama Sen, "Probability distributions of persistent spins in an Ising chain," J. Phys. A **37**, 7179–7184 (2004).

B. Derrida, A. J. Bray, and C. Godrèche, "Non-trivial exponents in the zero temperature dynamics of the 1D Ising and Potts models," J. Phys. A **27**, L357–L361 (1994); B. Derrida, V. Hakim, and V. Pasquier, "Exact first passage exponents in 1d domain growth: Relation to a reaction-diffusion model," Phys. Rev. Lett. **75**, 751 (1995).

Daniel H. E. Dubin and Hugh Dewitt, "Polymorphic phase transition for inverse-power-potential crystals keeping the first-order anharmonic correction to the free energy," Phys. Rev. B **49**, 3043–3048 (1994).

Jerome J. Erpenbeck and Marshall Luban, "Equation of state for the classical hard-disk fluid," Phys. Rev. A **32**, 2920–2922 (1985). These workers use a combined molecular dynamics/-Monte Carlo method and consider 1512 and 5822 disks.

Alan M. Ferrenberg, D. P. Landau, and Y. Joanna Wong, "Monte Carlo simulations: hidden errors from "good" random number generators," Phys. Rev. Lett. **69**, 3382 (1992).

Alan M. Ferrenberg and Robert H. Swendsen, "New Monte Carlo technique for studying phase transitions," Phys. Rev. Lett. **61**, 2635 (1988); "Optimized Monte Carlo data analysis," Phys. Rev. Lett. **63**, 1195 (1989); "Optimized Monte Carlo data analysis," Computers in Physics **3** 5, 101 (1989). The second and third papers discuss using the multiple histogram method with data from simulations at more than one temperature.

P. Fratzl and O. Penrose, "Kinetics of spinodal decomposition in the Ising model with vacancy diffusion," Phys. Rev. B **50**, 3477–3480 (1994).

Daan Frenkel and Berend Smit, *Understanding Molecular Simulation*, second edition, Academic Press (2002).

Harvey Gould and W. Klein, "Spinodal effects in systems with long-range interactions," Physica D **66**, 61–70 (1993). This paper discusses nucleation in the Ising model and Lennard-Jones systems.

Harvey Gould and Jan Tobochnik, "Overcoming critical slowing down," Computers in Physics **3** (4), 82 (1989).

James E. Gubernatis, *The Monte Carlo Method in the Physical Sciences*, AIP Press (2004). June 2003 was the 50th anniversary of the Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller publication of what is now called the Metropolis algorithm. This algorithm established the Monte Carlo method in physics and other fields and lead to the development of other Monte Carlo algorithms. Six of the papers in the proceedings of the conference give historical perspectives.

Hong Guo, Martin Zuckermann, R. Harris, and Martin Grant, "A fast algorithm for simulated annealing," Physica Scripta **T38**, 40–44 (1991).

Gary S. Grest, Michael P. Anderson, and David J. Srolovitz, "Domain-growth kinetics for the Q-state Potts model in two and three dimensions," Phys. Rev. B **38**, 4752–4760 (1988).

R. Harris, "Demons at work," Computers in Physics **4** (3), 314 (1990).

S. Istrail, "Statistical mechanics, three-dimensionality and NP-completeness: I. Universality of intractability of the partition functions of the Ising model across non-planar lattices," Proceedings of the 32nd ACM Symposium on the Theory of Computing, ACM Press, pp. 87–96, Portland, Oregon, May 21–23, 2000. This paper shows that it is impossible to obtain an analytical solution for the three-dimensional Ising model.

J. Kertész, J. Cserti and J. Szép, "Monte Carlo simulation programs for microcomputer," Eur. J. Phys. **6**, 232–237 (1985).

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," Science **220**, 671–680 (1983). See also, S. Kirkpatrick and G. Toulouse, "Configuration space analysis of traveling salesman problems," J. Physique **46**, 1277–1292 (1985).

J. M. Kosterlitz and D. J. Thouless, "Ordering, metastability and phase transitions in two-dimensional systems," J. Phys. C **6**, 1181–1203 (1973); J. M. Kosterlitz, "The critical properties of the two-dimensional xy model," J. Phys. C **7**, 1046–1060 (1974).

D. P. Landau, Shan-Ho Tsai, and M. Exler, "A new approach to Monte Carlo simulations in statistical physics: Wang-Landau sampling, Am. J. Phys. **72**, 1294–1302 (2004).

D. P. Landau, "Finite-size behavior of the Ising square lattice," Phys. Rev. B **13**, 2997–3011 (1976). A clearly written paper on a finite-size scaling analysis of Monte Carlo data. See also D. P. Landau, "Finite-size behavior of the simple-cubic Ising lattice," Phys. Rev. B **14**, 255–262 (1976).

D. P. Landau and R. Alben, "Monte Carlo calculations as an aid in teaching statistical mechanics," Am. J. Phys. **41**, 394–400 (1973).

