4

MONTE CARLO METHODS

4.1 Introduction

The Monte Carlo method was developed by von Neumann, Ulam, and Metropolis at the end of the Second World War to study the diffusion of neutrons in fissionable material. The name 'Monte Carlo', chosen because of the extensive use of random numbers in the calculation, was coined by Metropolis in 1947 and used in the title of a paper describing the early work at Los Alamos [Metropolis and Ulam 1949].

Statisticians had used model sampling experiments to investigate problems long before this time. The English statistician W. S. Gossett ('Student') [1908] estimated the correlation coefficients in his 't' distribution with the help of a sampling experiment, and Lord Kelvin's assistant generated 5000 random trajectories to study the elastic collisions of particles with shaped walls [Kelvin 1901]. The novel contribution of von Neumann and Ulam [1945] was to realize that determinate mathematical problems could be treated by finding a probabilistic analogue which is then solved by a stochastic sampling experiment.

These sampling experiments involve the generation of random numbers followed by a limited number of arithmetic and logical operations, which are often the same at each step. These are tasks that are well suited to a computer and the growth in the importance of the method can be linked to the rapid development of these machines. The arrival of the MANIAC computer at Los Alamos in 1952 prompted the study of the many-body problem by Metropolis *et al.* [1953] and the development of the Metropolis Monte Carlo method [Wood 1986], which is the subject of this chapter.

As always, there are those who cannot wait for technology. Buffon, the eminent eighteenth-century French naturalist, discovered a beautiful theorem in geometrical probability. If a needle of length l is thrown at random onto a set of equally spaced parallel lines, d apart (where $d > l$), the probability of the needle crossing a line is $2l/\pi d$. In 1901, the Italian mathematician Lazzerini performed a simulation by spinning round and dropping a needle 3407 times. He estimated π to be 3.1415929 [Pedoe 1958]. We shall use this as an example of a simple Monte Carlo integration in the next section. From this exhausting beginning the method has grown to the point where it is, arguably, 'the most powerful and commonly used technique for analysing complex problems' [Rubinstein 1981].

As outlined in Chapter 2, the Metropolis Monte Carlo method aims to generate a trajectory in phase space which samples from a chosen statistical ensemble. There are several difficulties involved in devising such a prescription and making it work for a system of molecules in a liquid. So we take care to

introduce the Monte Carlo method through some simple examples in the following sections.

4.2 Monte Carlo integration

4.2.1 Hit and miss

We can illustrate the use of the MC technique as a method of integration by returning to the evaluation of π. This can be done by finding the area of a circle of unit radius. The circle, centred at the origin and inscribed in a square, is shown in Fig. 4.1.

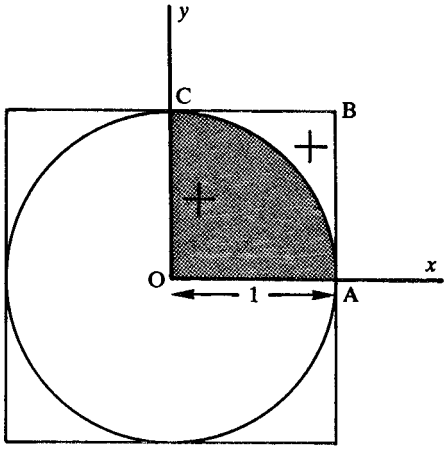

Fig. 4.1 The geometry for the hit and miss integration to find the area of the circle.

A number of trial shots are generated in the square OABC. At each trial two independent random numbers are chosen from a uniform distribution on (0, 1). These numbers are used as the coordinates of a point, (examples are marked as crosses in the figure). The distance from the random point to the origin is calculated. If this distance is less than or equal to one, the shot has landed in the shaded region and a hit is scored. If a total of τ_{shot} shots are fired and τ_{hit} hits scored then

$$\pi \approx \frac{4 \times \text{Area under the curve CA}}{\text{Area of the square OABC}} = \frac{4\tau_{\text{hit}}}{\tau_{\text{shot}}}. \tag{4.1}$$

The key to this method is the generation of $2\tau_{\text{shot}}$ random numbers from a uniform distribution. Random number generators are simple programs and their construction and performance are discussed in Appendix G.

The estimate of this area will depend on the numbers of trials; in fact the error in the estimate is $\mathcal{O}(\tau_{\text{shot}}^{-1/2})$. The results from a hit and miss experiment are

shown in Fig. 4.2; the correct value for the area of the circle is, of course, π and after 10^7 shots the MC estimate is 3.14173 correct to four figures. To calculate another decimal place would require an order of magnitude increase in the number of shots. It is straightforward to devise a similar hit-and-miss experiment to simulate Buffon's needle. In Fig. 4.2 an estimate of π obtained in this way has been included. After 10^7 shots the result is 3.140472 (only accurate to three figures), confirming that Lazzerini had a lucky afternoon.

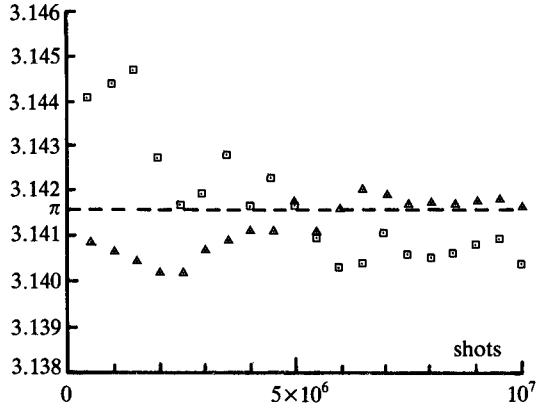

Fig. 4.2 The cumulative estimate of π as a function of the number of MC shots by hit-and-miss area of a circle (triangles) and the Buffon needle experiment (squares).

4.2.2 Sample mean integration

Hit and miss integration is conceptually easy to understand but the sample mean method is more generally applicable and offers a more accurate estimate for most integrals [Hammersley and Handscomb 1964; Rubinstein 1981]. In this case the integral of interest

$$F = \int_{x_1}^{x_2} \mathrm{d}x f(x) \tag{4.2}$$

is rewritten as

$$F = \int_{x_1}^{x_2} \mathrm{d}x \left(\frac{f(x)}{\rho(x)}\right) \rho(x) \tag{4.3}$$

where $\rho(x)$ is an arbitrary probability density function. Consider performing a number of trials τ, each consisting of choosing a random number ζ_τ from the distribution $\rho(x)$ in the range (x_1, x_2). Then

$$F = \left\langle \frac{f(\zeta_\tau)}{\rho(\zeta_\tau)} \right\rangle_{\text{trials}} \tag{4.4}$$

where the brackets represent an average over all trials. A simple application would be to choose $\rho(x)$ to be uniform, i.e.

$$\rho(x) = \frac{1}{(x_2 - x_1)} \qquad x_1 \leqslant x \leqslant x_2 \tag{4.5}$$

and then the integral F can be estimated as

$$F \approx \frac{(x_2 - x_1)}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} f(\zeta_\tau) . \tag{4.6}$$

To apply this approach to the estimation of π we consider the equation for the circle in the first quadrant, $f(x) = (1 - x^2)^{-1/2}$, with x between $x_1 = 0$ and $x_2 = 1$. In a typical experiment the estimate of π after 10^7 trials using eqn (4.6) is 3.14169.

For the simple one-dimensional integration, eqn (4.2), the MC technique is not competitive with straightforward numerical methods such as Simpson's rule (the Simpson's rule estimate of π with only 10^4 function evaluations is 3.141593). However, for the multidimensional integrals of statistical mechanics, the sample mean method, with a suitable choice of $\rho(x)$, is the only sensible solution. To understand this, we consider the evaluation of the configurational integral $Z_{NVT} = \int \mathrm{d}\mathbf{r} \exp(-\beta \mathscr{V})$, (eqn (2.26)), for a system of, say, $N = 100$ molecules in a cube of side L. Even a crude Simpson's rule integration might require 10 function evaluations for each of the 300 coordinates, so as to span the range $(-\frac{1}{2}L, \frac{1}{2}L)$. This total of 10^{300} function evaluations is quite infeasible. Moreover, the overwhelming proportion of these would give a zero result since the Boltzmann factor is extremely small (zero for hard spheres) whenever molecules overlap significantly. The sample mean approach to this integral, using a uniform distribution, might proceed as follows. A trial τ is carried out:

(a) pick a point at random in the 300-dimensional configuration space, by generating 300 random numbers, on $(-\frac{1}{2}L, \frac{1}{2}L)$, which, taken in triplets, specify the coordinates of each molecule;
(b) calculate the potential energy, $\mathscr{V}(\tau)$, and hence the Boltzmann factor for this configuration.

This procedure is repeated for many trials and the configurational integral is estimated using

$$Z_{NVT} \approx \frac{V^N}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} \exp(-\beta \mathscr{V}(\tau)) . \tag{4.7}$$

In principle, the number of trials $\tau_{\max}$ may be increased until Z_{NVT} is estimated to the desired accuracy. We would not expect to have to conduct 10^{300} function evaluations, as for Simpson's rule, but again a large number of the trials would give a very small contribution to the average. An accurate estimation of Z_{NVT} for a dense liquid using a uniform sample mean method is

beyond the capabilities of current computers, although methods of this type have been used to examine the structural properties of the hard sphere fluid at low densities [Alder, Frankel, and Lewinson 1955]. The difficulties in the calculation of Z_{NVT} apply equally to the calculation of ensemble averages such as

$$\langle \mathscr{A} \rangle_{NVT} = \frac{\int \mathrm{d}\mathbf{r} \mathscr{A} \exp(-\beta \mathscr{V})}{\int \mathrm{d}\mathbf{r} \exp(-\beta \mathscr{V})} \approx \frac{\sum_{\tau=1}^{\tau_{max}} \mathscr{A}(\tau) \exp(-\beta \mathscr{V}(\tau))}{\sum_{\tau=1}^{\tau_{max}} \exp(-\beta \mathscr{V}(\tau))}, \tag{4.8}$$

if we attempt to estimate the numerator and denominator separately by using the uniform sample mean method. However, at realistic liquid densities the problem can be solved using a sample mean integration where the random coordinates are chosen from a non-uniform distribution. This method of 'importance sampling' is discussed in the next section.

