a function of functions) that is proportional to ϵ^2 or a higher power of ϵ. An important extremum principle in classical mechanics is based on the action S:

$$S = \int_{t_0}^{t_{\rm final}} L\, dt, \tag{7.64}$$

where t_0 and $t_{\rm final}$ are the initial and final times, respectively. The Lagrangian L in (7.64) is the kinetic energy minus the potential energy. The extremum principle for the action is known as *the principle of least action* or Hamilton's action principle. The path where (7.64) is stationary (either a minimum or a saddle point) satisfies Newton's second law (for conservative forces). One reason for the importance of the principle of least action is that quantum mechanics can be formulated in terms of an integral over the action (see Section 16.10).

To use (7.64) to find the motion of a single particle in one dimension, we fix the position at the chosen initial and final times, $x(t_0)$ and $x(t_{\rm final})$, and then choose the velocities and positions for the intermediate times $t_0 < t < t_{\rm final}$ to minimize the action. One way to implement this procedure numerically is to convert the integral in (7.64) to a sum:

$$S \approx \sum_{i=1}^{N-1} L(t_i)\,\Delta t, \tag{7.65}$$

where $t_i = t_0 + i\,\Delta t$. (The approximation used to obtain (7.65) is known as the rectangular approximation and is discussed in Chapter 11.) For a single particle in one dimension moving in an external potential $u(x)$, we can write

$$L_i \approx \frac{m}{2(\Delta t)^2}(x_{i+1} - x_i)^2 - u(x_i), \tag{7.66}$$

where m is the mass of the particle, and $u(x_i)$ is the potential energy of the particle at x_i. The velocity has been approximated as the difference in position divided by the change in time Δt.

Problem 7.39 Principle of least action

(a) Write a program to minimize the action S given in (7.64) for the motion of a single particle in one dimension. Use the approximate form of the Lagrangian given in (7.66). One way to write the program is to modify class `Fermat` so that the vertical coordinate for the light ray becomes the position of the particle, and the horizontal region number `i` becomes the discrete time interval of duration Δt.

(b) Verify your program for the case of free fall for which the potential energy is $u(y) = mgy$. Choose $y(t=0) = 2$ m and $y(t = 10\text{ s}) = 8$ m and begin with $N = 20$. Allow the maximum change in the position to be 5 m.

(c) Consider the harmonic potential $u(x) = \frac{1}{2}kx^2$. What shape do you expect the path $x(t)$ to be? Increase N to approximately 50 and estimate the path by minimizing the action. ■

It is possible to extend the principle of least action to more dimensions or particles; however, it is necessary to begin with a path close to the optimum one to obtain a good approximation to the optimum path in a reasonable time.

In Problems 7.37–7.39, a simple Monte Carlo algorithm that always accepts paths that reduce the time or action is sufficient. However, for more complicated index of refraction distributions or potentials, it is possible that such a simple algorithm will find only a local minimum, and the global minimum will be missed. The problem of finding the global minimum is very general and is shared by all optimization algorithms if the system has many relative minima. Optimization is a very active area of research in many fields of science and engineering. Ideas from physics, biology, and computer science have led to many improved algorithms. We will discuss some of these algorithms in Chapter 15. In most of these algorithms, paths that are worse than the current path are sometimes accepted in an attempt to climb out of a local minimum. Other algorithms involve ways of sampling over a wider range of possible paths. Another approach is to convert the Monte Carlo algorithm into a deterministic algorithm. We have already mentioned that an analytical variational calculation leads to Newton's second law. Passerone and Parrinello discuss an algorithm for looking for extrema in the action by maintaining the discrete structure in (7.66) and then finding the extremum by taking the derivative with respect to each coordinate x_i and setting the resulting equations equal to zero. This procedure leads to a set of deterministic equations that need to be solved numerically. The performance can be improved by enforcing energy conservation and using some other tricks.

7.11 ■ PROJECTS

Almost all of the problems in this chapter can be done using more efficient programs, greater number of trials, and larger systems. More applications of random walks and random number sequences are discussed in subsequent chapters. Many more ideas for projects can be gained from the references.

Project 7.40 Competition between diffusion and fragmentation

As we have discussed, random walks are useful for understanding diffusion in contexts more general than the movement of a particle. Consider a particle in solution whose mass can grow either by the absorption of particles or shrink by the loss of small particles, including fragmentation. We can model this process as a random walk by replacing the position of the particle by its mass. One difference between this case and the random walks we have studied so far is that the random variable, the mass, must be positive. The model of Ferkinghoff–Berg et al. can be summarized as follows:

(i) Begin with N objects with some distribution of lengths. Let the integer L_i represent the length of the ith object.

(ii) All the objects change their length by ± 1. This step is analogous to a random walk. If the length of an object becomes equal to 0, it is removed from the system. An easy way to eliminate the ith object is to set its length equal to the length of the last object and reduce N by unity.

(iii) Choose one object at random with a probability that is proportional to the length of the object. Fragment this object into two objects, where the fraction of the mass going to each object is random.

(iv) Repeat steps (ii) and (iii).