David Landau and Kurt Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, second edition, Cambridge University Press (2005).

D. P. Landau, Shan-Ho Tsai, and M. Exler, "A new approach to Monte Carlo simulations in statistical physics: Wang-Landau sampling," Am. J. Phys. **72**, 1294–1302 (2005).

Jooyoung Lee and J. M. Kosterlitz, "New numerical method to study phase transitions," Phys. Rev. Lett. **65**, 137 (1990); *ibid.*, "Finite-size scaling and Monte Carlo simulations of first-order phase transitions," Phys. Rev. B **43**, 3265–3277 (1991).

Jooyoung Lee and Katherine J. Strandburg, "First-order melting transition of the hard-disk system," Phys. Rev. B **46**, 11190–11193 (1992).

Jiwen Liu and Erik Luijten, "Rejection-free geometric cluster algorithm for complex fluids," Phys. Rev. Lett. **92** 035504 (2004) and ibid., Phys. Rev. E **71**, 066701-1–12 (2005).

J. Machta, Y. S. Choi, A. Lucke, T. Schweizer, and L. Chayes, "Invaded cluster algorithm for Potts models," Phys. Rev. E 54, 1332–1345 (1996).

S. S. Mak, "The analytical demon of the Ising model," Phys. Lett. A **196**, 318 (1995).

J. Marro and R. Toral, "Microscopic observations on a kinetic Ising model," Am. J. Phys. **54**, 1114–1121 (1986).

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Keller, "Equation of state calculations for fast computing machines," J. Chem. Phys. **21**, 1087–1092 (1953).

A. Alan Middleton, "Improved extremal optimization for the Ising spin glass," Phys. Rev. E **69**, 055701-1–4 (2004). The extremal optimization algorithm, which was inspired by the Bak-Sneppen algorithm for evolution (see Problem 14.12), preferentially flips spins that are "unfit." The adaptive algorithm proposed in this paper is an example of an heuristic that finds exact ground states efficiently for systems with frozen-in disorder.

M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics*, Oxford University Press (1999).

M. A. Novotny, "A new approach to an old algorithm for the simulation of Ising-like systems," Computers in Physics **9** (1), 46 (1995). The n-fold way algorithm is discussed. Also, see M. A. Novotny, "A tutorial on advanced dynamic Monte Carlo methods for systems with discrete state spaces," in *Annual Reviews of Computational Physics IX*, edited by Dietrich Stauffer, World Scientific Publishing Company (2001), pp. 153–210.

Ole G. Mouritsen, *Computer Studies of Phase Transitions and Critical Phenomena*, Springer-Verlag (1984).

E. P. Münger and M. A. Novotny, "Reweighting in Monte Carlo and Monte Carlo renormalization-group studies," Phys. Rev. B **43**, 5773–5783 (1991). The authors discuss the histogram method and combine it with renormalization group calculations.

Michael Plischke and Birger Bergersen, *Equilibrium Statistical Physics*, third edition, Prentice Hall (2005). A graduate level text that discusses some contemporary topics in statistical physics, many of which have been influenced by computer simulations.

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery, *Numerical Recipes*, second edition, Cambridge University Press (1992). A Fortran program for the traveling salesman problem is given in Section 10.9.

Stephen H. Shenker and Jan Tobochnik, "Monte Carlo renormalization-group analysis of the classical Heisenberg model in two dimensions," Phys. Rev. B **22**, 4462–472 (1980).

Amihai Silverman and Joan Adler, "Animated simulated annealing," Computers in Physics **6**, 277 (1992). The authors describe a simulation of the annealing process to obtain a defect free single crystal of a model material.

H. Eugene Stanley, *Introduction to Phase Transitions and Critical Phenomena*, Oxford University Press (1971). See Appendix B for the exact solution of the zero-field Ising model for a two-dimensional lattice.

Jan Tobochnik and G. V. Chester, "Monte Carlo study of the planar model," Phys. Rev. B **20**, 3761–3769 (1979).

Jan Tobochnik, Harvey Gould, and Jon Machta, "Understanding the temperature and the chemical potential through computer simulations," Am. J. Phys. **73** (8), 708–716 (2005). This paper extends the demon algorithm to compute the chemical potential.

Simon Trebst, David A. Huse, and Matthias Troyer, "Optimizing the ensemble for equilibration in broad-histogram Monte Carlo simulations," Phys. Rev. E **70**, 046701-1–5 (2004). The adaptive algorithm presented in this paper overcomes critical slowing down and improves upon the Wang-Landau algorithm and is another example of the flexibility of Monte Carlo algorithms.

I. Vattulainen, T. Ala-Nissila, and K. Kankaala, "Physical tests for random numbers in simulations," Phys. Rev. Lett. **73**, 2513 (1994).

B. Widom, "Some topics in the theory of fluids," J. Chem. Phys. **39**, 2808–2812 (1963). This paper discusses the insertion method for calculating the chemical potential.