4.3 Importance sampling

Importance sampling techniques choose random numbers from a distribution $\rho(x)$, which allows the function evaluation to be concentrated in the regions of space that make important contributions to the integral. Consider the canonical ensemble. In this case the desired integral is

$$\langle \mathscr{A} \rangle_{NVT} = \int \mathrm{d}\Gamma \rho_{NVT}(\Gamma) \mathscr{A}(\Gamma)$$

i.e. the integrand is $f = \rho_{NVT} \mathscr{A}$. By sampling configurations at random, from a chosen distribution ρ we can estimate the integral as

$$\langle \mathscr{A} \rangle_{NVT} = \langle \mathscr{A} \rho_{NVT} / \rho \rangle_{\text{trials}}. \tag{4.9}$$

For most functions $\mathscr{A}(\Gamma)$, the integrand will be significant where ρ_{NVT} is significant. In these cases choosing $\rho = \rho_{NVT}$ should give a good estimate of the integral. In this case

$$\langle \mathscr{A} \rangle_{NVT} = \langle \mathscr{A} \rangle_{\text{trials}}. \tag{4.10}$$

(This is not always true and sometimes we choose alternative distributions $\rho(\Gamma)$ (see Section 7.2.2).)

Such a method, with $\rho = \rho_{NVT}$, was originally developed by Metropolis *et al.* [1953]. The problem is not solved, simply rephrased. The difficult job is finding a method of generating a sequence of random states so that by the end of the simulation each state has occurred with the appropriate probability. It turns out that it is possible to do this without ever calculating the normalizing factor for ρ_{NVT}, i.e. the partition function (see eqns (2.11)–(2.13)).

The solution is to set up a Markov chain of states of the liquid, which is constructed so that it has a limiting distribution of ρ_{NVT}. A Markov chain is a sequence of trials that satisfies two conditions:

(a) The outcome of each trial belongs to a finite set of outcomes, $\{\Gamma_1, \Gamma_2, \ldots \Gamma_m, \Gamma_n, \ldots\}$, called the state space.
(b) The outcome of each trial depends only on the outcome of the trial that immediately precedes it.

Two states Γ_m and Γ_n are linked by a transition probability π_{mn}, which is the probability of going from state m to state n. The properties of a Markov chain are best illustrated with a simple example. Suppose the reliability of your mainframe computer follows a certain pattern. If it is up and running on one day it has a 60 per cent chance of running correctly on the next. If, however, it is down, it has a 70 per cent chance of also being down the next day. The state space has two components, up (↑) and down (↓), and the transition matrix has the form

$$\pi = \begin{matrix} \\ \uparrow \\ \downarrow \end{matrix} \begin{matrix} \begin{matrix} \uparrow & \downarrow \end{matrix} \\ \begin{pmatrix} 0.6 & 0.4 \\ 0.3 & 0.7 \end{pmatrix} \end{matrix}. \tag{4.11}$$

If the computer is equally likely to be up or down to begin with, then the initial probability can be represented as a vector, which has the dimensions of the state space

$$\rho^{(1)} = \begin{matrix} (\uparrow & \downarrow) \\ (0.5 & 0.5) \end{matrix}. \tag{4.12}$$

The probability that the computer is up on the second day is given by the matrix equation

$$\rho^{(2)} = \rho^{(1)}\pi = (0.45,\ 0.55) \tag{4.13}$$

i.e. there is a 45 per cent chance of running a program. The next day would give

$$\rho^{(3)} = \rho^{(2)}\pi = \rho^{(1)}\pi\pi = \rho^{(1)}\pi^2 = (0.435,\ 0.565), \tag{4.14}$$

and a 43.5 per cent chance of success. If you are anxious to calculate your chances in the long run, then the limiting distribution is given by

$$\rho = \lim_{\tau \to \infty} \rho^{(1)}\pi^{\tau}. \tag{4.15}$$

A few applications of eqn (4.15) show that the result converges to $\rho = (0.4286, 0.5714)$. It is clear from eqn (4.15) that the limiting distribution, ρ, must satisfy the eigenvalue equation

$$\rho\pi = \rho \tag{4.16a}$$

$$\sum_m \rho_m \pi_{mn} = \rho_n \tag{4.16b}$$

with eigenvalue unity. π is termed a stochastic matrix since its rows add to one

$$\sum_n \pi_{mn} = 1. \tag{4.17}$$

It is the transition matrix for an irreducible Markov chain. (An irreducible or ergodic chain is one where every state can eventually be reached from another state.) More formally, we note that the Perron–Frobenius theorem [Chung 1960; Feller 1957] states that an irreducible stochastic matrix has one left eigenvalue which equals unity, and the corresponding eigenvector is the limiting distribution of the chain. The other eigenvalues are less than unity and they govern the rate of convergence of the Markov chain. The limiting distribution, ρ implied by the chain is quite independent of the initial condition $\rho^{(1)}$ (so don't worry if your machine is likely to be down today).

In the case of a liquid, we must construct a much larger transition matrix, which is stochastic and ergodic (see Chapter 2). In contrast to the previous problem, the elements of the transition matrix are unknown, but the limiting distribution of the chain is the vector with elements $\rho_m = \rho_{NVT}(\Gamma_m)$ for each point Γ_m in phase space. It is possible to determine elements of π which satisfy eqns (4.16) and (4.17) and thereby generate a phase space trajectory in the canonical ensemble. We have considerable freedom in finding an appropriate transition matrix, with the crucial constraint that the elements of the matrix should be independent of Q_{NVT}. A useful trick in searching for a solution of eqn (4.16) is to replace it by the unnecessarily strong condition of 'microscopic reversibility':

$$\rho_m \pi_{mn} = \rho_n \pi_{nm} . \tag{4.18}$$

Summing over all states m and making use of eqn (4.17) we regain eqn (4.16)

$$\sum_m \rho_m \pi_{mn} = \sum_m \rho_n \pi_{nm} = \rho_n \sum_m \pi_{nm} = \rho_n . \tag{4.19}$$

A suitable scheme for constructing a phase space trajectory in the canonical ensemble involves choosing a transition matrix which satisfies eqns (4.17) and (4.18). The first such scheme was suggested by Metropolis *et al.* [1953] and is often known as the asymmetrical solution. If the states m and n are distinct, this solution considers two cases

$$\pi_{mn} = \alpha_{mn} \qquad \rho_n \geqslant \rho_m \qquad m \neq n \tag{4.20a}$$

$$\pi_{mn} = \alpha_{mn}(\rho_n/\rho_m) \qquad \rho_n < \rho_m \qquad m \neq n . \tag{4.20b}$$

It is also important to allow for the possibility that the liquid remains in the same state,

$$\pi_{mm} = 1 - \sum_{n \neq m} \pi_{mn} . \tag{4.20c}$$

In this solution α is a symmetrical stochastic matrix, ($\alpha_{mn} = \alpha_{nm}$), often called the underlying matrix of the Markov chain. The symmetric properties of α can be used to show that for the three cases ($\rho_m = \rho_n$, $\rho_m < \rho_n$, and $\rho_m > \rho_n$) the transition matrix defined in eqn (4.20) satisfies eqns (4.17) and (4.18). It is worth

stressing that it is the symmetric property of α that is essential in satisfying microscopic reversibility in this case. Non-symmetrical α matrices which satisfy microscopic reversibility or just the weaker condition, eqn (4.16), can be constructed but these are not part of the basic Metropolis recipe [Owicki and Scheraga 1977a]. Finally, this solution only involves the ratio ρ_n/ρ_m and is therefore independent of Q_{NVT}.

There are other solutions to eqns (4.17) and (4.18). The symmetrical solution [Wood and Jacobson 1959; Flinn and McManus 1961; Barker 1965], is often referred to as Barker sampling:

$$\pi_{mn} = \alpha_{mn}\rho_n/(\rho_n + \rho_m) \quad m \neq n \tag{4.21a}$$

$$\pi_{mm} = 1 - \sum_{n \neq m} \pi_{mn} . \tag{4.21b}$$

Equation (4.21) also satisfies the condition of microscopic reversibility.

If states of the fluid are generated using transition matrices such as eqns (4.20) and (4.21), then a particular property, $\langle \mathscr{A} \rangle_{\text{run}}$, obtained by averaging over the τ_{run} trials in the Markov chain is related to the average in the canonical ensemble [Chung 1960, p. 99; Wood 1968a]

$$\langle \mathscr{A} \rangle_{NVT} = \langle \mathscr{A} \rangle_{\text{run}} + \mathcal{O}(\tau_{\text{run}}^{-1/2}) \tag{4.22}$$

As mentioned in Chapter 2, we usually restrict simulations to the configurational part of phase space, calculate average configurational properties of the fluid, and add the ideal gas parts after the simulation.

