

V. Notes: Monte Carlo Simulations

The Monte Carlo method is an example of what is called a sampling experiment. These experiments typically involve the generation of a big number of random numbers followed by a limited number of arithmetic and logical operations. One of the so-called *simple* Monte Carlo most popular uses is in obtaining values of integrals by generating a uniform sampling of the integrand and adding it up to obtain an answer. There are several problems with this scheme, especially when applied to thermodynamic states. In general, these problems are best taken care by using the *Metropolis* Monte Carlo, often called the importance sampling technique, that aims to generate a trajectory in phase space which samples only states that make a large contribution to the integral from a chosen statistical ensemble.

1. Simple Monte Carlo

The simple MC method can be illustrated by the following example. Consider a circle as in Figure V.1 and let's estimate the value of π by considering one quadrant and determining the area of the shaded region by generating random sample points in that quadrant.

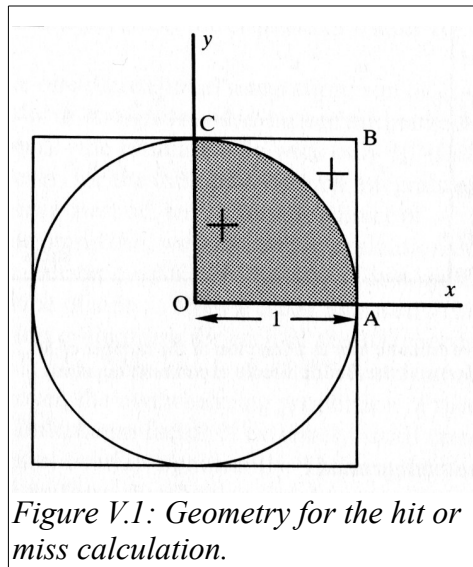


Figure V.1: Geometry for the hit or miss calculation.

The procedure is the following, for each trial point we generate two random numbers that will constitute the 2D coordinates of a point inside the square OABC. The distance of this random point is then compared with the radius of the circle. If this distance is less than or equal to the radius, the trial is considered in the shaded region and a hit is scored. For a total of τ_{shots} trials generated, and τ_{hit} hits scored, then

$$\pi \approx \frac{4 \times \text{area under the curve } AC}{\text{area of the square } OABC} = \frac{4\tau_{hit}}{\tau_{shots}}. \quad (1)$$

The key to this method is the generation of $2\tau_{shots}$ random numbers from a uniform distribution.

The estimate of this area will depend on the numbers of trials. The results of this hit or miss experiment are shown in Figure V.2. After 10^7 trials the MC estimate is still 3.14173 for π . Because the error in the estimate of π is of order $O(\tau_{shots}^{-1/2})$, to calculate another decimal place would require an order of magnitude increase in the number of trials.

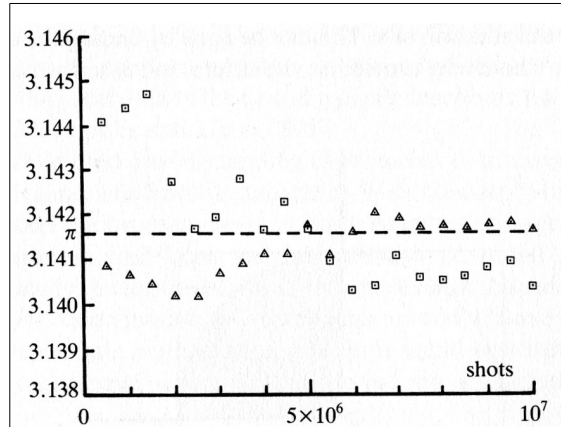


Figure V.2: Results (triangles) for π calculation using the hit or miss calculation. Squares correspond to Buffon's needle experiment (see lecture notes).

An illustration of this idea is shown in Figure V.3 (b) in particular, but also works in general. In Figure V.3 (a) we determine the area of the shaded area by generating again a series of random points within

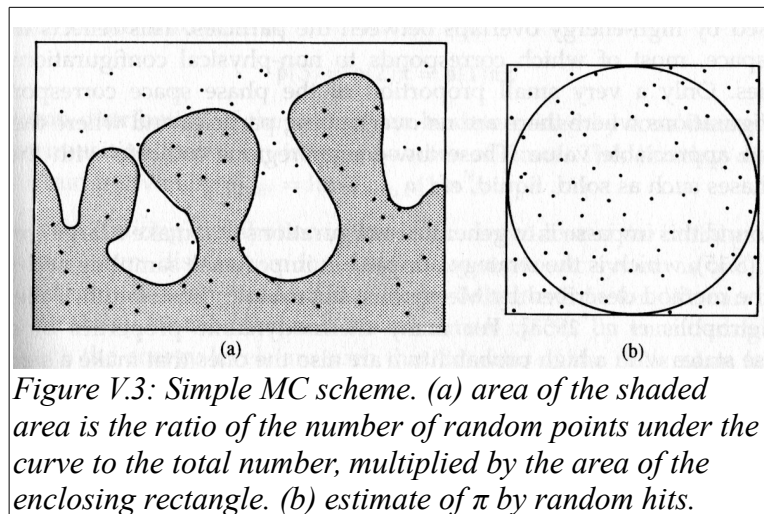


Figure V.3: Simple MC scheme. (a) area of the shaded area is the ratio of the number of random points under the curve to the total number, multiplied by the area of the enclosing rectangle. (b) estimate of π by random hits.

the bounding rectangle. The area under the curve is then calculated by multiplying the bounding area by the ratio of the number of trial points that lie under the curve to the total number of points generated.

This hit-or-miss method described in the simple MC can be improved by using the sample mean

method. In this improved method, our goal is to calculate an integral as in equation

$$F = \int_{x_1}^{x_2} dx f(x) \quad (2)$$

that can be rewritten

$$F = \int_{x_1}^{x_2} dx \left(\frac{f(x)}{\rho(x)} \right) \rho(x) \quad (3)$$

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

$$F = \left\langle \frac{f(\zeta_\tau)}{\rho(\zeta_\tau)} \right\rangle_{trials} \quad (4)$$

where the brackets represent an average over all trials. To recover the case of the simple MC method above, we choose a uniform probability distribution, as e.g.

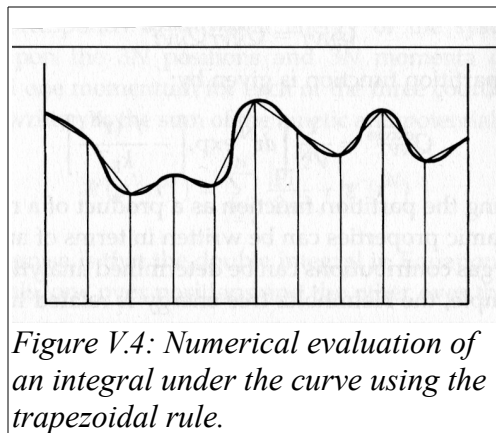
$$\rho(x) = \frac{1}{(x_2 - x_1)} \quad (5)$$

for $x_1 \leq x \leq x_2$, and then the integral F can be estimated as

$$F \approx \frac{(x_2 - x_1)}{\tau_{max}} \sum_{\tau=1}^{\tau_{max}} f(\zeta_\tau) . \quad (6)$$

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 $x_2=1$.

