depends upon the way in which the atomic position $\mathbf{r}(t)$ changes with time. At a time t the difference between $\mathbf{r}(t)$ and $\mathbf{r}(0)$ is given by:

$$|\mathbf{r}(t) - \mathbf{r}(0)| = \int_0^t \mathbf{v}(t')\, dt' \tag{7.90}$$

If both sides of Equation (7.90) are now squared then we obtain the following for the mean-square value:

$$\langle |\mathbf{r}(t) - \mathbf{r}(0)| \rangle^2 = \int_0^t dt' \int_0^t dt'' \langle \mathbf{v}(t') \cdot \mathbf{v}(t'') \rangle \tag{7.91}$$

The crucial feature to recognise is that the relevant correlation functions are unaffected by changing the origin, which means that the following holds:

$$\langle \mathbf{v}(t') \cdot \mathbf{v}(t'') \rangle = \langle \mathbf{v}(t'' - t') \cdot \mathbf{v}(0) \rangle \tag{7.92}$$

Integration of the double integral, Equation (7.91) leads to the *Green–Kubo* formula:

$$\frac{\langle |\mathbf{r}(t) - \mathbf{r}(0)|^2 \rangle}{2t} = \int_0^t \langle \mathbf{v}(\tau) \cdot \mathbf{v}(0) \rangle \left(1 - \frac{\tau}{t}\right) d\tau \tag{7.93}$$

In the limit:

$$\int_0^\infty \langle \mathbf{v}(\tau) \cdot \mathbf{v}(0) \rangle\, d\tau = \lim_{t \to \infty} \frac{\langle |\mathbf{r}(t) - \mathbf{r}(0)|^2 \rangle}{2t} = 3D \tag{7.94}$$

We can now see why long time-tails in the autocorrelation functions are so important The area under the curve during the slow decay towards zero may be a significant part of the integral in the Green–Kubo formula. In practice, these integrals are determined numerically. The long time-tail may be dealt with by fitting a function to the curve and then attempting to integrate to infinity.

7.7 Molecular Dynamics at Constant Temperature and Pressure

Molecular dynamics is traditionally performed in the constant *NVE* (or *NVEP*) ensemble. Although thermodynamic results can be transformed between ensembles, this is strictly only possible in the limit of infinite system size ('the thermodynamic limit'). It may therefore be desired to perform the simulation in a different ensemble. The two most common alternative ensembles are the constant *NVT* and the constant *NPT* ensembles. In this section we will therefore consider how molecular dynamics simulations can be performed under conditions of constant temperature and/or constant pressure.

7.7.1 Constant Temperature Dynamics

There are several reasons why we might want to maintain or otherwise control the temperature during a molecular dynamics simulation. Even in a constant *NVE* simulation it is common practice to adjust the temperature to the desired value during the equilibration phase. A constant temperature simulation may be required if we wish to determine how

the behaviour of the system changes with temperature, such as the unfolding of a protein or glass formation. Simulated annealing algorithms require the temperature of the system to be reduced gradually while the system explores its degrees of freedom. Simulated annealing is used in searching conformational space and in the elucidation of macromolecular structure from NMR and X-ray data and is discussed in Section 9.9.2.

The temperature of the system is related to the time average of the kinetic energy, which for an unconstrained system is given by:

$$\langle \mathscr{K} \rangle_{NVT} = \tfrac{3}{2} N k_B T \tag{7.95}$$

An obvious way to alter the temperature of the system is thus to scale the velocities [Woodcock 1971]. If the temperature at time t is $T(t)$ and the velocities are multiplied by a factor λ, then the associated temperature change can be calculated as follows:

$$\Delta T = \frac{1}{2}\sum_{i=1}^{N} \frac{2}{3} \frac{m_i(\lambda v_i)^2}{N k_B} - \frac{1}{2}\sum_{i=1}^{N} \frac{2}{3} \frac{m_i v_i^2}{N k_B} \tag{7.96}$$

$$\Delta T = (\lambda^2 - 1)T(t) \tag{7.97}$$

$$\lambda = \sqrt{T_{new}/T(t)} \tag{7.98}$$

The simplest way to control the temperature is thus to multiply the velocities at each time step by the factor $\lambda = \sqrt{T_{req}/T_{curr}}$, where T_{curr} is the current temperature as calculated from the kinetic energy and T_{req} is the desired temperature.

An alternative way to maintain the temperature is to couple the system to an external heat bath that is fixed at the desired temperature [Berendsen *et al.* 1984]. The bath acts as a source of thermal energy, supplying or removing heat from the system as appropriate. The velocities are scaled at each step, such that the rate of change of temperature is proportional to the difference in temperature between the bath and the system:

$$\frac{dT(t)}{dt} = \frac{1}{\tau}(T_{bath} - T(t)) \tag{7.99}$$

τ is a coupling parameter whose magnitude determines how tightly the bath and the system are coupled together. This method gives an exponential decay of the system towards the desired temperature. The change in temperature between successive time steps is:

$$\Delta T = \frac{\delta t}{\tau}(T_{bath} - T(t)) \tag{7.100}$$

The scaling factor for the velocities is thus:

$$\lambda^2 = 1 + \frac{\delta t}{\tau}\left(\frac{T_{bath}}{T(t)} - 1\right) \tag{7.101}$$

If τ is large, then the coupling will be weak. If τ is small, the coupling will be strong and when the coupling parameter equals the time step ($\tau = \delta t$) then the algorithm is equivalent to the simple velocity scaling method. A coupling constant of approximately 0.4 ps has been suggested as an appropriate value to use when the time step is 1 fs, giving $\delta t/\tau \approx 0.0025$. The advantage of this approach is that it does permit the system to fluctuate about the desired temperature.

