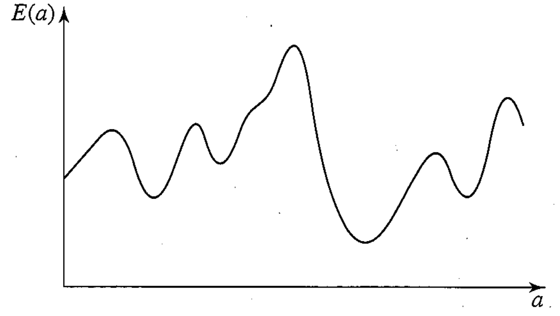

Figure 15.8 Plot of the function $E(a)$ as a function of the parameter a.

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, 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 can also 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