For the simple one-dimensional integration, the simple MC technique is not competitive with straightforward numerical methods such as using the trapezoidal rule, as in Figure V.4, where the area



under the curve is approximated by the sum of the areas of the individual trapezoids. In such a method

the estimate of π can be realized to a higher accuracy (3.141593) by only 10^4 function evaluations. However, when performing integrals that involve average values of macroscopic quantities using N particles, we have a $3N$ dimensional integral in which the number of function evaluations would be m^{3N} , where m is the number of points needed to determine the integral in each dimension. This number is enormous. For example, with just 50 particles and three points per dimension, a total of 3^{150} ($\sim 10^{71}$) evaluations would be required. This is clearly not a feasible approach.

For the multidimensional integrals required in statistical mechanics, some sort of MC method is the only choice. However, taking the simple MC approach is still not possible. For example, the average potential energy given by the integral

$$\langle V(\mathbf{r}^N) \rangle = \int d\mathbf{r}^N V(\mathbf{r}^N) \rho(\mathbf{r}^N) \quad (7)$$

is our problematic $3N$ dimensional integral where

$$\rho(\mathbf{r}^N) = \frac{\exp[-V(\mathbf{r}^N)/k_B T]}{Z} \quad (8)$$

and

$$Z = \int d\mathbf{r}^N \exp\left(-\frac{V(\mathbf{r}^N)}{k_B T}\right). \quad (9)$$

To calculate the partition function in the simple MC we would:

1. obtain a configuration of the system by randomly generating $3N$ Cartesian coordinates which are assigned to the particles,
2. calculate the potential energy of the configuration $V(\mathbf{r}^N)$,
3. from the potential energy, calculate the Boltzmann factor, $\exp(-V(\mathbf{r}^N)/k_B T)$,
4. add the Boltzmann factor to the accumulated sum of Boltzmann factors and the potential energy contribution to its accumulated sum and return to step 1,
5. and after a number, N_{trial} of iterations, the mean value of the potential energy would be obtained by:

$$\langle V(\mathbf{r}^N) \rangle = \frac{\sum_{i=1}^{N_{\text{trial}}} V_i(\mathbf{r}^N) \exp[-V_i(\mathbf{r}^N)/k_B T]}{\sum_{i=1}^{N_{\text{trial}}} \exp[-V_i(\mathbf{r}^N)/k_B T]} \quad (10)$$

that as argued, is not a feasible approach due to the large number of configurations that have extremely small Boltzmann factors.

2. Metropolis Monte Carlo

One way around the integration problem above is to generate only those configurations that make a

large contribution to the integral in equation 10, which is the strategy adopted in the Metropolis MC, also called importance sampling method. These configurations are generated by using random numbers drawn from a specific distribution function $\rho(x)$ in equation 3. For many thermodynamic properties of a molecular system, those states with a high probability are also the ones that make a significant contribution to the integral. The crucial feature of the Metropolis method is that it biases the generation of configurations towards those that make the most significant contribution to the integral. Specifically, it generates states with a probability $\rho(\mathbf{r}^N) = \exp(-V(\mathbf{r}^N)/k_B T)$ and then counts each of them equally. By contrast, the simple MC integration method generates states with equal probability (both high- and low-energy) and then assigns them a weight $\exp(-V(\mathbf{r}^N)/k_B T)$.

For example, in figure V.5 we compare two ways to measure the depth of the Nile with the simple MC on the left with Metropolis on the right. In the simple MC the value of the integrand is measured at a predetermined set of points. As the choice of these points does not depend on the value of the integrand, many points may be located in regions where the integrand vanishes. In contrast, in the Metropolis scheme, a *random walk* is constructed through that region of space where the integrand is non negligible (i.e., through the Nile itself). In this random walk, a trial move is rejected if it takes you out of the water and is accepted otherwise.

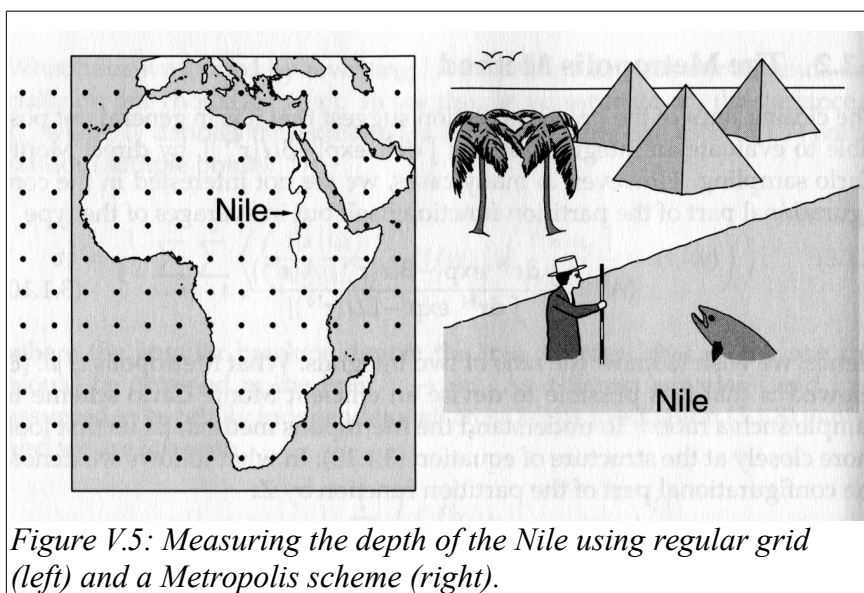


Figure V.5: Measuring the depth of the Nile using regular grid (left) and a Metropolis scheme (right).

After every trial move (accepted or not), the depth of the water is measured. The (unweighted) average of all these measurements yields an estimate of the average depth of the Nile. This, then is the essence of the Metropolis method.

In more depth, in the Metropolis MC the random walk along phase space has to generate a sequence of random states so that by the end of the simulation each state has occurred with the appropriate probability. The solution is to setup this random walk as a Markov chain of states of the liquid, which is constructed so that it has a limiting distribution of $\rho(\mathbf{r}^N)$. A Markov chain is a sequence of trials that satisfies the following two conditions.

1. The outcome of each trial belongs to a finite set of outcomes, $\{\Gamma_1, \Gamma_2, \dots, \Gamma_m, \Gamma_n, \dots\}$ called the state space.

2. 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 . Given then the probability of state Γ_m to be ρ_m , then the probability to go to state Γ_n is given by $\rho_m \pi_{mn}$.