These two relatively simple temperature scaling methods do not generate rigorous canonical averages. Velocity scaling artificially prolongs any temperature differences among the components of the system, which can lead to the phenomenon of 'hot solvent, cold solute', in which the 'temperature' of the solute is lower than that of the solvent, even though the overall temperature of the system is at the desired value. One 'solution' to this problem is to apply temperature coupling separately to the solute and to the solvent but the problem of unequal distribution of energy between the various components (and between the various modes of motion) may still remain. Two methods that do generate rigorous canonical ensembles if properly implemented are the *stochastic collisions* method and the *extended system* method.

In the stochastic collisions method a particle is randomly chosen at intervals and its velocity is reassigned by random selection from the Maxwell–Boltzmann distribution [Anderson 1980]. This is equivalent to the system being in contact with a heat bath that randomly emits 'thermal particles' which collide with the atoms in the system. Between each collision the system is simulated at constant energy and so the overall effect is equivalent to a series of microcanonical simulations, each performed at a slightly different energy. The distribution of the energies of these 'mini microcanonical' simulations will be a Gaussian function. The stochastic collisions method does not, of course, generate a smooth trajectory, which may be a drawback. By calculating the energy change due to a collision, Anderson showed that the mean rate (ν) at which each particle should suffer a stochastic collision is given by:

$$\nu = \frac{2a\kappa}{3k_B \mathcal{N}^{1/3} N^{2/3}} \tag{7.102}$$

a is a dimensionless constant, κ is the thermal conductivity and $\mathcal{N}$ is the number density of the particles. If the thermal conductivity is not known then a suitable value of ν can be obtained from the intermolecular collision frequency ν_c:

$$\nu = \nu_c / N^{2/3} \tag{7.103}$$

If the collision rate is too low then the system does not sample from a canonical distribution of energies. If it is too high then the temperature control algorithm dominates and the system does not show the expected fluctuations in kinetic energy. The velocity of more than one particle can be changed in the stochastic collision method; in the limit the velocities of all the particles are changed simultaneously, though it is preferable to do this at quite long intervals. A distinction can thus be made between 'minor' collisions, in which only one (or a few) particles are affected, and 'major' (or 'massive') collisions, where the velocities of all particles are changed. It is also possible to use a combined approach, with minor collisions occurring relatively frequently and major collisions at longer intervals.

Extended system methods, originally introduced for performing constant temperature molecular dynamics by Nosé [Nosé 1984] and subsequently developed by Hoover [Hoover 1985], consider the thermal reservoir to be an integral part of the system. The reservoir is represented by an additional degree of freedom, labelled s. The reservoir has potential energy $(f+1)k_B T \ln s$, where f is the number of degrees of freedom in the physical system and T is the desired temperature. The reservoir also has kinetic energy $(Q/2)(ds/dt)^2$. Q is a parameter with the dimensions of energy $\times$ (time)^2 and can be considered the

(fictitious) mass of the extra degree of freedom. The magnitude of Q determines the coupling between the reservoir and the real system and so influences the temperature fluctuations.

Each state of the extended system that is generated by the molecular dynamics simulation corresponds to a unique state of the real system. There is not, however, a direct correspondence between the velocities and the time in the real and the extended systems The velocities of the atoms in the real system are given by:

$$\mathbf{v}_i = s\frac{d\mathbf{r}_i}{dt} \tag{7.104}$$

$\mathbf{r}_i$ is the position of particle i in the simulation and $\mathbf{v}_i$ is considered to be the real velocity of the particle. The time step $\delta t'$ is related to the time step in 'real time' δt by

$$\delta t = s\delta t' \tag{7.105}$$

The value of the additional degree of freedom s can change and so the time step in real time can fluctuate. Thus regular time intervals in the extended system correspond to a trajectory of the real system which is unevenly space in time.

The parameter Q controls the energy flow between the system and the reservoir. If Q is large then the energy flow is slow; in the limit of infinite Q, conventional molecular dynamics is regained and there is no energy exchange between the reservoir and the real system. However, if Q is too small then the energy oscillates, resulting in equilibration problems. Nosé has suggested that Q should be proportional to fk_BT; the constant of proportionality can then be obtained by performing a series of trial simulations for a test system and observing how well the system maintains the desired temperature.

7.7.2 Constant Pressure Dynamics

Just as one may wish to specify the temperature in a molecular dynamics simulation, so it may be desired to maintain the system at a constant pressure. This enables the behaviour of the system to be explored as a function of the pressure, enabling one to study phenomena such as the onset of pressure-induced phase transitions. Many experimental measurements are made under conditions of constant temperature and pressure, and so simulations in the isothermal–isobaric ensemble are most directly relevant to experimental data. Certain structural rearrangements may be achieved more easily in an isobaric simulation than in a simulation at constant volume. Constant pressure conditions may also be important when the number of particles in the system changes (as in some of the 'test particle' methods for calculating free energies and chemical potentials; see Section 8.9).

The pressure often fluctuates much more than quantities such as the total energy in a constant *NVE* molecular dynamics simulation. This is as expected because the pressure is related to the virial, which is obtained as the product of the positions and the derivative of the potential energy function. This product, $r_{ij}dV(r_{ij})/dr_{ij}$, changes more quickly with r than does the internal energy, hence the greater fluctuation in the pressure.