Since there are a number of suitable transition matrices, it is useful to choose a particular solution which minimizes the variance in the estimate of $\langle \mathscr{A} \rangle_{\text{run}}$. Suitable prescriptions for defining the variance in the mean, $\sigma^2(\langle \mathscr{A} \rangle_{\text{run}})$, are discussed in Chapter 7. In particular the 'statistical inefficiency' s (Section 6.4.1)

$$s = \lim_{\tau_{\text{run}} \to \infty} \tau_{\text{run}} \sigma^2 (\langle \mathscr{A} \rangle_{\text{run}})/\sigma^2(\mathscr{A}) \tag{4.23}$$

measures how slowly a run converges to its limiting value. Peskun [1973] has shown that it is reasonable to order two transition matrices,

$$\boldsymbol{\pi}_1 \leqslant \boldsymbol{\pi}_2 \tag{4.24}$$

if each off-diagonal element of $\boldsymbol{\pi}_1$ is less than the corresponding element in $\boldsymbol{\pi}_2$. If this is the case, then

$$s(\langle \mathscr{A} \rangle, \boldsymbol{\pi}_1) \geqslant s(\langle \mathscr{A} \rangle, \boldsymbol{\pi}_2) \tag{4.25}$$

for any property $\mathscr{A}$. If the off-diagonal elements of $\boldsymbol{\pi}$ are large then the probability of remaining in the same state is small and the sampling of phase space will be improved. With the restriction that ρ_m and ρ_n are positive, eqns (4.20) and (4.21) show that the Metropolis solution leads to a lower statistical inefficiency of the mean than the Barker solution.

Valleau and Whittington [1977a] stress that a low statistical inefficiency is not the only criterion for choosing a particular π. Since the simulations are of finite length, it is essential that the Markov chain samples a representative portion of phase space in a reasonable number of moves. All the results derived in this section depend on the ergodicity of the chain (i.e. that there is some non-zero multi-step transition probability of moving between any two allowed states of the fluid). If these allowed states are not connected the MC run may produce a low s but in addition a poor estimate of the canonical average. When the path between two allowed regions of phase space is difficult to find, the situation is described as a bottleneck (see Fig. 2.1). These bottlenecks are always a worry in MC simulations but are particularly troublesome in the simulation of two-phase coexistence [Lee *et al.* 1974], in the simulation of phase transitions [Evans, Tildesley, and Sluckin 1984], and in simulations of ordinary liquids at unusually high density.

Where a comparison has been made between the two common solutions to the transition matrix, eqns (4.20) and (4.21), the Metropolis solution appears to lead to a faster convergence of the chain [Valleau and Whittington 1977b]. The Metropolis method becomes more favourable as the number of available states at a given step increases and as the energy difference between the states increases. (For two-state problems such as the Ising model the symmetric algorithm may be favourable [Cunningham and Meijer 1976]). In the next section we describe the implementation of the asymmetric solution.

4.4 The Metropolis method

To implement the Metropolis solution to the transition matrix, it is necessary to specify the underlying stochastic matrix α. This matrix is designed to take the system from state m into any one of its neighbouring states n with equal probability. There is considerable freedom in choosing α and the only constraint is that $\alpha_{mn} = \alpha_{nm}$. A useful but arbitrary definition of a neighbouring state is illustrated in Fig. 4.3. This diagram shows six atoms in a state m; to construct a neighbouring state n one atom (i) is chosen at random and displaced from its position $\mathbf{r}_i^m$ with equal probability to any point $\mathbf{r}_i^n$ inside the square $\mathscr{R}$. This square is of side $2\delta r_{\max}$ and is centred at $\mathbf{r}_i^m$. In a three-dimensional example, $\mathscr{R}$ would be a small cube. On the computer there are a large but finite number of new positions, $N_{\mathscr{R}}$, for the atom i and in this case α_{mn} can be simply defined as

$$\begin{aligned} \alpha_{mn} &= 1/N_{\mathscr{R}} \qquad & \mathbf{r}_i^n &\in \mathscr{R} \\ \alpha_{mn} &= 0 & \mathbf{r}_i^n &\notin \mathscr{R}. \end{aligned} \tag{4.26}$$

With this choice of α, eqn (4.20) is readily implemented. At the beginning of an MC move an atom is picked at random and given a uniform random displacement along each of the coordinate directions. The maximum displace-

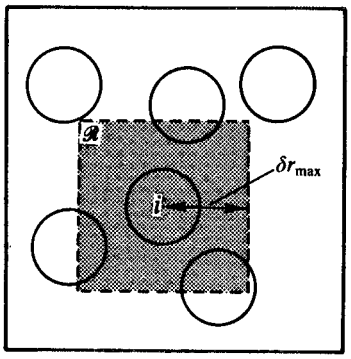

Fig. 4.3 State n is obtained from state m by moving atom i with a uniform probability to any point in the shaded region $\mathscr{R}$.

ment, δr_{max}, is an adjustable parameter that governs the size of the region $\mathscr{R}$ and controls the convergence of the Markov chain. The new position is obtained with the following code. RANF(DUMMY) is a library function for generating a uniform random number on (0, 1); a dummy argument is required by FORTRAN-77 syntax. DRMAX is the maximum displacement δr_{max}.

```
RXINEW = RX(I) + ( 2.0 * RANF ( DUMMY ) - 1.0 ) * DRMAX
RYINEW = RY(I) + ( 2.0 * RANF ( DUMMY ) - 1.0 ) * DRMAX
RZINEW = RZ(I) + ( 2.0 * RANF ( DUMMY ) - 1.0 ) * DRMAX
```

The appropriate element of the transition matrix depends on the relative probabilities of the initial state m and the final state n. There are two cases to consider. If $\delta\mathscr{V}_{nm} = \mathscr{V}_n - \mathscr{V}_m \leqslant 0$ then $\rho_n \geqslant \rho_m$ and eqn (4.20a) applies. If $\delta\mathscr{V}_{nm} > 0$ then $\rho_n < \rho_m$ and eqn (4.20b) applies. (The symbol $\mathscr{V}_m$ is used as a shorthand for $\mathscr{V}(\mathbf{\Gamma}_m)$.) The next step in an MC move is to determine $\delta\mathscr{V}_{nm}$. The determination of $\delta\mathscr{V}_{nm}$ does not require a complete recalculation of the configurational energy of the mth state, just the changes associated with the moving atom. For example (see Fig. 4.4) the change in potential energy is calculated by computing the energy of atom i with all the other atoms before and after the move

$$\delta\mathscr{V}_{nm} = \left(\sum_{j=1}^{N} v(r_{ij}^n) - \sum_{j=1}^{N} v(r_{ij}^m) \right) \tag{4.27}$$

where the sum over the atoms excludes atom i. In calculating the change of energy, the explicit interaction of atom i with all its neighbours out to a cutoff distance r_c is considered. The contribution from atoms beyond the cutoff could be estimated using a mean field correction (see Section 2.8), but in fact the correction for atom i in the old and new positions is exactly the same and does not need to be included explicitly in the calculation of $\delta\mathscr{V}_{nm}$.

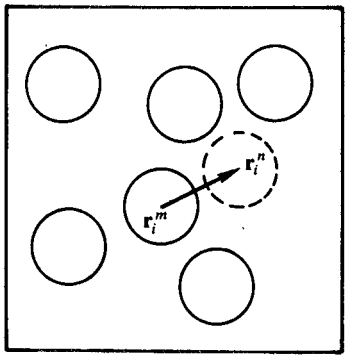

Fig. 4.4 State n is generated from state m by displacing atom i from $\mathbf{r}_i^m$ to $\mathbf{r}_i^n$.

If the move is downhill in energy ($\delta\mathscr{V}_{nm} \leqslant 0$), then the probability of state n is greater than state m and the new configuration is accepted. The method of choosing trial moves ensures that the transition probability $\pi_{mn} = \alpha_{mn}$, the value required by eqn (4.20a).

If the move is uphill in energy ($\delta\mathscr{V}_{nm} > 0$), then the move is accepted with a probability ρ_n/ρ_m according to eqn (4.20b). Again the factor α_{mn} is automatically included in making the move. This ratio can be readily expressed as the Boltzmann factor of the energy difference:

$$\frac{\rho_n}{\rho_m} = \frac{Z_{NVT}^{-1}\exp(-\beta\mathscr{V}_n)}{Z_{NVT}^{-1}\exp(-\beta\mathscr{V}_m)} = \frac{\exp(-\beta\mathscr{V}_n)\exp(-\beta\delta\mathscr{V}_{nm})}{\exp(-\beta\mathscr{V}_n)} = \exp(-\beta\delta\mathscr{V}_{nm}). \tag{4.28}$$

To accept a move with a probability of $\exp(-\beta\delta\mathscr{V}_{nm})$, a random number ξ is generated uniformly on (0, 1). The random number is compared with $\exp(-\beta\delta\mathscr{V}_{nm})$. If it is less than $\exp(-\beta\delta\mathscr{V}_{nm})$ the move is accepted. This procedure is illustrated in Fig. 4.5. During the run, suppose that a particular uphill move,

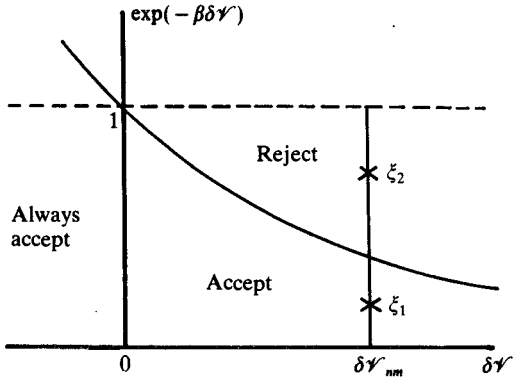

Fig. 4.5 Accepting uphill moves in the MC simulation.