A useful trick to determine the form of the transition probability is to assume *microscopic reversibility* in that

$$\rho_m \pi_{mn} = \rho_n \pi_{nm} . \quad (11)$$

A suitable scheme for the construction of a phase space trajectory in the canonical ensemble involves choosing a transition matrix that in addition to equation 11 it also satisfies

$$\sum_n \pi_{mn} = 1 . \quad (12)$$

A solution considers two cases,

$$\begin{aligned} \pi_{mn} &= \alpha_{mn} & \rho_n \geq \rho_m & \quad m \neq n \\ \pi_{mn} &= \alpha_{mn}(\rho_n/\rho_m) & \rho_n < \rho_m & \quad m \neq n . \end{aligned} \quad (13)$$

It can be shown that there is considerable freedom on choosing α , and that in our case for a MC move that decreases the energy the move would be accepted with probability 1. If the move would increase the energy, then the move is accepted with probability

$$\frac{\rho_n}{\rho_m} = \frac{\exp(-\beta V_n)/Z}{\exp(-\beta V_m)/Z} = \frac{\exp(-\beta V_n) \exp(-\beta \delta V_{mn})}{\exp(-\beta V_m)} = \exp(-\beta \delta V_{nm}) \quad (14)$$

where V_{nm} is the difference in potential energy between state n and state m .

Note that the transition probability is both, statistically efficient in allowing transitions, and because simulations are of finite length, it allows the Markov chain to sample a representative portion of phase space in a reasonable number of moves.

2.1. Implementation

The implementation of a Metropolis MC is quite simple. At each iteration of the simulation a new configuration is generated. This is usually done by making a random change to the Cartesian coordinates of a single randomly chosen particle using a random number generator. The new positions are given by

$$\begin{aligned} x_{new} &= x_{old} + (2\zeta - 1)\delta r_{max} \\ y_{new} &= y_{old} + (2\zeta - 1)\delta r_{max} \\ z_{new} &= z_{old} + (2\zeta - 1)\delta r_{max} \end{aligned} \quad (15)$$

where ζ are drawn from a uniform random distribution in the $[0,1]$ range. The energy of the new

configuration is then calculated. If the new configuration is lower in energy than its predecessor then the new configuration is retained as the starting point for the next iteration. If the new configuration is higher in energy than its predecessor, then the Boltzmann factor is compared to a random number between 0 and 1. If the Boltzmann factor is greater than the random number, then the new configuration is accepted. If not, then it is rejected and the initial configuration is retained for the next move. This acceptance condition can be written in the following concise fashion:

$$\text{rand}(0, 1) \leq \exp(-\Delta V(\mathbf{r}^N)/k_B T) \quad (16)$$

and illustrated in Figure V.6. The size of the move at each iteration is governed by the maximum displacement, δr_{\max} . This is an adjustable parameter whose value is usually chosen so that approximately 50% of the trial moves are accepted. If the maximum displacement is too small, then many moves will be accepted but the states will be very similar and the phase space will only be

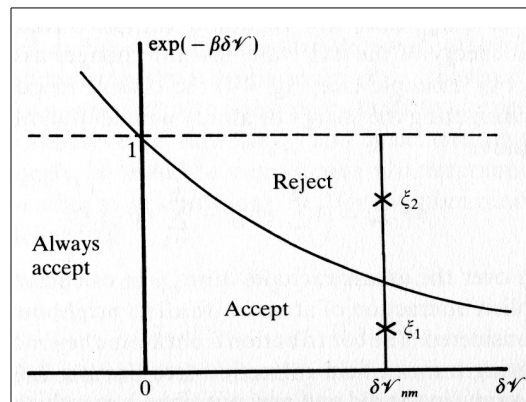


Figure V.6: Acceptance and rejection graph for MC moves.

explored very slowly. Too large a value and many trial moves will be rejected because they lead to unfavorable overlaps. The maximum displacement can be adjusted automatically while the program is running to achieve the desired acceptance ratio by keeping a running score of the proportion of moves that are accepted. Every so often, the maximum displacement is then scaled by a few percent: if too many moves have been accepted then the maximum displacement is increased; too few and it is reduced.

2.2. Tips

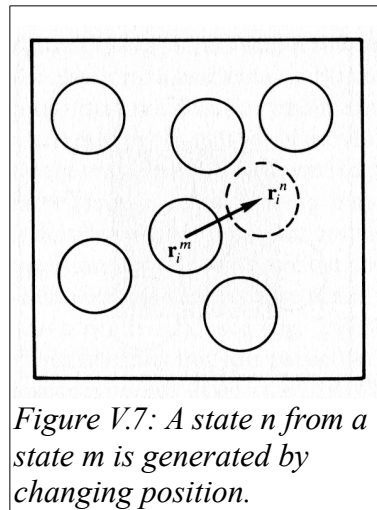
When generating a new state from the previous state, as illustrated in Figure V.7, the calculation of the energy need not require a complete recalculation of the energy of the entire system but only those contributions involving the particle that has just been moved. For example, the change in potential energy is calculated by computing the energy of atom i with all the other atoms before and after the move

$$\Delta V_{nm} = \left(\sum_{j=1}^N V(r_{ij}^n) - \sum_{j=1}^N V(r_{ij}^m) \right) \quad (17)$$

where the sum over the atoms excludes atom i . As a consequence, the neighbor list used by a MC simulation must contain all the neighbors of each atom because it is necessary to identify *all* the atoms which interact with the moving atom. For short-range potentials, however, only the explicit interaction of atom i with all its neighbors out to a cutoff distance r_c may be considered. The contribution from atoms beyond the cutoff could be estimated using a mean field correction, but in fact the correction for atom i in the old and new positions is exactly the same and do not need to be included explicitly in the calculation of ΔV_{nm} .

Also, proper account should be taken of periodic boundary conditions and the minimum image convention when generating new configurations and calculating energies.

As an alternative to the random selection of particles for a move, it is possible to move the atoms sequentially (this requires one fewer call to the random number generator per iteration). In addition, several atoms can be moved at once; if an appropriate value for the maximum displacement is chosen then this may enable phase space to be covered more efficiently.



As with a typical MD simulation, a MC simulation comprises an equilibration phase followed by a production phase. During equilibration, appropriate thermodynamic and structural quantities such as the total energy, the partitioning of the energy among the various components, mean square displacement and order parameters, are monitored until they achieve stable values.

2.3. Other Ensembles

Canonical (constant-NVT) Ensemble

This ensemble is demonstrated in the previous sections, but as a review, the partition function is given by

$$Q(N, V, T) \equiv \frac{1}{\Lambda^{3N} N!} \int d\mathbf{r}^N \exp[-\beta U(\mathbf{r}^N)] \quad (18)$$

where $\Lambda = \sqrt{h^2/(2\pi m k_B T)}$ is the thermal de Broglie wavelength. From the partition function it follows that the probability of finding configuration \mathbf{r}^N is given by

$$N(\mathbf{r}^N) \propto \exp[-\beta U(\mathbf{r}^N)] . \quad (19)$$

The implementation of this canonical ensemble has been shown above where the probability of accepting a move is given by

$$acc(o \rightarrow n) = \min(1, \exp\{-\beta[U(n) - U(o)]\}) . \quad (20)$$

Isobaric-Isothermal (constant-NPT) Ensemble

The NPT ensemble is widely used in Monte Carlo simulations. This is not surprising because most real experiments are also carried out under conditions of controlled pressure and temperature. Moreover, NPT simulations can be used to measure the equation of state of a model system even if the virial expression for the pressure cannot be readily evaluated. Finally, it is often convenient to use NPT MC to simulate systems in the vicinity of a first-order phase transition, because at constant pressure the system is free (given enough time) to transform completely into the state of lower (Gibbs) free energy, whereas in an NVT simulation the system may be kept at a density where it would like to phase separate into two bulk phases of different density but is prevented from doing so by finite-size effects.