A macroscopic system maintains constant pressure by changing its volume. A simulation in the isothermal–isobaric ensemble also maintains constant pressure by changing the volume

of the simulation cell. The amount of volume fluctuation is related to the isothermal compressibility, κ.

$$\kappa = -\frac{1}{V}\left(\frac{\partial V}{\partial P}\right)_T \tag{7.106}$$

An easily compressible substance has a larger value of κ, so larger volume fluctuations occur at a given pressure than in a more incompressible substance. Conversely, in a constant volume simulation a less compressible substance shows larger fluctuations in the pressure. The isothermal compressibility is the pressure analogue of the heat capacity, which is related to the energy fluctuations.

A volume change in an isobaric simulation can be achieved by changing the volume in all directions, or in just one direction. It is instructive to consider the range of volume changes that one might expect to observe in a constant pressure simulation of a 'typical' system. The isothermal compressibility is related to the mean square volume displacement by:

$$\kappa = \frac{1}{k_B T}\frac{\langle V^2\rangle - \langle V\rangle^2}{\langle V^2\rangle} \tag{7.107}$$

The isothermal compressibility of an ideal gas is approximately 1 atm ^{1}. So for a simulation in a box of side 20 Å (volume 8000 Å^3) at 300 K, the root mean square change in the volume is approximately 18 100 Å^3. This is larger than the initial size of the box! For a relatively incompressible substance such as water ($\kappa = 44\ 75 \times 10^{-6}$ atm^{-1}) the fluctuation is 121 Å^3, which corresponds to the box only changing by about 0 1 Å in each direction. These values have clear implications for the appropriate size of the simulation system.

Many of the methods used for pressure control are analogous to those used for temperature control Thus, the pressure can be maintained at a constant value by simply scaling the volume. An alternative is to couple the system to a 'pressure bath', analogous to a temperature bath [Berendsen *et al.* 1984]. The rate of change of pressure is given by:

$$\frac{dP(t)}{dt} = \frac{1}{\tau_p}(P_{\text{bath}} - P(t)) \tag{7.108}$$

τ_p is the coupling constant, P_{bath} is the pressure of the 'bath', and $P(t)$ is the actual pressure at time t. The volume of the simulation box is scaled by a factor λ, which is equivalent to scaling the atomic coordinates by a factor $\lambda^{1/3}$. Thus:

$$\lambda = 1 - \kappa\frac{\delta t}{\tau_p}(P - P_{\text{bath}}) \tag{7.109}$$

The new positions are given by:

$$\mathbf{r}_i' = \lambda^{1/3}\mathbf{r}_i \tag{7 110}$$

The constant κ can be combined with the relaxation constant τ_p as a single constant. This expression can be applied isotropically (i.e. such that the scaling factor is equal for all three directions) or anisotropically (where the scaling factor is calculated independently for each of the three axes). In general, it is best to use the anisotropic approach as this enables the box dimensions to change independently. Unfortunately, it has not been possible to determine from which ensemble this method samples.

In the extended pressure-coupling system methods, first introduced by Anderson [Anderson 1980], an extra degree of freedom, corresponding to the volume of the box, is added to the system. The kinetic energy associated with this degree of freedom (which can be considered to be equivalent to a piston acting on the system) is $\frac{1}{2}Q(dV/dt)^2$, where Q is the 'mass' of the piston. The piston also has potential energy PV, where P is the desired pressure and V is the volume of the system. A piston of small mass gives rise to rapid oscillations in the box, whereas a large mass has the opposite effect. An infinite mass returns normal molecular dynamics behaviour. The volume can vary during the simulation, with the average volume being determined by the balance between the internal pressure of the system and the desired external pressure. The extended-system temperature-scaling method of Nosé uses a scaled time; in the extended pressure method the coordinates of the extended system are related to the 'real' coordinates by:

$$\mathbf{r}_i' = V^{-1/3}\mathbf{r}_i \tag{7.111}$$

7.8 Incorporating Solvent Effects into Molecular Dynamics: Potentials of Mean Force and Stochastic Dynamics

In many simulations of solute–solvent systems the primary focus is the behaviour of the solute; the solvent is of relatively little interest, particularly in regions far from the solute molecule. The use of non-rectangular periodic boundary conditions, stochastic boundaries and 'solvent shells' can all help to reduce the number of solvent molecules required and enable a larger proportion of the computing time to be spent simulating the solute In this section we consider a group of techniques that incorporate the effects of solvent without requiring any explicit specific solvent molecules to be present.

One approach to this problem is to use a *potential of mean force* (PMF), which describes how the free energy changes as a particular coordinate (such as the separation of two atoms or the torsion angle of a bond) is varied. The free energy change described by the potential of mean force includes the averaged effects of the solvent.

Potentials of mean force may be determined using a molecular dynamics or Monte Carlo simulation using the techniques of umbrella sampling or free energy perturbation, which will be discussed in Chapter 11. Here we illustrate the concept using an example The energy difference between the *trans* and *gauche* conformations for an isolated molecule of 1,2-dichloroethane (i.e. in the gas phase) is approximately 1.14 kcal mol^{-1} with a population containing 77% *trans* and 23% *gauche* conformers. In liquid 1,2-dichloroethane, however, the relative population of the *gauche* conformer is significantly increased relative to the *trans* conformer by comparison with the isolated molecule, with 44% *trans* and 56% *gauche*. These experimental results were reproduced by Jorgensen (see Figure 7.13) using Monte Carlo simulations [Jorgensen *et al.* 1981]. The potential of mean force would be designed to reproduce this new population and so enable a single 1,2-dichloroethane molecule to be simulated as if it were present in the liquid.