$\delta\mathscr{V}_{nm}$, is attempted. If at that point a random number ξ_1 is chosen (see Fig. 4.5), the move is accepted. If ξ_2 is chosen the move is rejected. Over the course of the run the net result is that energy changes such as $\delta\mathscr{V}_{nm}$ are accepted with a probability $\exp(-\beta\delta\mathscr{V}_{nm})$. If the uphill move is rejected, the system remains in state m in accord with the finite probability π_{mm} of eqn (4.20c). In this case, the atom is retained at its old position and the old configuration is recounted as a new state in the chain. This procedure can be summarized by noting that we accept any move (uphill or downhill) with probability $\min(1, \exp(-\beta\delta\mathscr{V}_{nm}))$, where min has the same meaning as the FORTRAN MIN function.

A complete MC program for a fluid of Lennard-Jones atoms is given in F.11. Here, we show the typical code for the heart of the program, the acceptance and rejection of moves. In this code, DELTV is the energy difference $\delta\mathscr{V}_{nm}$ between the states. One point to note is that we must guard against a trial move which results in significant molecular overlap, since a very large value of $\delta\mathscr{V}_{nm}$ might cause underflow problems in the computation of $\exp(-\beta\delta\mathscr{V}_{nm})$. We do this by testing $\beta\delta\mathscr{V}_{nm}$ (DELTVB below). If it is too large (say > 75) then the move is immediately rejected. This also results in a saving of time, since exponentiation is usually an expensive operation.

```
DELTV  = VNEW - VOLD
DELTVB = BETA * DELTV

IF ( DELTVB .LT. 75.0 ) THEN

   IF ( DELTVB .LE. 0.0 ) THEN

      V      = V + DELTV
      RX(I)  = RXINEW
      RY(I)  = RYINEW
      RZ(I)  = RZINEW
      NACCPT = NACCPT + 1

   ELSEIF ( EXP ( - DELTVB ) .GT. RANF ( DUMMY ) ) THEN

      V      = V + DELTV
      RX(I)  = RXINEW
      RY(I)  = RYINEW
      RZ(I)  = RZINEW
      NACCPT = NACCPT + 1

   ENDIF

ENDIF

NTRIAL = NTRIAL + 1

... accumulate averages ...
```

So far we have said little about the maximum allowed displacement of the atom, δr_{max}, which governs the size of the trial move. If this parameter is too small then a large fraction of moves are accepted but the phase space of the liquid is explored slowly, i.e. consecutive states are highly correlated. If δr_{max} is

too large then nearly all the trial moves are rejected and again there is little movement through phase space. In fact δr_{max} is often adjusted during the simulation so that about half the trial moves are rejected. This adjustment can be handled automatically using the following code, which adjusts the maximum displacement every NADJST trial moves.

```
IF ( MOD ( NTRIAL, NADJST ) .EQ. 0 ) THEN
   RATIO = REAL ( NACCPT ) / REAL ( NADJST )
   IF ( RATIO .GT. 0.5 )  THEN
      DRMAX = DRMAX * 1.05
   ELSE
      DRMAX = DRMAX * 0.95
   ENDIF
   NACCPT = 0
ENDIF
```

It is not clear that an acceptance ratio of 0.5 is optimum. A reported study of the parameter δr_{max} [Wood and Jacobson 1959] suggests that an acceptance ratio of only 0.1 maximizes the root mean square displacement of atoms as a function of computer time. The root mean square displacement is one possible measure of the movement through phase space and the work suggests that a small number of large moves is most cost effective. Few simulators would have the courage to reject nine out of ten moves on this limited evidence and an acceptance ratio of 0.5 is still common. This issue highlights a difficulty in assessing particular simulation methods. The work of Wood and Jacobson was performed on 32 hard spheres, at a particular packing fraction, on a first generation computer. There is no reason to believe that their results would be the same for a different potential, at a different state point on a different machine. The MC technique is time-consuming and since most researchers are more interested in new results rather than methodology there has been little work on the optimization of parameters such as δr_{max} and the choice of transition matrix.

In the original Metropolis method one randomly chosen atom is moved to generate a new state. The underlying stochastic matrix can be changed so that several or all of the atoms are moved simultaneously [Ree 1970; Ceperley, Chester, and Kalos 1977]. $\delta \mathscr{V}_{nm}$ is calculated using a straightforward extension of eqn (4.27) and the move is accepted or rejected using the normal criteria. Chapman and Quirke [1985] have performed a simulation of 32 Lennard-Jones atoms at a typical liquid density and temperature. In this study, all 32 atoms were moved simultaneously, and an acceptance ratio of ≈ 30 per cent was obtained using $\delta r_{max} \approx 0.3\sigma$. Chapman and Quirke found that

equilibration (see Chapter 5) was achieved more rapidly by employing multi-particle moves rather than single-particle moves. The relative efficiency of multi-particle and single-particle moves, as measured by their ability to sample phase space in a given amount of computer time, has not been subjected to systematic study.

A common practice in MC simulation is to select the atoms to move sequentially (i.e. in order of atom index) rather than randomly. This cuts down on the amount of random number generation and is an equally valid method of generating the correctly weighted states [Hastings 1970]. The length of a MC simulation is conveniently measured in 'cycles', i.e. N trial moves whether selected sequentially or randomly. The computer time involved in a MC cycle is comparable (although obviously not equivalent) to that in a MD time step.

The simulation of hard spheres is particularly easy using the MC method. The same Metropolis procedure is used, except that, in this case, the overlap of two spheres results in an infinite positive energy change and $\exp(-\beta\delta\mathscr{V}_{nm}) = 0$. All trial moves involving an overlap are immediately rejected since $\exp(-\beta\delta\mathscr{V}_{nm})$ would be smaller than any random number generated on (0, 1). Equally all moves that do not involve overlap are immediately accepted. As before in the case of a rejection the old configuration is recounted in the average.

The importance sampling technique only generates states that make a substantial contribution to ensemble averages such as the energy. In practice we cannot sum over all the possible states of the fluid and so cannot calculate Z_{NVT}. Consequently, this is not a direct route to the 'statistical' properties of the fluid such as A, S, and μ. In the canonical ensemble there are a number of ways around this problem, such as thermodynamic integration and the particle insertion methods (see Section 2.4). It is also possible to use umbrella sampling to calculate free energy differences (see Chapter 7). Alternatively the problem can be tackled at root by conducting simulations in the grand canonical ensemble (Section 4.6).

4.5 Isothermal–isobaric Monte Carlo

An advantage of the MC method is that it can be readily adapted to the calculation of averages in any ensemble. Wood [1968a, b; 1970] first showed that the method could be extended to the isothermal–isobaric ensemble. This ensemble was introduced in Section 2.2, and in designing a simulation method we should recall that the number of molecules, the temperature, and the pressure are fixed while the volume of the simulation box is allowed to fluctuate. The original constant-NPT simulations were performed on hard spheres and disks, but McDonald [1969, 1972] extended the technique to cover continuous potentials in his study of Lennard-Jones mixtures. This ensemble was thought to be particularly appropriate for simulating mixtures since experimental measurements of excess properties are recorded at constant

pressure and theories of mixing are often formulated with this assumption. The method has also been used in the simulation of single-component fluids [Vorontstov-Vel'yaminov, El'y-Ashevich, Morgenshtern, and Chakovskikh 1970] and in the study of phase transitions [Abraham 1982]. It is worth recalling that at constant N, P, T we should not see two phases coexisting in the same simulation cell, a problem which bedevils the simulation of phase transitions in the canonical ensemble.

In the constant-NPT ensemble the configurational average of a property $\mathscr{A}$ is given by

$$\langle \mathscr{A} \rangle_{NPT} = \frac{\int_0^\infty \mathrm{d}V \exp(-\beta PV)\, V^N \int \mathrm{ds}\, \mathscr{A}(\mathbf{s}) \exp(-\beta \mathscr{V}(\mathbf{s}))}{Z_{NPT}}. \quad (4.29)$$

In eqn (4.29), Z_{NPT} is the appropriate configurational integral eqn. (2.30) and V is the volume of the fluid. Note that in this equation we use a set of scaled coordinates $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N)$ where

$$\mathbf{s} = L^{-1}\mathbf{r}. \quad (4.30)$$

In this case the configurational integral in eqn (4.29) is over the unit cube and the additional factor of V^N comes from the volume element dr. (In this section the simulation box is assumed to be a cube of side $L = V^{1/3}$; the arguments can be easily extended to non-cubic boxes.)

The Metropolis scheme is implemented by generating a Markov chain of states which has a limiting distribution proportional to

$$\exp(-\beta(PV + \mathscr{V}(\mathbf{s})) + N \ln V)$$

and the method used is a direct extension of the ideas discussed in Section 4.4.