The partition function for the NPT ensemble is given by

$$Q(N, P, T) \equiv \frac{\beta P}{\Lambda^{3N} N!} \int dV V^N \exp(-\beta P V) \int d\mathbf{s}^N \exp[-\beta U(\mathbf{s}^N; L)] , \quad (21)$$

where $\mathbf{s} = L^{-1} \cdot \mathbf{r}$. The probability for finding configuration \mathbf{r}^N is given by

$$\begin{aligned} N(V; \mathbf{s}^N) &\propto V^N \exp(-\beta P V) \exp[-\beta U(\mathbf{s}^N; L)] \\ &= \exp\{-\beta[U(\mathbf{s}^N, V) + P V - N\beta^{-1} \ln V]\} \end{aligned} \quad (22)$$

Implementation for the MC algorithms follows:

$$\begin{aligned} acc(o \rightarrow n) &= \min(1, \exp\{-\beta[U(\mathbf{s}^N, V') - U(\mathbf{s}^N, V)] \\ &\quad + P(V' - V) - N\beta^{-1} \ln(V'/V)\}) \end{aligned} \quad (23)$$

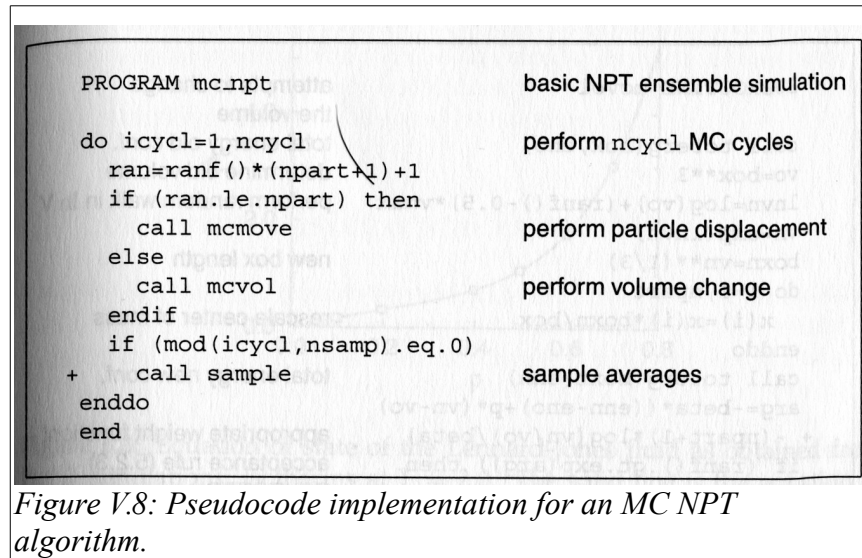
The algorithmic implementation of the NPT ensemble is in many ways very similar to that for the NVT

ensemble. The volume V is treated just as an additional coordinate where trial 'moves' are generated in much the same way as for trial positions. Trial moves

$$V' = V + \Delta V \quad (24)$$

involve calculating the change in volume by $\Delta V = (\xi - 0.5) * \delta V_{max}$ where ξ is a uniformly distributed random number and δV_{max} is the maximum step allowed for the volume to change. Because changing the volume involves recalculating all pairwise interaction (order N^2) usually the volume is changed once for every MC step over all particles.

A pseudo code for an NPT MC implementation is shown in Figure V.8. Note that a regular update of the volume should not be periodically done, but randomly chosen with probability $1/N$.



It has been shown that instead of performing moves in V better results are acquired when updating $\ln V$ instead. Thus updates of the form

$$\ln V' = \ln V_o + (\xi - 0.5) * v_{max} \quad (25)$$

are done instead.

Interaction potentials that depend on powers of distance have an additional advantage when updating the volume. Because $V = L^3$ (for a cubic box), then changes in volume involve changes in the size of the box side. Then for

$$\begin{aligned} U_n &= \sum_{i < j} \varepsilon \left(\frac{\sigma}{r_{ij}} \right)^n \\ &= \sum_{i < j} \varepsilon \left(\frac{\sigma}{L s_{ij}} \right)^n \end{aligned} \quad (26)$$

where $\mathbf{r}_i = L \mathbf{s}_i$. Thus

$$U_n(L') = \left(\frac{L}{L'}\right)^n U_n(L) . \quad (27)$$

Because this volume update is so computationally “cheap” to perform more updates per MC step can be allowed.

3. *MC vs MD*

Sometimes the decision on which technique to use is a trivial one, for example because a suitable program is readily available. In other cases there are clear reasons for choosing one method instead of the other. For example, MD is required if one wishes to calculate time-dependent quantities such as transport coefficients. Conversely, MC is often the most appropriate method to investigate systems in certain ensembles, for example, it is much easier to perform simulations at exact temperatures and pressures with the MC method than using the sometimes awkward and ill-defined constant temperature and constant pressure MD simulation methods. The MC method is also well suited to certain types of models such as the lattice models.

The two methods can differ in their ability to explore phase space. A MC simulation often gives much more rapid convergence of the calculated thermodynamic properties of a simple molecular liquid, but it may explore the phase space of large molecules very slowly due to the need for small steps unless special techniques are employed. However, the ability of the MC method to make non-physical moves can significantly enhance its capacity to explore phase space in appropriate cases. This may arise for simulations of isolated molecules, where there are a number of minimum energy states separated by high barriers. MD may not be able to cross the barriers between the conformations sufficiently often to ensure that each conformation is sampled according to the correct statistical weight. MD advances the positions and velocities of all the particles simultaneously and it can be very useful for exploration of the local phase space whereas the MC method may be more effective for conformational changes, which jump to a completely different area of phase space.

4. *Molecular Liquids*

The MC method is most easily implemented for atomic systems because it is only necessary to consider the translational degrees of freedom. The algorithm is easy to implement and accurate results can be obtained from relatively short simulations of a few tens of thousands of steps. There can be practical problems in applying the method to molecular system, and especially to molecules which have a significant degree of conformational flexibility. This is because, in such systems, it is necessary to permit the internal degrees of freedom to vary. Unfortunately, such changes often lead to high-energy overlaps either within the molecule or between the molecule and its neighbors and thus a high rejection rate. However, let's first visit the method for molecules that do not allow changes in their internal degrees of freedom. Under this circumstance, MC is very convenient for this kinds of simulations.