A simulation performed using a potential of mean force enables the modulating effects of the solvent to be taken into account. The solvent also influences the dynamic behaviour of the

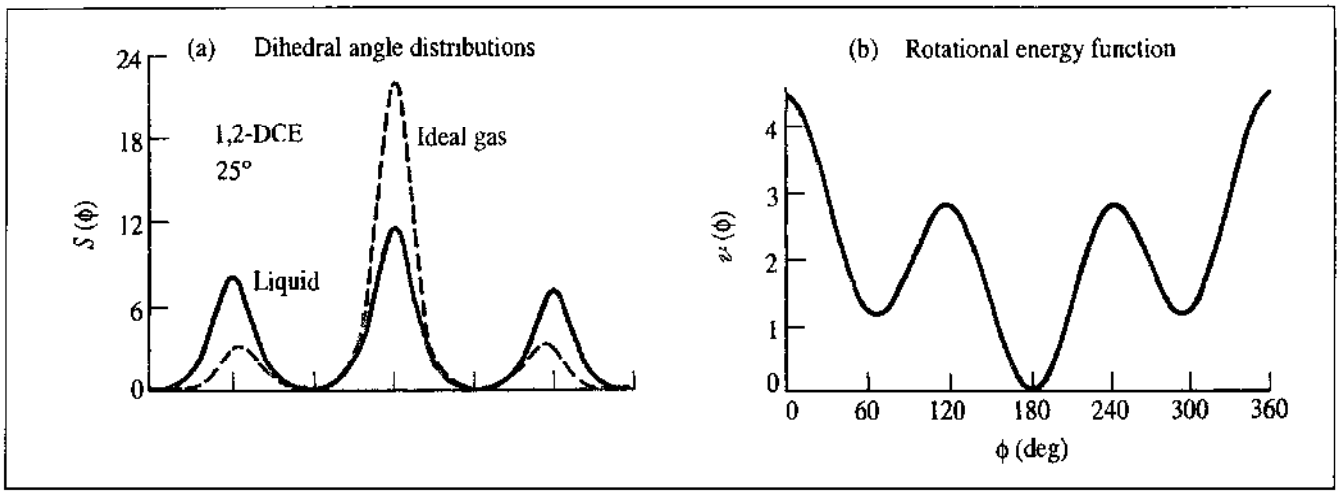


Fig 7 13· Population distribution for 1,2-dichloroethane in the gas and liquid phases (Figure redrawn from Jorgensen W L, R C Binning Jr and B Bigot 1981. Structures and Properties of Organic Liquids. n-Butane and 1,2-Dichloroethane and Their Conformational Equilibria. Journal of the American Chemical Society ***103*** *4393–4399)*

solute via random collisions, and by imposing a frictional drag on the motion of the solute through the solvent. The Langevin equation of motion is the starting point for the *stochastic dynamics* models, which also incorporate these two effects. In stochastic dynamics the force on a particle is considered to arise from three sources. The first component is due to interactions between the particle and other particles This force ($\mathbf{F}_i$) depends upon the position of the particle relative to the other particles and is modelled using a potential of mean force. The second force arises from the motion of the particle through the solvent and is equivalent to the frictional drag on the particle due to the solvent. This frictional force is proportional to the speed of the particle with the constant of proportionality being the friction coefficient:

$$\mathbf{F}_{\text{frictional}} = -\xi\mathbf{v} \tag{7.112}$$

where $\mathbf{v}$ is the velocity and ξ is the friction coefficient. The friction coefficient is related to the collision frequency (γ) by $\gamma = \xi/m$ (m is the mass of the particle). γ^{-1} can be considered as the time taken for the particle to lose memory of its initial velocity (the velocity relaxation time). For a spherical particle the friction coefficient is related to the diffusion constant D by:

$$\xi = k_B T/D \tag{7.113}$$

If the radius of the spherical particle is a then the frictional force is given by Stokes' law:

$$\mathbf{F}_{\text{frictional}} = 6\pi a\eta\mathbf{v} \tag{7.114}$$

where η is the viscosity of the fluid.

The third contribution to the force on the particle is due to random fluctuations caused by interactions with solvent molecules. We will write this force as $\mathbf{R}(t)$. The Langevin equation of motion for a particle i can therefore be written*:

$$m_i \frac{d^2x_i(t)}{dt^2} = \mathbf{F}_i\{x_i(t)\} - \gamma_i \frac{dx_i(t)}{dt} m_i + \mathbf{R}_i(t) \tag{7.115}$$

* γ_i in Equation (7 115) is often referred to as the friction coefficient in the literature.

A number of simulation methods based on Equation (7.115) have been described. These differ in the assumptions that are made about the nature of frictional and random forces. A common simplifying assumption is that the collision frequency γ is independent of time and position. The random force $\mathbf{R}(t)$ is often assumed to be uncorrelated with the particle velocities, positions and the forces acting on them, and to obey a Gaussian distribution with zero mean. The force $\mathbf{F}_i$ is assumed to be constant over the time step of the integration.

Three different situations can be considered, depending upon the relative magnitudes of the integration time step and the velocity relaxation time. The first category corresponds to timescales that are short relative to the velocity relaxation time ($\gamma\delta t \ll 1$). Under such circumstances the solvent does not activate or deactivate the particle to any significant extent. In the limit of zero γ (when there are no effects due to solvent) then the Langevin Equation (7.115) reduces to that obtained from Newton's laws of motion. At the other extreme the velocity relaxation time is much smaller than the time step. This corresponds to the diffusive regime, where the motion is rapidly damped by the solvent. The third situation is intermediate between these two extremes Various methods have been proposed for integrating the Langevin equation of motion in these three regions.