A new state is generated by displacing a molecule randomly and/or making a random volume change from V_m to V_n

$$\mathbf{s}_i^n = \mathbf{s}_i^m + \delta s_{\max}(2\boldsymbol{\xi} - \mathbf{1})$$
$$V_n = V_m + \delta V_{\max}(2\xi - 1). \quad (4.31)$$

Here, as usual, ξ is a random number generated uniformly on (0, 1), while $\boldsymbol{\xi}$ is a vector whose components are also uniform random numbers on (0, 1) and $\mathbf{1}$ is the vector (1, 1, 1). $\delta s_{\max}$ and $\delta V_{\max}$ govern the maximum changes in the scaled coordinates of the particles, and in the volume of the simulation box, respectively. Their precise values will depend on the state point studied and they are chosen to produce an acceptance ratio of 35–50 per cent [McDonald 1972]. These values are initial guesses and can be automatically adjusted by the program, although in this case there are two independent maximum displacements and many different combinations will produce a given acceptance ratio.

Once the new state n has been produced the quantity δH is calculated,

$$\delta H_{nm} = \delta \mathscr{V}_{nm} + P(V_n - V_m) - N\beta^{-1} \ln(V_n/V_m). \quad (4.32)$$

δH_{nm} is closely related to the enthalpy change in moving from state m to state n. Moves are accepted with a probability equal to $\min(1, \exp(-\beta\delta H_{nm}))$ using the techniques discussed in Section 4.4. A move may proceed with a change in particle position or a change in volume or a combination of both.

Eppenga and Frenkel [1984] have pointed out that it may be more convenient to make random changes in $\ln V$ rather than in V itself. A random number $\delta(\ln V)$ is chosen uniformly in some range $(-\delta(\ln V)_{\max}, \delta(\ln V)_{\max})$, the volume multiplied by $\exp(\delta(\ln V))$ and the molecular positions scaled accordingly. The only change to the acceptance/rejection procedure is that the factor N in eqn (4.32) is replaced by $N+1$.

One important difference between this ensemble and the canonical ensemble is that when a move involves a change in volume the density of the liquid changes. In this case the long-range corrections to the energy in states m and n are different and must be included directly in the calculation of $\delta\mathscr{V}_{nm}$ (see Section 2.8).

In the general case, changing the volume is computationally more expensive than displacing a molecule. For a molecule displacement there are at most $2(N-1)$ calculations of the pair potential in calculating $\delta\mathscr{V}_{nm}$. In general, a volume change in a pair-additive fluid requires the recalculation of all the $\frac{1}{2}N(N-1)$ interactions. Fortunately, for the simplest potentials, the change in $\mathscr{V}$ with volume can be calculated by scaling. As an example, consider the configurational energy of a Lennard-Jones fluid in state m:

$$\begin{aligned}\mathscr{V}_m &= 4\varepsilon\sum_i\sum_{j>i}\left(\frac{\sigma}{L_m s_{ij}^m}\right)^{12} - 4\varepsilon\sum_i\sum_{j>i}\left(\frac{\sigma}{L_m s_{ij}^m}\right)^6 \\ &= \mathscr{V}_m^{(12)} + \mathscr{V}_m^{(6)}. \end{aligned} \tag{4.33}$$

Here we have divided up the potential into its separate twelfth-power and sixth-power components. If the only change between the states m and n is the length of the box then the energy of the new state is

$$\mathscr{V}_n = \mathscr{V}_m^{(12)}\left(\frac{L_m}{L_n}\right)^{12} + \mathscr{V}_m^{(6)}\left(\frac{L_m}{L_n}\right)^6$$

and

$$\delta\mathscr{V}_{nm} = \delta\mathscr{V}_{nm}^{\text{vol}} = \mathscr{V}_m^{(12)}\left[\left(\frac{L_m}{L_n}\right)^{12} - 1\right] + \mathscr{V}_m^{(6)}\left[\left(\frac{L_m}{L_n}\right)^6 - 1\right]. \tag{4.34}$$

This calculation is extremely rapid and only requires that the two components of the potential energy, $\mathscr{V}^{(12)}$ and $\mathscr{V}^{(6)}$, be stored separately. If the potential cutoff is taken to scale with the box length (i.e. $r_c = s_c L$ with s_c constant) then the separate terms $\mathscr{V}_{\text{LRC}}^{(12)}$ and $\mathscr{V}_{\text{LRC}}^{(6)}$ scale just like $\mathscr{V}^{(12)}$ and $\mathscr{V}^{(6)}$ respectively. If in addition to a box-length change a molecule is simultaneously displaced, then there are two contributions

$$\delta\mathscr{V}_{nm} = \delta\mathscr{V}_{nm}^{\text{dis}} + \delta\mathscr{V}_{nm}^{\text{vol}} \tag{4.35}$$

where $\delta\mathscr{V}_{nm}^{\text{vol}}$ is given by eqn (4.34) and

$$\delta\mathscr{V}_{nm}^{\text{dis}} = \mathscr{V}_n(L_n) - \mathscr{V}_m(L_n). \tag{4.36}$$

Thus the energy change on displacement is obtained using the new box-length L_n (think of scaling the box, followed by moving the molecule).

This simple prescription for the calculation of $\delta\mathscr{V}_{nm}$ relies on there being just one characteristic length in the potential function. This may not be the case for some complicated pair potentials, and it is also not true for most molecular models, where intramolecular bond lengths as well as site–site potentials appear. For an interaction site model, simple scaling would imply a non-physical change in the molecular shape. For these cases the calculation of $\delta\mathscr{V}_{nm}^{\text{vol}}$ is expensive and so volume changes must be carried out much less frequently than the displacement of a particle [Owicki and Scheraga 1977b]. In a constant-NPT simulation of 125 H_2O molecules, Jorgensen [1982] attempted to change the volume of the box once every sixth cycle. The range for a possible volume move was $\sim \pm 50$ Å^3. The code for a constant-NPT simulation is given in program F.12.

By averaging over the states in the Markov chain it is possible to calculate mechanical properties such as the volume and the enthalpy, and various properties related to their fluctuations. In common with the constant-NVT simulation, this method only samples important regions of phase space and it is not possible to calculate the 'statistical' properties such as the Gibbs free energy. During the course of a particular run the virial can be calculated in the usual manner to produce an estimate of the pressure. This calculated pressure (including the long-range correction) should be equal to the input pressure, P, used in eqn (4.32) to generate the Markov chain. This test is a useful check of a properly coded constant-NPT program.

From the limited evidence available, it appears that the fluctuations of averages calculated in a constant-NPT MC simulation are greater than those associated with the averages in a constant-NVT simulation. However, the error involved in calculating excess properties of mixtures in the two ensembles is comparable, since they can be arrived at more directly in a constant-NPT calculation [McDonald 1972].

Finally, constant-pressure simulations of hard disks and spheres, [Wood 1968b, 1970], can be readily performed using the methods described in this section. Wood [1968b] has also developed an elegant method for hard-core systems where the integral over $\exp(-\beta PV)$ in eqn (4.29) is used to define a Laplace transform. The simulation is performed by generating a Markov chain in the transform space using a suitably defined pseudo-potential. This method avoids direct scaling of the box; details can be found in the original paper.

4.6 Grand canonical Monte Carlo

In grand canonical ensemble MC (GCMC) the chemical potential is fixed while the number of molecules fluctuates. The simulations are carried out at

constant μ, V, and T, and the average of some property $\mathscr{A}$ is given by

$$\langle \mathscr{A} \rangle_{\mu VT} = \frac{\sum_{N=0}^{\infty} (N!)^{-1} V^N z^N \int \mathrm{d}\mathbf{s}\, \mathscr{A}(\mathbf{s}) \exp(-\beta \mathscr{V}(\mathbf{s}))}{Q_{\mu VT}} \tag{4.37}$$

where $z = \exp(\beta\mu)/\Lambda^3$ is the activity, Λ is defined in eqn (2.24) and $Q_{\mu VT}$ in eqn (2.32). Again it is convenient to use a set of scaled coordinates $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N)$ defined as in eqn (4.30) for each particular value of N. In common with the other ensembles discussed in this chapter only the configurational properties are calculated during the simulation and the ideal gas contributions are added at the end. A minor complication is that these contributions will depend on $\langle N \rangle_{\mu VT}$, which must be calculated during the run. N is not a continuous variable (the minimum change in N is one), and the sum in eqn (4.37) will not be replaced by an integral.

In GCMC the Markov chain is constructed so that the limiting distribution is proportional to

$$\exp(-\beta(\mathscr{V}(\mathbf{s}) - N\mu) - \ln N! - 3N \ln \Lambda + N \ln V). \tag{4.38}$$

A number of methods of generating this chain have been proposed. A method applied in early studies of lattice systems [Salsburg, Jacobson, Fickett, and Wood 1959; Chesnut 1963], uses a set of variables $(c_1, c_2 \ldots)$, each taking the value 0 (unoccupied) or 1 (occupied), to define a configuration. In the simplest approach a trial move attempts to turn either a 'ghost' site ($c_i = 0$) into a real site ($c_i = 1$) or vice versa.

This method has been extended to continuous fluids by Rowley, Nicholson, and Parsonage [1975] and used more recently by Yao, Greenkorn, and Chao [1982]. In this application real and ghost molecules are moved throughout the system using the normal Metropolis method for displacement. This means that 'ghost' moves are always accepted because no interactions are involved. In addition there are frequent conversion attempts between 'ghost' and real molecules. Unfortunately a 'ghost' molecule tends to remain close to the position at which its real precursor was destroyed, and is likely to re-materialize, at some later step in the simulation, in this same 'hole' in the liquid. This memory effect does not lead to incorrect results [Barker and Henderson 1976], but may result in a slow convergence of the chain. The total number of real and ghost molecules, M, must be chosen so that if all the molecules became real $\mathscr{V}$ would be very high for all possible configurations. In this case the sum in eqn (4.37) can be truncated at M. This analysis makes it clear that, in GCMC simulations, we are essentially transferring molecules between our system of interest and an ideal gas system, each of which is limited to a maximum of M molecules. Thus the system properties are measured relative to those of this restricted ideal gas; if M is sufficiently large this should not matter.