4.1. Rigid Molecules

For rigid, non-spherical molecules, the orientations of the molecules must be varied as well as their positions, in space. It is usual to translate and rotate one molecule during each MC step. Translations are usually described in terms of the position of the center of mass. There are various ways to generate a new orientation of a molecule. The simplest approach is to choose one of the three Cartesian axes and to rotate about the chosen axis by a randomly chosen angle $\delta\omega$, chosen to lie within the maximum angle variation, $\delta\omega_{\max}$. The rotation is achieved by applying the usual trigonometric relationships. For example, if the vector $(x\mathbf{i}, y\mathbf{j}, z\mathbf{k})$ describes the orientation of a molecule, then the new vector $(x'\mathbf{i}, y'\mathbf{j}, z'\mathbf{k})$ that corresponds to rotation by $\delta\omega$ about the x axis is given by:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\delta\omega & \sin\delta\omega \\ 0 & -\sin\delta\omega & \cos\delta\omega \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}. \quad (28)$$

The *Euler angles* are often used to describe the orientations of a molecule. There are three; ϕ , θ , and ψ , as shown in Figure V.9. If the Euler angles are randomly changed by small amounts $\delta\phi$, $\delta\theta$, and $\delta\psi$ then a vector \mathbf{v}_{old} is moved according to the following matrix equation:

$$\mathbf{v}_{\text{new}} = \mathbf{A}\mathbf{v}_{\text{old}} \quad (29)$$

where the matrix \mathbf{A} is

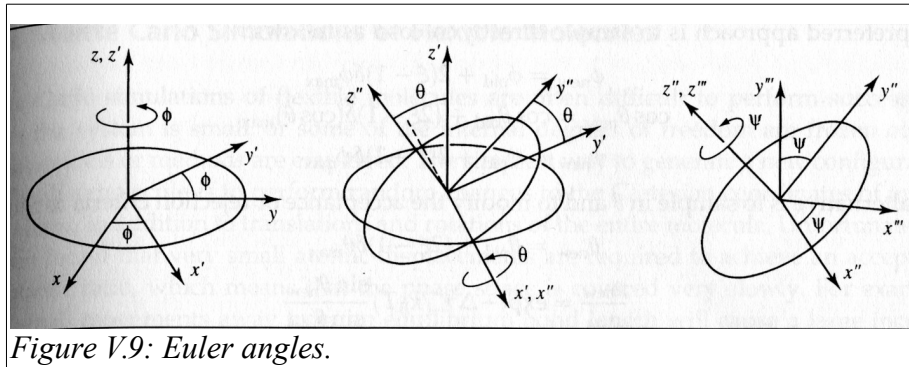


Figure V.9: Euler angles.

$$\begin{pmatrix} \cos\delta\phi\cos\delta\psi - \sin\delta\phi\cos\delta\theta\sin\delta\psi & \sin\delta\phi\cos\delta\psi + \cos\delta\phi\cos\delta\theta\sin\delta\psi & \sin\delta\theta\sin\delta\psi \\ -\cos\delta\phi\sin\delta\psi - \sin\delta\phi\cos\delta\theta\cos\delta\psi & -\sin\delta\phi\sin\delta\psi + \cos\delta\phi\cos\delta\theta\cos\delta\psi & \sin\delta\theta\cos\delta\psi \\ \sin\delta\phi\sin\delta\theta & -\cos\delta\phi\sin\delta\theta & \cos\delta\theta \end{pmatrix} \quad (30)$$

It is important to note that simply sampling displacements of the three Euler angles does not lead to a uniform distribution; it is necessary to sample from $\cos\theta$ rather than θ (Figure V.10).

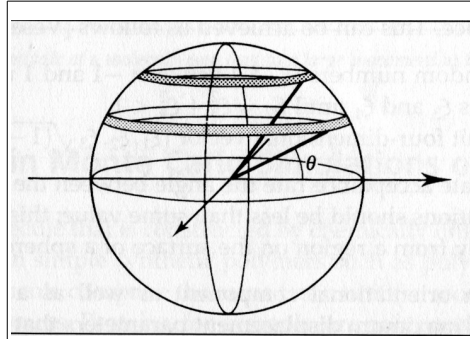


Figure V.10: If the sampling is uniform in θ then the number of points per unit area increases with θ , leading to an uneven distribution over the sphere.

The preferred approach is to sample directly in $\cos\theta$ as follows:

$$\begin{aligned}\phi_{new} &= \phi_{old} + (2\xi - 1)\delta\phi_{max} \\ \cos \theta_{new} &= \cos \theta_{old} + (2\xi - 1)\delta(\cos \theta)_{max} \cdot \\ \psi_{new} &= \psi_{old} + (2\xi - 1)\delta\psi_{max}\end{aligned}\tag{31}$$

But if one still wants to sample in θ , then one has to modify the acceptance or rejection criteria as follows:

$$\begin{aligned}\theta_{new} &= \theta_{old} + (2\xi - 1)\delta\theta_{max} \\ \frac{\rho_{new}}{\rho_{old}} &= \exp(-\Delta V/k_B T) \frac{\sin \theta_{new}}{\sin \theta_{old}}.\end{aligned}\tag{32}$$

This second approach may give problems if θ_{old} equals zero.

A disadvantage of the Euler angle approach is that the rotation matrix contains a total of six trigonometric functions. These trigonometric functions are computationally expensive to calculate. An alternative is to use *quaternions*. A quaternion is a four-dimensional vector such that its components sum to 1: $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. The quaternion components are related to the Euler angles as follows:

$$\begin{aligned}q_0 &= \cos \frac{1}{2}\theta \cos \frac{1}{2}(\phi + \psi) \\ q_1 &= \sin \frac{1}{2}\theta \cos \frac{1}{2}(\phi + \psi) \\ q_2 &= \sin \frac{1}{2}\theta \sin \frac{1}{2}(\phi + \psi) \\ q_3 &= \cos \frac{1}{2}\theta \sin \frac{1}{2}(\phi + \psi)\end{aligned}\tag{33}$$

The Euler angle rotation matrix can then be written

$$\mathbf{A} = \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 + q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 + q_0 q_1) \\ 2(q_1 q_3 + q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix}. \quad (34)$$