In the region where $\gamma\delta t \ll 1$ the following is a simple integration algorithm [van Gunsteren *et al.* 1981]:

$$x_{i+1} = x_i + v_i\delta t + \tfrac{1}{2}(\delta t)^2\{-\gamma v_i + m^{-1}(F_i + R_i)\} \tag{7.116}$$

$$v_{i+1} = v_i + (\delta t)\{-\gamma v_i + m^{-1}(F_i + R_i)\} \tag{7.117}$$

The average random force over the time step is taken from a Gaussian with a variance $2mk_BT\gamma(\delta t)^{-1}$. x_i is one of the $3N$ coordinates at time step i; F_i and R_i are the relevant components of the frictional and random forces at that time; v_i is the velocity component.

An alternative expression is based on the following finite difference approximations [Brunger *et al.* 1984]:

$$d^2x/dt^2 \approx (x_{i+1} - 2x_i + x_{i-1})/\delta t^2 \tag{7.118}$$

$$dx/dt \approx (x_{i+1} - x_{i-1})/2\delta t \tag{7.119}$$

This leads to the following expressions for the coordinates x_{i+1}:

$$x_{i+1} = x_i + (x_i - x_{i-1})\frac{1 - \frac{1}{2}\gamma\delta t}{1 + \frac{1}{2}\gamma\delta t} + \left(\frac{\delta t^2}{m}\right)\frac{F_i + R_i}{1 + \frac{1}{2}\gamma\delta t} \tag{7.120}$$

In the region where $\gamma\delta t \gg 1$ then if the interparticle force is assumed to be constant over the integration time step the following result is obtained [van Gunsteren *et al.* 1981]:

$$x_{i+1} = x_i + F_i(m\gamma)^{-1}\delta t + X_i(\delta t) \tag{7.121}$$

where X_i is a Gaussian distribution with zero mean and a variance of $2k_BT(m\gamma)^{-1} = 2D\delta t$ An extension of this treatment is to permit force F_i to vary linearly over the time step, giving:

$$x_{i+1} = x_i + \frac{\delta t}{m\gamma}(F_i + \tfrac{1}{2}\dot{F}_i\delta t) + X_i \tag{7.122}$$

$\dot{F}_i$ is the derivative of the force at the time step i and is obtained numerically:

$$\dot{F}_i = (F_i - F_{i-1})/\delta t \tag{7.123}$$

In the intermediate region, where there are no restrictions on $\gamma\delta t$, then integration of the equations of motion gives the following rather complicated result [van Gunsteren and Berendsen 1982]:

$$\begin{aligned} x_{i+1} &= x_i + v_i\gamma^{-1}(1 - \exp(-\gamma\delta t)) + F_i(m\gamma)^{-1}[\delta t - \gamma^{-1}(1 - \exp(-\gamma\delta t))] \\ &\quad + (mg)^{-1}\int_{t_i}^{t_{i+1}} [1 - \exp(-\gamma(t_{i+1} - t'))]R(t')\,dt' \end{aligned} \tag{7.124}$$

$$\begin{aligned} v_{i+1} &= v_i \exp(-\gamma\delta t) + F_i(m\gamma)^{-1}(1 - \exp(-\gamma\delta t)) \\ &\quad + (m)^{-1}\int_{t_i}^{t_{i+1}} \exp(-\gamma(t_{i+1} - t'))R(t')\,dt' \end{aligned} \tag{7.125}$$

The important feature of these two equations is that the new positions and the new velocities both depend upon an integral over the random force, $R(t)$ (the final terms in Equations (7.124) and (7.125). As both of these integrals depend upon $R_i(t)$ they are correlated. Specifically, they obey a *bivariate* Gaussian distribution. Such a distribution provides the probability that a particle located at x_i at time t with velocity v_i and experiencing a force F_i will be at x_{i+1} at time $t + \delta t$ with velocity v_{i+1}. In practice, this means that the distribution for the second variable depends upon the value selected for the first variable. It can be difficult to properly sample from such distributions, but van Gunsteren and Berendsen showed that the equations can be reformulated in terms of sampling from two independent Gaussian functions.

More complex stochastic dynamics treatments are possible; our treatment has only provided a rather simple treatment of solvent effects. For example, we have assumed that the frictional force at a given instant is proportional only to its velocity at the same time. A more realistic model assumes that the frictional forces are correlated; they have a 'memory' of previous values. The friction coefficient can also be made to depend on the coordinates of the other particles.

7.8.1 Practical Aspects of Stochastic Dynamics Simulations

A stochastic dynamics simulation requires a value to be assigned to the collision frequency friction coefficient γ. For simple particles such as spheres this can be related to the diffusion constant in the fluid. For the simulation of a rigid molecule it may be possible to derive γ via the diffusion coefficient from a standard molecular dynamics situation. In the more general case we require the friction coefficient of each atom. For simple molecules such as butane the friction coefficient can be considered to be the same for all atoms. The optimal value for γ can be determined by trial and error, performing a stochastic dynamics simulation for different values of γ and comparing the results with those from experiment (where available) or from standard molecular dynamics simulations. For large molecules the atomic friction coefficient is considered to depend upon the degree to which each atom is in contact with the solvent and is usually taken to be proportional to the accessible surface area of the atom (as defined in Section 1.5).

One of the main advantages of the stochastic dynamics methods is that dramatic time savings can be achieved, which enables much longer stimulations to be performed For example, Widmalm and Pastor performed 1 ns molecular dynamics and stochastic dynamics simulations of an ethylene glycol molecule in aqueous solution of the solute and 259 water molecules [Widmalm and Pastor 1992]. The molecular dynamics simulation required 300 hours whereas the stochastic dynamics simulation of the solute alone required just 24 minutes. The dramatic reduction in time for the stochastic dynamics calculation is due not only to the very much smaller number of molecules present but also to the fact that longer time steps can often be used in stochastic dynamics simulations.