Most workers now adopt the original method of Norman and Filinov [1969]. In this technique there are three different types of move:

(a) a molecule is displaced;
(b) a molecule is destroyed (no record of its position is kept);
(c) a molecule is created at a random position in the fluid.

Displacement is handled using the normal Metropolis method. If a molecule is destroyed the ratio of the probabilities of the two states is (N is the number of molecules initially in state m)

$$\frac{\rho_n}{\rho_m} = \exp(-\beta\delta\mathscr{V}_{nm})\exp(-\beta\mu)\frac{N\Lambda^3}{V}, \tag{4.39}$$

which in terms of the activity is

$$\frac{\rho_n}{\rho_m} = \exp(-\beta\delta\mathscr{V}_{nm} + \ln(N/zV)) = \exp(-\beta\delta D_{nm}). \tag{4.40}$$

Here we have defined the 'destruction function' δD_{nm}. A destruction move is accepted with probability $\min(1, \exp(-\beta\delta D_{nm}))$ using the methods of Section 4.4. Finally, in a creation step, similar arguments give

$$\frac{\rho_n}{\rho_m} = \exp(-\beta\delta\mathscr{V}_{nm} + \ln(zV/N+1)) = \exp(-\beta\delta C_{nm}) \tag{4.41}$$

(defining the 'creation function' δC_{nm}) and the move is accepted or rejected using the same criteria.

In this scheme there is the danger of using an underlying stochastic matrix which is unsymmetric with respect to creation/destruction. The condition of microscopic reversibility can be satisfied by making the probability of an attempted creation, α^c, equal to the probability of an attempted destruction, α^d [Nicholson and Parsonage 1982, p. 154]. The method outlined allows for the destruction or creation of only one molecule at a time. Except at low densities, moves which involve the addition or removal of more than one molecule would be highly improbable and such changes are not cost effective [Norman and Filinov 1969].

Although α^d must equal α^c there is some freedom in chosing between creation/destruction and a simple displacement, α^m. Again Norman and Filinov [1969] varied α^m and found that $\alpha^m = \alpha^d = \alpha^c = 1/3$ gave the fastest convergence of the chain, and these are the values commonly employed. Thus moves, destructions, and creations are selected at random, with equal probability.

Typically, the configurational energy, pressure, and density are calculated as ensemble averages during the course of the GCMC simulations. The beauty of this type of simulation is that the free energy can be calculated directly,

$$A/N = \mu - \langle \mathscr{P} \rangle_{\mu VT} V/\langle N \rangle_{\mu VT} \tag{4.42}$$

and using eqn (4.42) it is possible to determine all the 'statistical' properties of the liquid.

Variations on the method described in this section have been employed by a number of workers. The Metropolis method for creation and destruction can be replaced by a symmetrical algorithm. In this case the decisions for creation and destruction are respectively

$$\text{create if} \quad \left(1+\frac{N+1}{zV}\exp\left(\beta\delta\mathscr{V}_{nm}\right)\right)^{-1} \geqslant \xi$$

and

$$\text{destroy if} \quad \left(1+\frac{zV}{N}\exp\left(\beta\delta\mathscr{V}_{nm}\right)\right)^{-1} \geqslant \xi$$

with ξ generated uniformly on (0, 1).

Adams [1974, 1975] has also suggested an alternative formulation which splits the chemical potential into the ideal gas and excess parts:

$$\begin{aligned}\mu &= \mu^{\text{ex}} + \mu^{\text{id}} \\ &= (\mu^{\text{ex}} + kT\ln\langle N\rangle_{\mu VT}) + kT\ln(\Lambda^3/V) \\ &= kTB + kT\ln(\Lambda^3/V). \end{aligned} \tag{4.43}$$

Adams performed the MC simulation at constant B, V, and T, where B is defined by eqn (4.43). μ can be obtained by calculating $\langle N\rangle_{\mu VT}$ during the run and using it in eqn (4.43). The technique is completely equivalent to the normal method at constant z, V, and T.

There are a number of technical points to be considered in performing GCMC. In common with the constant-NPT ensemble, the density is not constant during the run. In these cases the long-range corrections must be included directly in the calculation of $\delta\mathscr{V}_{nm}$. The corrections should also be applied during the run to other configurational properties such as the virial. If this is not done, difficulties may arise in correcting the pressure at the end of the simulation: this can affect the calculation of the free energy through eqn (4.42) [Barker and Henderson 1976; Rowley, Nicholson, and Parsonage 1978].

A problem which is peculiar to GCMC is that, when molecules are created or destroyed, the array indices which identify the molecule need to be reordered. This problem can be handled neatly using the following technique [Nicholson 1984]. In this simple illustration, we consider a simulation which begins with six molecules and where we expect a maximum of ten. An array LOCATE is the key to which molecules are 'alive' at the current step of the simulation. At the first step LOCATE looks like

I	1	2	3	4	5	6	7	8	9	10
LOCATE(I)	1	2	3	4	5	6	0	0	0	0

$N = 6$ (a)

If molecule 3 is destroyed the array is updated and 3 is moved to the 'dead' area of the array.

I	1	2	3	4	5	6	7	8	9	10
LOCATE(I)	1	2	4	5	6	3	0	0	0	0

$N = 5$ (b)

If a new molecule is created it is given the index LOCATE($N + 1$) i.e. the array would remain unchanged but N would be increased by one.

I	1	2	3	4	5	6	7	8	9	10
LOCATE(I)	1	2	4	5	6	3	0	0	0	0

$N = 6$ (c)

Suppose a second new molecule is created. In this case LOCATE($N + 1$) = 0 so the new molecule index is set to $N + 1$ and N is then increased by one.

I	1	2	3	4	5	6	7	8	9	10
LOCATE(I)	1	2	4	5	6	3	7	0	0	0

$N = 7$ (d)

At any stage in the program it is easy to search over all the molecules actually present by running over a loop with upper index N as follows:

```
      DO 10 I = 1, N

         IATOM = LOCATE(I)
         RXI   = RX(IATOM)
         RYI   = RY(IATOM)
         RZI   = RZ(IATOM)

         ... calculate energy etc. ...

10    CONTINUE
```

In the Norman and Filinov method, the new molecule 3, which is in LOCATE(6) at the end of step (c), has coordinates RX(3), RY(3), RZ(3), which are chosen at random and which are not related to the original coordinates of molecule 3 at step (a). The code for creation and destruction attempts in GCMC is given in program F.13 and the code for updating and tidying the array LOCATE is given in program F.14.

Grand canonical simulations are more complicated to program than those in the canonical ensemble. The advantage of the method is that it provides a direct route to the 'statistical' properties of the fluid. For example, by determining the free energy of two different solid structures in two independent GCMC simulations we can say which of the two structures is thermodynamically stable at a particular μ and T. GCMC is particularly useful for studying inhomogeneous systems such as monolayer and multilayer adsorption near a surface [Whitehouse, Nicholson, and Parsonage 1983] or the electrical double-layer [Carnie and Torrie 1984; Guldbrand, Jönsson, Wennerström, and Linse 1984]. In these systems the surface often attracts the molecules strongly so that when a molecule diffuses into the vicinity of the surface it may tend to remain there throughout the simulation. GCMC additionally destroys particles in the dense region near the surface and creates them in the dilute region away from the surface. In this way it should encourage efficient sampling of some less likely but allowed regions of phase space as well as helping to break up metastable structures near the surface.

GCMC simulations of fluids have not been used widely. The problem is that as the density of the fluid is increased the probability of successful creation or destruction steps becomes small. Creation attempts fail because of the high risk of overlap. Destruction attempts fail because the removal of a particle without the subsequent relaxation of the liquid structure results in the loss of attractive interactions. Clearly this means that destructions in the vicinity of a surface may be infrequent and this somewhat offsets the advantage of GCMC in the simulation of adsorption [Nicholson 1984]. To address these problems, Mezei [1980] has extended the basic method to search for cavities in the fluid which are of an appropriate size to support a creation. Once these cavities are located, creation attempts are made more frequently in the region of the cavity. In the Lennard-Jones fluid at $T^* = 2.0$, the highest density at which the system could be successfully studied was increased from $\rho^* = 0.65$ (conventional GCMC) to $\rho^* = 0.85$ (extended GCMC). The techniques for preferential sampling close to a molecule or a cavity are discussed in Section 7.3.

4.7 Molecular liquids

4.7.1 Rigid molecules

In the MC simulation of a molecular liquid the underlying matrix of the Markov chain is altered to allow moves which usually consist of a combined translation and rotation of one molecule. Chains involving a number of purely translational and purely rotational steps are perfectly proper but are not usually exploited in the simulation of molecular liquids. (There have been a number of simulations of idealized models of liquid crystals and plastic crystals where the centres of the molecules are fixed to a three-dimensional

lattice. These simulations consist of purely rotational moves [see e.g. Luckhurst and Simpson 1982; O'Shea 1978].)