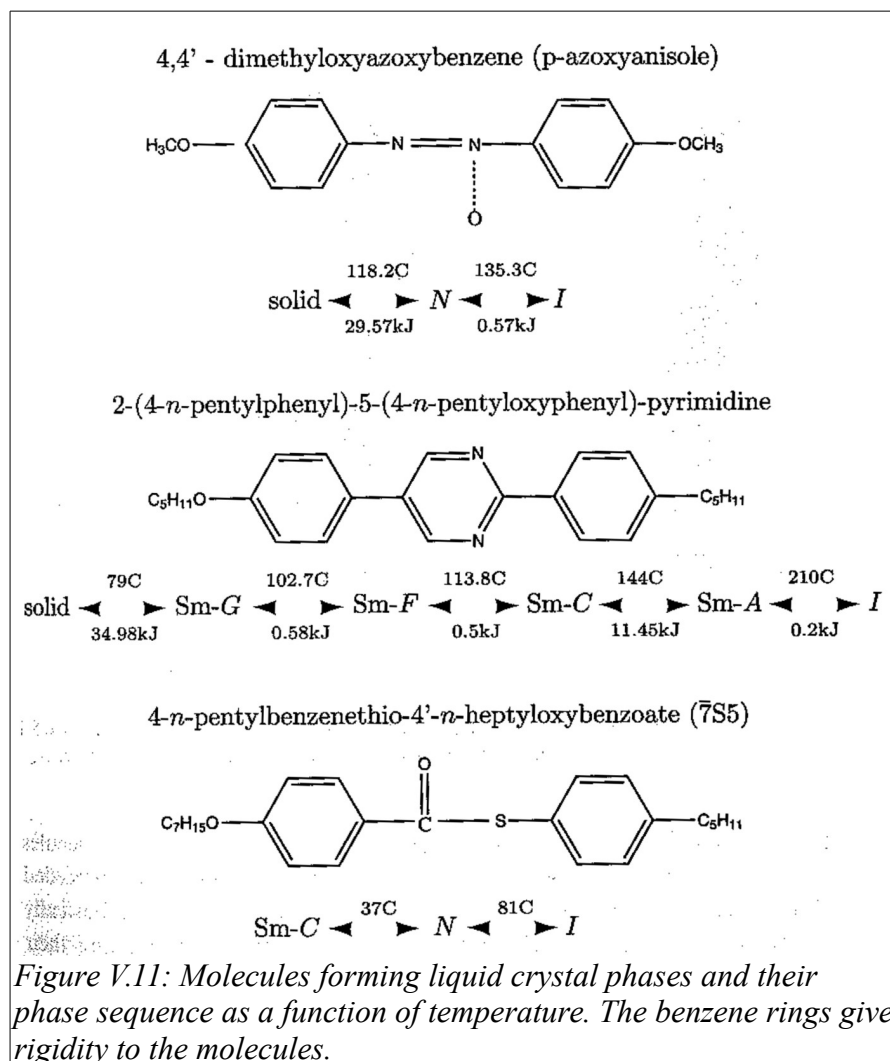
To generate a new quaternion orientation, it is necessary to rotate the quaternion vector to a new (random) orientation. As it is a four-dimensional vector, the orientation must be performed in four-dimensional space. This can be achieved as follows.

1. Generate pairs of random number (ξ_1, ξ_2) between -1 and 1 until $S_1 = \xi_1^2 + \xi_2^2 < 1$.
2. Do the same for pairs ξ_3 and ξ_4 until $S_2 = \xi_3^2 + \xi_4^2 < 1$.
3. Form the random unit four-dimensional vector $(\xi_1, \xi_2, \xi_3 \sqrt{((1-S_1)/S_2)}, \xi_4 \sqrt{((1-S_1)/S_2)})$.

To achieve an appropriate acceptance rate the angle between the two vectors that describe the new and old orientations should be less than some value; this corresponds to sampling randomly and uniformly from a region on the surface of a sphere.

Liquid Crystals

An area where these kinds of models can be applied is the area of liquid crystals. Liquid crystals are able to form phases that are characterized by a long-range order of the molecular orientations in at least one dimension while their centers of mass are (to some extent) disordered and transitionally mobile. Many of the molecules that exhibit liquid crystalline behavior are rod-shaped, but disk-like molecules can also form liquid crystalline phases. The simplest phase formed by liquid crystals is the nematic phase. In the nematic phase the rod-shaped molecules (see examples in Figure V.11) are aligned with their long axes pointing in approximately the same direction while the molecular centers are completely translationally disordered (Figure V.12(b)).



Molecules forming liquid crystals are generally quite anisometric. Most of the early simulation work on liquids concentrated on atoms and diatomic molecules, using site-site potentials. But a substantial number of site-site potentials would be required to model elongated molecules. The complexity of an example liquid crystal is shown in Figure V.13(a). For example, a simple nematogen such as quinquaphenyl would require five touching spheres to give the correct length-to-width ratio. To circumvent this, specific orientation-dependent interactions have been introduced. Hard ellipsoid models (Figure V.13(b)) carry the problems that calculation of overlaps are computationally very expensive. Cheaper models use spherocylinders (a cylinder with hemispherical caps at the ends). For soft potentials, an 'ellipsoidal' generalization of the Lennard-Jones potential has been worked out where the ϵ and σ parameters are now dependent on molecular orientations.

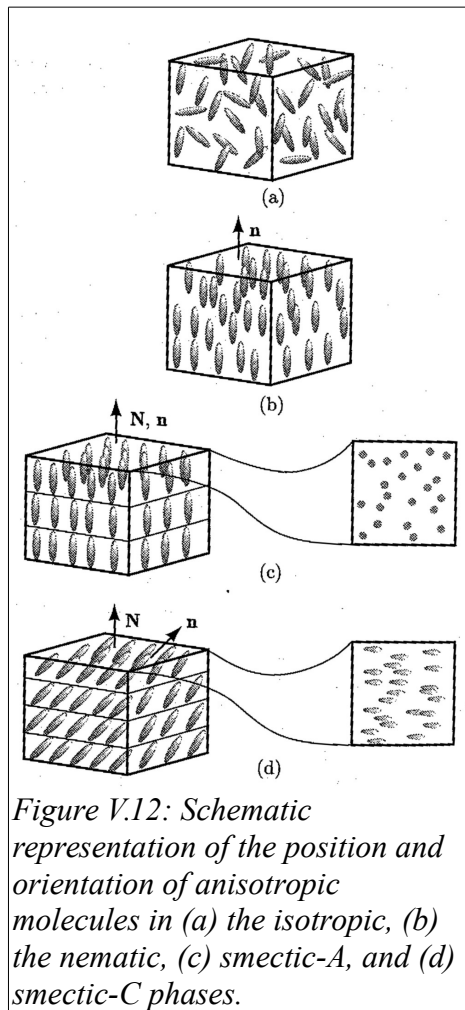


Figure V.12: Schematic representation of the position and orientation of anisotropic molecules in (a) the isotropic, (b) the nematic, (c) smectic-A, and (d) smectic-C phases.

Early attempts to simulate a liquid crystal whether using hard or soft potentials, had some discouraging features in common. It proved to be easy to generate an orientationally ordered liquid by melting a crystal, or by imposing an external field. The orientation would persist on and after equilibrating in the absence of the fields for long times, but eventually, in almost every case, orientational order would disappear. In no case, in this early work, was spontaneous ordering observed on cooling or compressing the isotropic liquid.