Stochastic dynamics has been widely used to study the behaviour of long-chain molecules and polymers. The advantages of stochastic dynamics are especially important for polymers [Helfand 1984], where many interesting phenomena occur over relatively long time periods, so putting them beyond the scope of conventional molecular dynamics. However, one must take care with the Langevin method when simulating systems in which specific solute–solvent interactions are present. For example, Yun-Yu, Lu and van Gunsteren used both stochastic dynamics and molecular dynamics to study the immunosuppressant drug cyclosporin (Figure 7.14) in two solvents: carbon tetrachloride and water [Yun-Yu *et al.* 1988]. The time-averaged structures obtained from each method were compared to determine the similarity between the average structure obtained for each simulation. Fluctuations in torsion angles were also compared. The analysis showed that the structures obtained from the molecular dynamics and stochastic dynamics simulations of cyclosporin in carbon tetrachloride were very similar, but that the results were very different for the Langevin and molecular dynamics simulations performed in water. This was due to an excessive degree of internal hydrogen bonding in the stochastic dynamics simulation; the equivalent

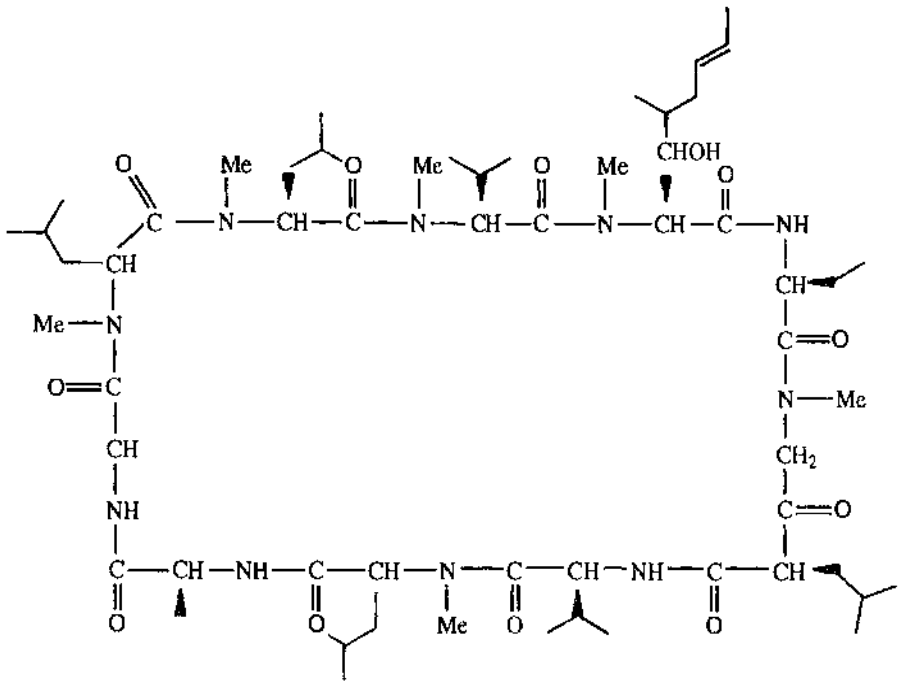


Fig 7 14· Cyclosporin

molecular dynamics simulation contained much more hydrogen bonding between cyclosporin and the solvent.

7.9 Conformational Changes from Molecular Dynamics Simulations

Molecular dynamics can provide information about the conformational properties of molecular systems and the way in which the conformation changes with time. Molecular graphics programs can facilitate the analysis of such simulations by displaying the structural parameters of interest in a manner that enables the time dimension to be taken into account. Perhaps the most direct way to demonstrate the conformational behaviour of the system is as a movie, where coordinate sets saved at regular intervals are displayed in sequence. For publication purposes, time-dependent data can be displayed graphically, with one of the axes corresponding to the time, such as the plots of energy or autocorrelation function versus time (Figures 7.3 and 7.10). The representation of bond rotations is difficult using x/y plots due to the 2π periodicity of a torsion angle. Lavery and Sklenar have developed a method to represent torsion data as a polar plot [Lavery and Sklenar 1988], where the distance from the origin corresponds to the time (Figure 7.15). Such 'dials' are very useful for detecting the presence of correlated conformational changes.

When viewing a movie of a molecular dynamics simulation of a complex molecule one is often struck by the chaotic nature of the motion. This should be expected; the motion of complex molecules *is* chaotic, but there are often underlying low-frequency motions which correspond to more significant and more interesting conformational changes. Fourier analysis techniques can be used to filter out the unwanted high-frequency motions, enabling the important low-frequency changes to be observed unhindered. Here we describe the filtering method of Dauber-Osguthorpe and Osguthorpe [Dauber-Osguthorpe and Osguthorpe 1990, 1993].

A Fourier transform enables one to convert the variation of some quantity as a function of time into a function of frequency, and vice versa. Thus, if we represent the quantity that varies in time as $x(t)$, then Fourier analysis enables us to also represent that quantity as a function $X(\nu)$, where ν is the frequency ($-\infty < \nu < \infty$). Fourier analysis is usually introduced by considering functions that vary in a periodic manner with time which can be written as a superposition of sine and cosine functions (a Fourier series; see Section 1 10.8). If the period of the function $x(t)$ is τ then the cosine and sine terms in the Fourier series are functions of frequencies $2\pi n/\tau$, where n can take integer values 1, 2, 3, ...