The translational part of the move is carried out by randomly displacing the centre of mass of a molecule along each of the space-fixed axes. As before the maximum displacement is governed by the adjustable parameter δr_{max}. The orientation of a molecule is often described in terms of the Euler angles defined in Section 3.3.1. A change in orientation can be achieved by taking small random displacements in each of the Euler angles of molecule i.

$$\phi_i^n = \phi_i^m + (2\xi_1 - 1)\delta\phi_{max} \tag{4.44a}$$

$$\theta_i^n = \theta_i^m + (2\xi_2 - 1)\delta\theta_{max} \tag{4.44b}$$

$$\psi_i^n = \psi_i^m + (2\xi_3 - 1)\delta\psi_{max} \tag{4.44c}$$

where $\delta\phi_{max}$, $\delta\theta_{max}$, and $\delta\psi_{max}$ are the maximum displacements in the Euler angles.

In an MC step the ratio of the probabilities of the two states is given by

$$\frac{\rho_n}{\rho_m} = \frac{\exp(-\beta(\mathscr{V}_m + \delta\mathscr{V}_{nm}))\mathrm{d}\mathbf{r}^n\mathrm{d}\boldsymbol{\Omega}^n}{\exp(-\beta\mathscr{V}_m)\mathrm{d}\mathbf{r}^m\mathrm{d}\boldsymbol{\Omega}^m}. \tag{4.45}$$

The appropriate volume elements have been included to convert the probability densities into probabilities. $\mathrm{d}\boldsymbol{\Omega}^m = \prod_{i=1}^{N} \mathrm{d}\boldsymbol{\Omega}_i^m$ and $\mathrm{d}\boldsymbol{\Omega}_i^m = \sin\theta_i^m \mathrm{d}\theta_i^m \mathrm{d}\psi_i^m \mathrm{d}\phi_i^m/\Omega$ for molecule i in state m. Ω is a constant which is $8\pi^2$ for non-linear molecules. In the case of linear molecules, the angle ψ is not required to define the orientation, and $\Omega = 4\pi$. The volume elements for states m and n have not previously been included in the ratio ρ_n/ρ_m (see eqn (4.28)), for the simple reason that they are the same in both states for a translational move, and cancel. For a move which only involves one molecule i

$$\frac{\rho_n}{\rho_m} = \exp(-\beta\delta\mathscr{V}_{nm})\frac{\sin\theta_i^n}{\sin\theta_i^m}. \tag{4.46}$$

The ratio of the sines must appear in the transition matrix π_{mn} either in the acceptance/rejection criterion or in the underlying matrix element α_{mn}. This last approach is most convenient. It amounts to choosing random displacements in $\cos\theta_i$ rather than in θ_i:

$$\cos\theta_i^n = \cos\theta_i^m + (2\xi_2 - 1)\delta(\cos\theta)_{max} \tag{4.47}$$

and adopting the usual Metropolis recipe of accepting or rejecting with a probability of min $(1, \exp(-\beta\delta\mathscr{V}_{nm}))$. Including the $\sin\theta$ factor in the underlying chain avoids difficulties with $\theta_i^m = 0$ analogous to the problems mentioned in Section 3.3.1. Equations (4.44a), (4.44c), and (4.47) move a molecule from one orientational state into any one of its neighbouring orientational states with equal probability and fulfil the condition of microscopic reversibility.

It is useful to keep the angles which describe the orientation of a particular molecule in the appropriate range $(-\pi, \pi)$ for ψ and ϕ, and $(0, \pi)$ for θ. This is not essential, but avoids unnecessary work and possible overflow in the subsequent evaluation of any trigonometric functions. This can be done by a piece of code which is rather like that used to implement periodic boundary conditions. If DPHIMX is the maximum change in ϕ, and TWOPI stores the value 2π,

```
PHINEW = PHIOLD + ( 2.0 * RANF ( DUMMY ) - 1.0 ) * DPHIMX
PHINEW = PHINEW - ANINT ( PHINEW / TWOPI ) * TWOPI
```

with similar code for ψ. In the case of eqn (4.47), it is necessary to keep $\cos\theta$ in the range $(-1, 1)$:

```
COSNEW = COSOLD + ( 2.0 * RANF ( DUMMY ) - 1.0 ) * DCOSMX
COSNEW = COSNEW - ANINT ( COSNEW / 2.0 ) * 2.0
```

Note that when the ANINT function is not zero the molecule is rotated by π.

An alternative method for rotating the molecules was originally proposed by Barker and Watts [1969] in their MC simulation of water. It involves selecting a molecule and rotating it by a random amount $\delta\gamma$ (selected uniformly in the usual way) about one of the three space-fixed axes chosen at random. For example we consider a fluid of linear molecules (these ideas can be readily extended to non-linear molecules). In this case it is more convenient to represent the molecular orientation by a unit vector **e** fixed in the molecule. The orientation of molecule i is represented by a vector with components

$$\begin{aligned} e_{ix} &= \cos\phi_i \sin\theta_i \\ e_{iy} &= \sin\phi_i \sin\theta_i \\ e_{iz} &= \cos\theta_i \,. \end{aligned} \tag{4.48}$$

A new configuration is generated using

$$\mathbf{e}_i^n = \mathbf{A}_x \mathbf{e}_i^m \tag{4.49}$$

where

$$\mathbf{A}_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\delta\gamma & \sin\delta\gamma \\ 0 & -\sin\delta\gamma & \cos\delta\gamma \end{pmatrix} \tag{4.50}$$

and the equation corresponds to a rotation of $\delta\gamma$ about the space-fixed x axis. There are similar equations for rotations about the y and z axes,

$$\mathbf{A}_y = \begin{pmatrix} \cos\delta\gamma & 0 & -\sin\delta\gamma \\ 0 & 1 & 0 \\ \sin\delta\gamma & 0 & \cos\delta\gamma \end{pmatrix} \tag{4.51}$$

and

$$\mathbf{A}_z = \begin{pmatrix} \cos\delta\gamma & \sin\delta\gamma & 0 \\ -\sin\delta\gamma & \cos\delta\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.52}$$

The advantage of this method is that the orientation of the molecule can be stored as one or more vectors. For interaction site model fluids, this means that the expensive evaluation of trigonometric functions can be avoided completely. This is not the case with methods that involve changes in the Euler angles. Similar formulae apply when quaternions are used to represent molecular orientations [Vesely 1982].

A third method for changing the orientation of a molecule has been suggested by Jansoone [1974]. The new trial orientation, $\mathbf{e}_i''$, is chosen randomly and uniformly on a region of the surface of a sphere with the constraint that

$$1 - \mathbf{e}_i'' \cdot \mathbf{e}_i''' < d \ll 1 . \tag{4.53}$$

d controls the size of the maximum displacement and a sensible first guess for this parameter is 0.2. There are a number of methods for generating a random vector on the surface of a sphere (see Appendix G.3). These can be easily combined with the method for generating randomly and uniformly in a restricted region (see Appendix G.4) to produce a simple algorithm which will generate orientations with the constraint eqn (4.53). Examples of code for these three methods of generating a new orientation are given in program F.15.

One difficulty with MC methods for molecular fluids is that there are usually a number of parameters governing the maximum translational and orientational displacement of a molecule during a move. As usual these parameters can be adjusted automatically to give an acceptance rate of ≈ 0.5, but there is not a unique set of maximum displacement parameters which will achieve this. A sensible set of values is best obtained by trial and error for the particular simulation in hand.

The MC method is particularly useful for simulating hard-core molecules. The complicated MD schemes mentioned in Section 3.6.2 can be avoided and the program consists simply of choosing one of the above schemes for moving a molecule and an algorithm for checking for overlap. The heart of a simple MC program for hard dumb-bells is given in F.16.

The MC method has been used successfully in the canonical ensemble for simulating hard-core molecules [Streett and Tildesley 1978; Wojcik and Gubbins 1983] and more realistic linear and non-linear molecules [Barker and Watts 1969; Romano and Singer 1979]. Simulations of molecular fluids have also been attempted in the isothermal–isobaric ensemble [Owicki and Scheraga 1977b; Eppenga and Frenkel 1984]. To our knowledge there have been no simulations of molecular liquids in the grand canonical ensemble.

4.7.2 Non-rigid molecules

Non-rigidity introduces new difficulties into the MC technique. The problem in this case is to find a suitable set of generalized coordinates to describe the positions and momenta of the molecules. Once the generalized momentum coordinates have been established, the integrations over the momenta can be performed analytically which will leave just the configurational part of the ensemble average. However, the integration over momenta will produce complicated Jacobians in the configurational integral, one for each molecule (see Section 2.10). The Jacobian will be some function of the generalized orientational variables, θ, ϕ which describe the overall orientation of the molecule and the bond bending and torsion angles which describe the internal configuration. A simple example of this type of term is the $\sin\theta_i$ in the configurational integral for rigid molecules, which comes from the integration over the momenta $(p_\phi)_i$. As we have already seen in Section 4.7.1, these Jacobians are important in calculating the ratio ρ_n/ρ_m used in generating the Markov chain in the Metropolis method or, correspondingly, in designing the correct underlying stochastic matrix. For non-rigid molecules, correctly handling the Jacobian terms is more difficult.

This problem can be solved satisfactorily for the class of non-rigid molecules where the overall moment of inertia is independent of the coordinates of internal rotation (e.g. *iso*-butane, acetone) [Pitzer and Gwinn 1942]. Generalized coordinates have also been developed for a non-rigid model of butane, which does not fall into this simple class [Ryckaert and Bellemans 1975; Pear and Weiner 1979], but the expressions are complicated and become increasingly so for longer molecules.

One way of working with generalized coordinates is as follows. In butane (see Section 1.3), it is possible to constrain bond lengths and bond bending angles, while allowing the torsional angle to change according to its potential function. The movement of the molecule in the simulation is achieved by random movements of randomly chosen atoms subject to the required constraints [Curro 1974]. An example of such a technique is shown for butane in Fig. 4.6.