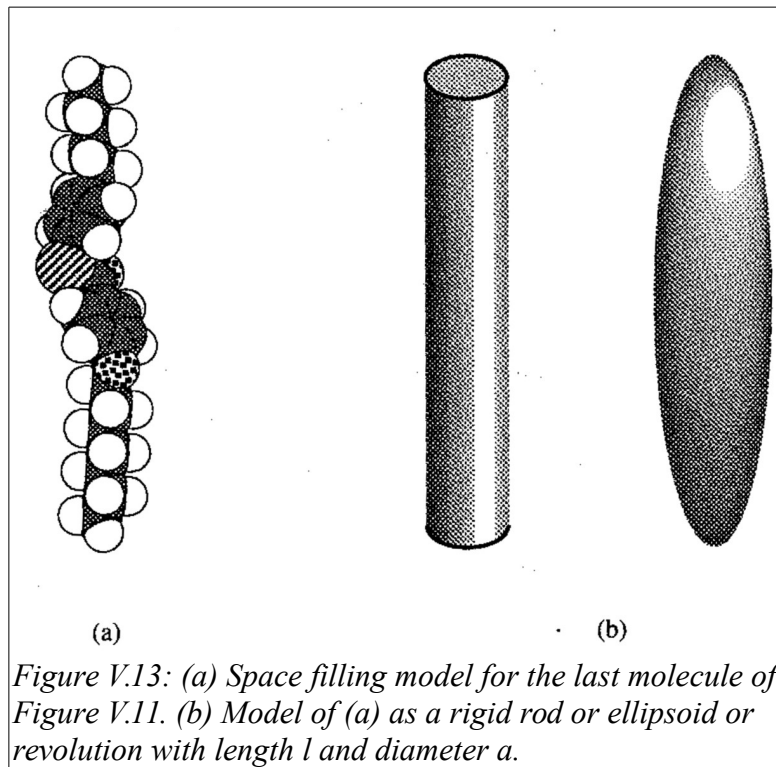


Figure V.13: (a) Space filling model for the last molecule of Figure V.11. (b) Model of (a) as a rigid rod or ellipsoid or revolution with length l and diameter a .

Accordingly, simulations of orientationally ordered phases have been mainly restricted to lattice systems, with the molecular center of mass being perfectly ordered. Typically, these simulations involve nearest-neighbor interactions using the Lebwohl-Lasher potential

$$v(r_{ij}, \Omega_i, \Omega_j) = AP_2(\cos\gamma_{ij}) \quad (35)$$

where γ_{ij} is the angle between molecules i and j . A is a negative constant, and P_2 a Legendre polynomial. This system shows an orientational phase transition.

Recently, the lattice restriction has been removed and by using a constant NPT Monte Carlo method a system of Lennard-Jones atoms with an orientational potential of the form of equation 35 has been realized. If for example the angle-dependent potential was introduced into an already-equilibrated Lennard-Jones liquid system, the orientational order-disorder transition could be observed.

4.2. Flexible Polymers

A polymer is a macromolecule that is constructed by chemically linking together a sequence of molecular fragments. In simple synthetic polymers such as polyethylene or polystyrene all of the molecular fragments comprise the same basic unit (or monomer). Other polymers contain mixtures of monomers. Proteins, for example, are polypeptide chains in which each unit is one of the twenty amino acids. Cross-linking between different chains gives rise to yet further variations in the constitution and structure of a polymer. All of these features may affect the overall properties of the molecule,

sometimes in a dramatic way. Moreover, one may be interested in the properties of the polymer under different conditions, such as in solution, in a polymer melt or in the crystalline state. Molecular modeling can help to develop theories for understanding the properties of polymers and can also be used to predict their properties.

A wide range of time and length scales are needed to completely describe a polymer's behavior. The timescale ranges from approximately 10^{-14} s (i.e. the period of a bond vibration) through to seconds, hours or even longer for collective phenomena. The size scale ranges from the 1-2 Å of chemical bonds to the diameter of a coiled polymer, which can be several hundreds of angstroms. Many kinds of models have been used to represent and simulate polymeric systems and predict their properties. Some of these models are based upon very simple ideas about the nature of the intra- and intermolecular interactions within the system but have nevertheless proved to be extremely useful.

Most simulations on polymers are performed using empirical energy models (though faster computers are allowing to apply quantum mechanics to larger and larger systems). Moreover, there are various ways in which the configurational and conformational degrees of freedom may be restricted so as to produce a computationally more efficient model. The simplest models use a lattice representation in which the polymer is constructed from connected interaction centers, which are required to occupy the vertices of a lattice. At the next level of complexity are the bead models, where the polymer is composed of a sequence of connected 'beads'. Each bead represents an 'effective monomer' and interacts with the other beads to which it is bonded and also with other nearby beads. The ultimate level of detail is achieved with the atomistic models, in which each non-hydrogen atom is explicitly represented (and sometimes all of the hydrogens as well). In the following, we briefly present some of these techniques in which Monte Carlo methods can be used to investigate polymeric systems.

Lattice Models of Polymers

Lattice models have provided many insights into the behavior of polymers despite the obvious approximations involved. The simplicity of a lattice model means that many states can be generated and examined very rapidly. Both two-dimensional and three-dimensional lattices are used. The simplest models use cubic or tetrahedral lattices in which successive monomers occupy adjacent lattice points, as shown in Figure V.14. The energy models are usually very simple, in part to reflect the simplicity of the representation but also to permit the rapid calculation of the energy.

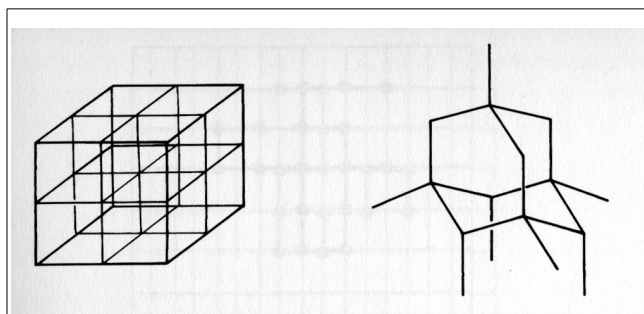


Figure V.14: Cubic and tetrahedral (diamond) lattices, which are commonly used for lattice simulations of polymers.

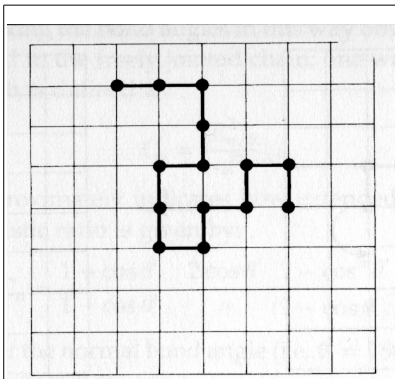


Figure V.16: Self-avoiding walk: only one monomer can occupy each lattice site.

Self-avoiding walks have been used to exhaustively enumerate all possible conformations for a chain of a given length on the lattice. If all states are known then the partition function can be determined and thermodynamic quantities calculated. The 'energy' of each state may be calculated using an appropriate interaction model. For example, the energy may be proportional to the number of adjacent pairs of occupied lattice sites. A variation on this is to use polymers consisting of two types of monomer (A and B), which have up to three different energy values: A-A, B-B, and A-B. Again, the energy is determined by counting the number of occupied adjacent lattice sites. The relationship between the mean square end-to-end distance and the length of the chain (n) has been investigated intensively. With the self-avoiding walk the result obtained is different from the random walk, with $\langle R_n^2 \rangle$ being proportional to $n^{1.18}$ in the asymptotic limit.