A Fourier series is rarely relevant to the interpretation of a molecular dynamics simulation as the movement of the atoms is not periodic but chaotic. The Fourier transform enables a non-periodic function to be converted into the equivalent frequency function (and vice versa). The Fourier transform can be developed from the Fourier series simply by considering the effect of increasing the period of a periodic function to infinity. The frequency function obtained from a Fourier transform is a continuous function rather than one written as a series of discrete frequencies. Further details are provided in Section 1.10.8.

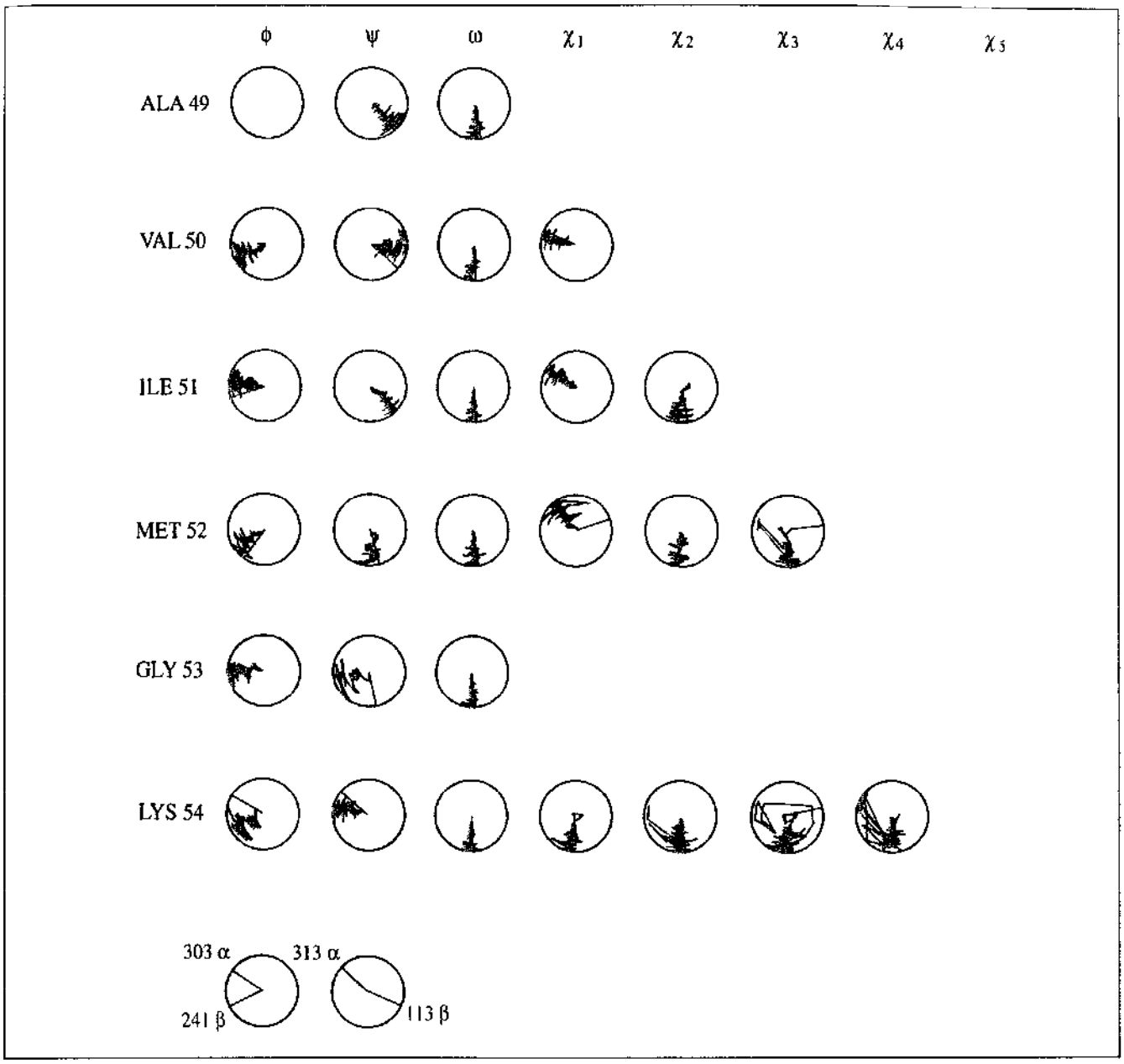


Fig 7 15 The variation in torsion angles can be effectively represented as a series of 'dials', where the time corresponds to the distance from the centre of the dial Data from a molecular dynamics simulation of an intermolecular complex between the enzyme dihydrofolate reductase and a triazine inhibitor [Leach and Klein 1995]

At each step of the Fourier analysis of a molecular dynamics simulation the variation with time of one of the Cartesian coordinates of one of the atoms in the system is converted into the correponding frequency function. Fast Fourier techniques are usually employed for this step. The frequency spectrum can then be filtered to remove high frequencies. This is achieved simply by setting the coefficients of the unwanted frequencies in the frequency function to zero. The resulting spectrum is then converted back to the time domain to give a new set of coordinate values at each of the time steps in the trajectory. This new coordinate set includes only the selected frequencies. This process can be repeated for the three coordinates of each atom to give a filtered trajectory for the entire system. It is also possible to select just a single frequency (i.e. a single normal mode) from the frequency spectrum and view this in isolation.