A typical MC sequence might be (assuming that each move is accepted):

(a) atom 1 is moved by rotating around the 2–3 bond;
(b) atoms 1 and 2 are moved simultaneously by rotating around the 3–4 bond;
(c) atom 4 is moved by rotating around the 2–3 bond.

Moves (a) and (c) involve a random displacement of the torsional angle ϕ, in the range $(-\pi, \pi)$. The entire molecule is translated and rotated through space by making random rotations of atoms around randomly chosen bonds. We can also include an explicit translation of the whole molecule, and an overall rotation about one of the space-fixed axes. The disadvantage of this simple

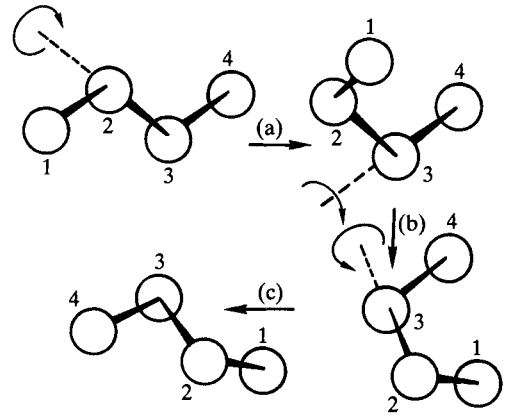

Fig. 4.6 A possible method for moving a chain molecule (butane), subject to bond length and angle constraints, in an MC simulation.

approach at high density is that a small rotation around the 1–2 bond can cause a substantial movement of atom 4, which is likely to result in overlap and a high rejection rate for new configurations.

If we consider the case of the simplified butane molecule introduced in Sections 1.3.3 and 2.10, then a trial MC move might consist of a translation and rotation of the whole molecule and a change in the internal configuration made by choosing a random increment in $d(\cos\theta)$, $d(\cos\theta')$, and $d\phi$ (see Fig. 1.8). To avoid the artefacts associated with the constraint approximation, the Markov chain should be generated with a limiting distribution proportional to

$$\exp(-\beta(\mathscr{V}+\mathscr{V}_c)) = \exp(-\beta(\mathscr{V}+\tfrac{1}{2}k_B T \ln[2+\sin^2\theta+\sin^2\theta'])). \tag{4.54}$$

If θ and θ' stay close to their equilibrium values throughout, it might be possible to introduce only a small error by neglecting the constraint potential $\mathscr{V}_c$ in eqn (4.54). The constraint term becomes more complicated and important in the case of bond-angle constraints. For this reason there have been few Metropolis MC simulations of long-chain flexible molecules. The technique of choice is constraint dynamics, using quadratic bond-angle potentials to avoid the metric term in the potential (see Section 3.4).

There have been a considerable number of studies of polymer systems using the MC method [Binder 1984]. Single chains can be simulated using crude MC methods. In this technique a polymer chain of specified length is built up randomly in space [Lal and Spencer 1971] or on a lattice [Suzuki and Nakata 1970]. A chain is abandoned if a substantial overlap is introduced during its construction. When a large number N of chains of the required length have been produced, the average of a property (such as the end-to-end distance) is calculated from

$$\langle \mathscr{A} \rangle = \frac{\sum_{i=1}^{N} \mathscr{A}_i \exp(-\beta \mathscr{V}_i)}{\sum_{i=1}^{N} \exp(-\beta \mathscr{V}_i)} \tag{4.55}$$

where the sums range over all the N polymer chains. The approach is inapplicable for a dense fluid of chains. A more conventional MC method, which avoids this problem, was suggested by Wall and Mandel [1975]. In a real fluid a chain is likely to move in a slithering fashion: the head of the chain moves to a new position and the rest of the chain follows like a snake or lizard. This type of motion is termed 'reptation' [de Gennes 1971]. A successful MC algorithm would mimic this motion. The 'slithering snake' model was originally applied to a polymer on a two-dimensional lattice and a simple example is shown in Fig. 4.7.

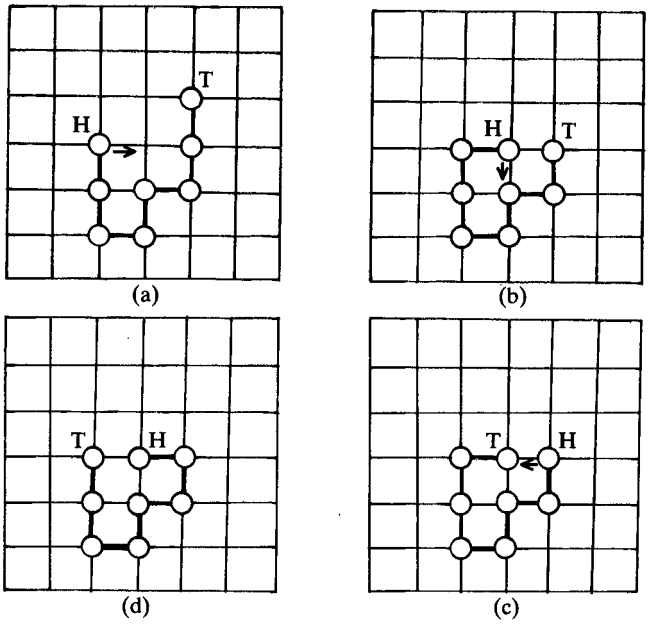

Fig. 4.7 The slithering snake polymer on a two-dimentional lattice. The configurations are generated in order (a), (b), (c), and (d).

A polymer of eight segments is simulated. One end of the molecule is chosen at random to be the head (H) while the other is the tail (T). The head is moved to a new position on the lattice, all the other atoms move one site along the chain and the tail position becomes vacant. In Fig. 4.7(b) the proposed head move is rejected, since the chains are not allowed to overlap. The resulting

configuration, which is identical to the previous one, is included in the averaging, and the simulation proceeds. We can see that random selection of the head and tail is important, since otherwise the system might become locked with the head unable to move. In Fig. 4.7(c), head and tail have been interchanged, and the proposed move is accepted, since the tail position of the chain will be empty when the move is complete. Although this example illustrates the method for a single polymer chain, it is easily extended to a dense fluid of chains, since only one atom moves in generating each new configuration.

Such 'reptation MC' algorithms have been applied to chains on a three-dimensional lattice [Wall, Chin, and Mandel 1977] and to continuum fluids [Brender and Lax 1983]. Bishop, Ceperley, Frisch, and Kalos [1980] have developed a reptation algorithm which is suitable for a chain with arbitrary intermolecular and intramolecular potentials in a continuum fluid. The method exploits the Metropolis solution to the transition matrix to asymptotically sample the Boltzmann distribution. In the case studied by Bishop and coworkers, the model consists of N chains each containing n_a atoms. All the atoms in the fluid interact through the repulsive part of the Lennard-Jones potential, $v^{\mathrm{RLJ}}(r)$, eqn (1.10a); this interaction controls the excluded volume of the chains. In addition, adjacent atoms in the same chain interact through a modified harmonic potential,

$$v^{\mathrm{H}}(r) = \begin{cases} -0.5\,k\,\sigma_1^2 \ln[1-(r/\sigma_1)^2] & 0 \leqslant r \leqslant \sigma_1 \\ \infty & r > \sigma_1 \end{cases} \tag{4.56}$$

where, typically, $\sigma_1 = 1.95\sigma$ and $k = 20$. Each chain is considered in turn and one end is chosen randomly as the head. The initial coordinates of the atoms in the ith chain are $(\mathbf{r}_{i1}, \mathbf{r}_{i2}, \ldots, \mathbf{r}_{in_a})$. A new position is selected for the head, atom n_a,

$$\mathbf{r}' = \mathbf{r}_{in_a} + \delta\mathbf{r}\,. \tag{4.57}$$

The direction of $\delta\mathbf{r}$ is chosen at random on the surface of a sphere, and the magnitude δr is chosen according to the probability distribution $\exp(-\beta v^{\mathrm{H}}(r))$ using a rejection technique (see Appendix G). Thus, the intramolecular bonding potential is used in selecting the trial move (other examples of introducing bias in this way will be seen in Chapter 7). The chain has a new trial configuration $(\mathbf{r}_{i2}, \mathbf{r}_{i3}, \ldots, \mathbf{r}_{in_a}, \mathbf{r}')$. The change in non-bonded interactions in creating a new configuration is calculated by summing over all the atoms

$$\begin{aligned} \delta\mathscr{V}_{nm} = &\sum_{a=2}^{n_a} v^{\mathrm{RLJ}}(|\mathbf{r}_{ia} - \mathbf{r}'|) - v^{\mathrm{RLJ}}(|\mathbf{r}_{ia} - \mathbf{r}_{i1}|) \\ &+ \sum_{j \neq i} \sum_{a=1}^{n_a} v^{\mathrm{RLJ}}(|\mathbf{r}_{ja} - \mathbf{r}'|) - v^{\mathrm{RLJ}}(|\mathbf{r}_{ja} - \mathbf{r}_{i1}|)\,. \end{aligned} \tag{4.58}$$

Non-bonded interactions from chain i and from all other chains j are included here. $\delta \mathscr{V}_{nm}$ is used to decide whether the move should be accepted or rejected according to the Metropolis criteria. As usual, rejected moves are recounted. The approach works well; there are no geometrical constraints to take into account in this example, all the atoms being free to move under the influence of the potentials.