Having grown a polymer onto the lattice, we now have to consider the generation of alternative configurations. Motion of the entire polymer chain or large-scale conformational changes are often difficult, especially for densely packed polymers. In a version of the Verdier-Stockmayer algorithm new configurations are generated using combinations of 'crankshaft', kink jump; ' and 'end rotation' moves (Figure V.17).

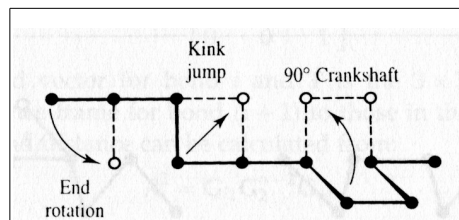
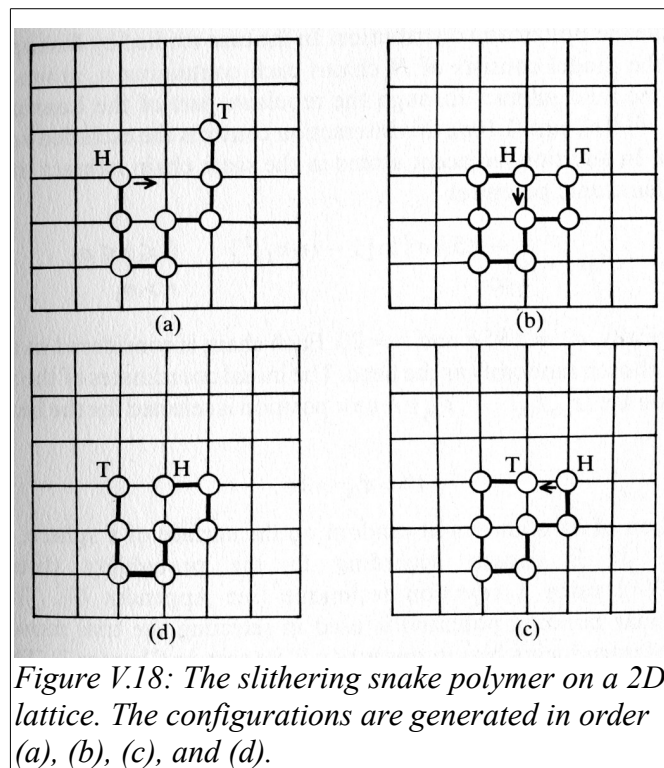


Figure V.17: 'crankshaft', 'kink jump' and 'end rotation' moves used in MC simulations of polymers.

Another widely used algorithm in MC simulations of polymers (not just in lattice models) is the 'slithering snake' model. Motion of the entire polymer chain is very difficult, especially for densely packed polymers, and one way in which the polymer can move is by wriggling around obstacles, a process known as *reptation*. To implement a slithering snake algorithm, one end of the polymer chain is randomly chosen as the 'head' and an attempt is made to grow a new bead at one of the available

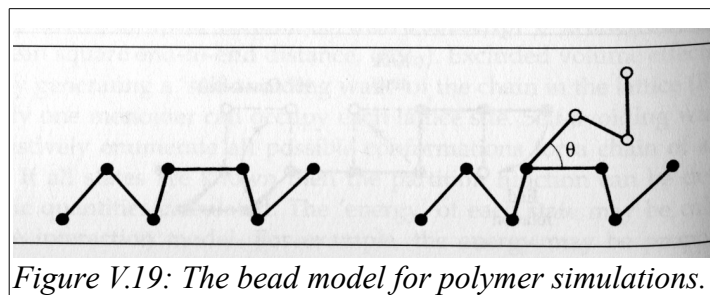
adjacent lattice positions. Each of the remaining beads is then advanced to that of its predecessor in the chain as illustrated in Figure V.18.



The procedure is then repeated. Even if it is impossible to move the chosen 'head', the configuration must still be included when ensemble averages are calculated.

Continuous Polymer Models

The simplest of the continuous polymer models consists of a string of connected beads (Figure V.19). The beads are freely jointed and interact with the other beads via a spherically symmetric potential such as the Lennard-Jones potential. The beads should not be thought of as being identical to the monomers in the polymer, though they are often referred to as such. Similarly, the links between the beads should not be thought of as bonds. The links may be modeled as rods of a fixed and invariant length or may be permitted to vary using a harmonic potential function.

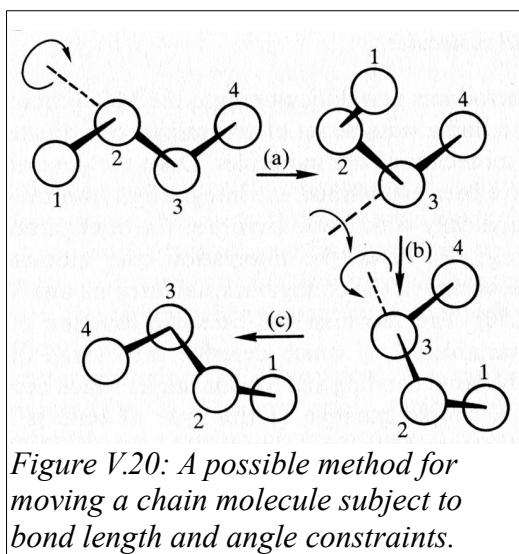


In MC studies with this freely jointed chain model the beads can sample from a continuum of positions. The *pivot algorithm* is one way that new configurations can be generated. Here, a segment of the

polymer is randomly selected and rotated by a random amount, as illustrated in Figure V.19. For isolated polymer chains the pivot algorithm can give a good sampling of the configurations/conformational space. However, for polymers in solution or in the melt, the proportion of accepted moves is often very small due to high-energy steric interactions.

The most unrealistic feature of the freely jointed chain model is the assumption that the bond angles can vary continuously. In the *freely rotating chain model* the bond angles are held fixed but free rotation is possible about the bonds, such that any torsion angle value between 0 and 360 is equally likely.

One way to deal with these molecules that contain a mix of constraints and freely moving degrees of freedom is to choose trial moves that change the free coordinates while at the same time keeping the constraints on the others. For example, in Figure V.20 we show a molecule of butane with bond constraints and free angle rotations.



A typical MC sequence might be:

- (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 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 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.

MC of other more complex flexible molecules are often difficult. The simplest way to generate a new configuration is to perform a random change to the Cartesian coordinates of the individual atoms, in addition to translations and rotations of the entire molecule. Unfortunately, it is often found that very

small atomic displacements are required to achieve an acceptable acceptance ratio, which means that the phase space is covered very slowly. The “trick” of freezing out some of the internal degrees of freedom, usually the 'hard' degrees of freedom such as the bond lengths and the bond angles, as shown above for butane only works for small molecules. For large molecules even relatively small bond rotations may cause large movements of atoms down the chain. This invariably leads to high-energy configurations as illustrated in Figure V.21.