7.10 Molecular Dynamics Simulations of Chain Amphiphiles

The molecular dynamics technique is widely used for simulating large molecular systems, some of which have many degrees of conformational freedom. In this section we will examine the application of molecular dynamics to chain amphiphiles, a class of molecules of interest to both the 'biological' and 'materials science' communities These molecules have a polar head group attached to one or more hydrocarbon chains. Some examples are shown in Figure 7.16. The head group has a high affinity for water, whereas the hydrocarbon tail prefers to exist in a hydrophobic environment. The molecules therefore exist in both phases at a water/oil interface. A characteristic feature of these molecules is their ability

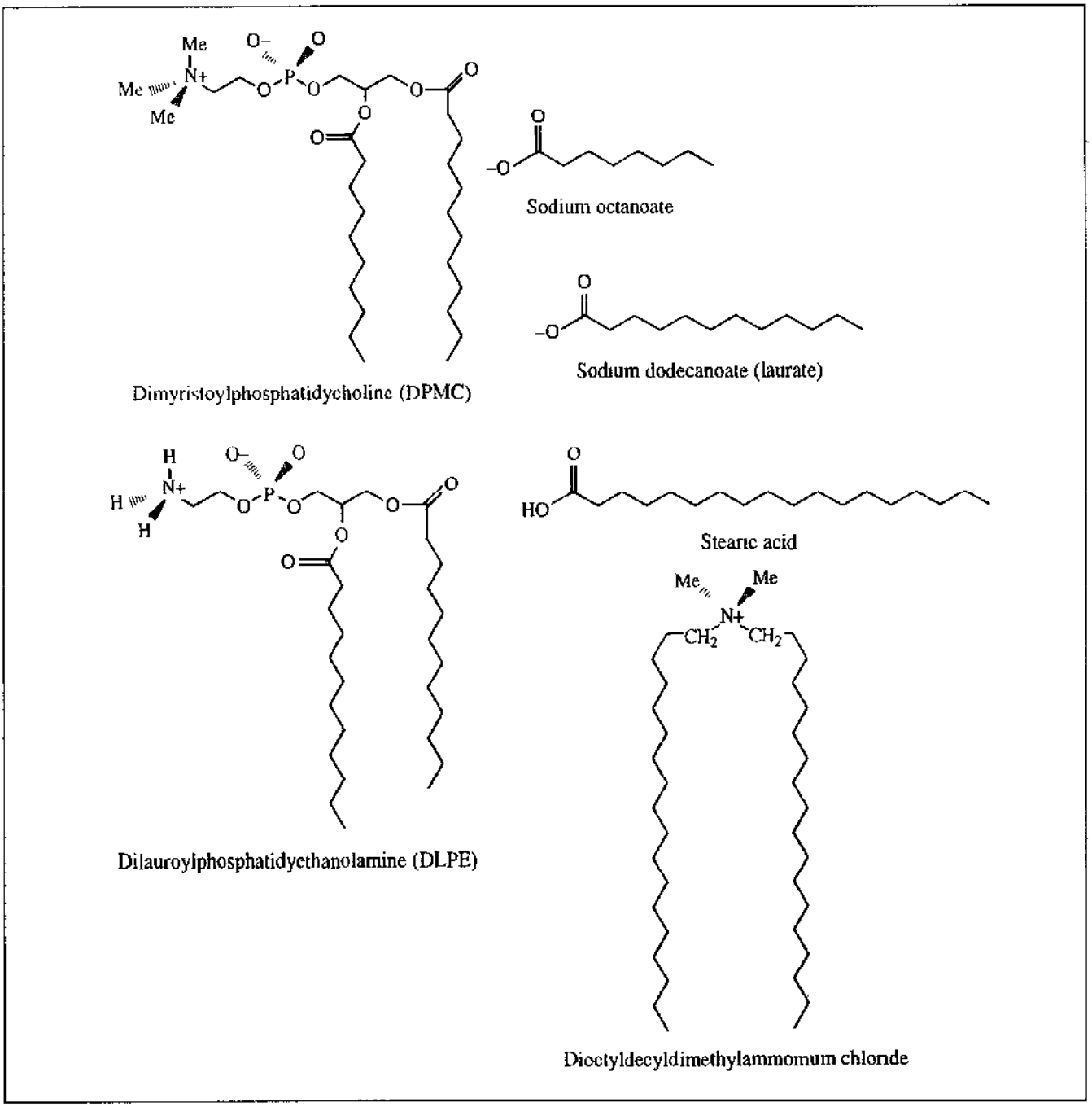


Fig. 7.16· Some typical amphiphiles

to form extended layer structures. Monolayers, bilayers and multiple layers are all possible. A monolayer at the water/air interface is known as a Langmuir film; when this is transferred to a solid substrate it is known as a Langmuir–Blodgett film. Langmuir–Blodgett films with many layers can be constructed in the laboratory but most simulation studies of these systems have been restricted to monolayers or bilayers. The ability to control the thickness of a Langmuir–Blodgett film and their high degree of order means that they are intensively investigated as insulators in semiconductors, filtration devices and as anti-reflective coatings. Amphiphiles are important in biology as cell membranes are formed from lipid bilayers. At a high enough concentration some amphiphiles can form micelles, which are globular structures that have the head groups all pointing into solution and the tails inside (Figure 7.17).

Amphiphiles often have a complex phase behaviour with several liquid crystalline phases. These liquid crystalline phases are often characterised by long-range order in one direction together with the formation of a layer structure. The molecules may nevertheless be able to move laterally within the layer and perpendicular to the surface of the layer. Structural information can be obtained using spectroscopic techniques including X-ray and neutron diffraction and NMR. The quadrupolar splitting in the deuterium NMR spectrum can be

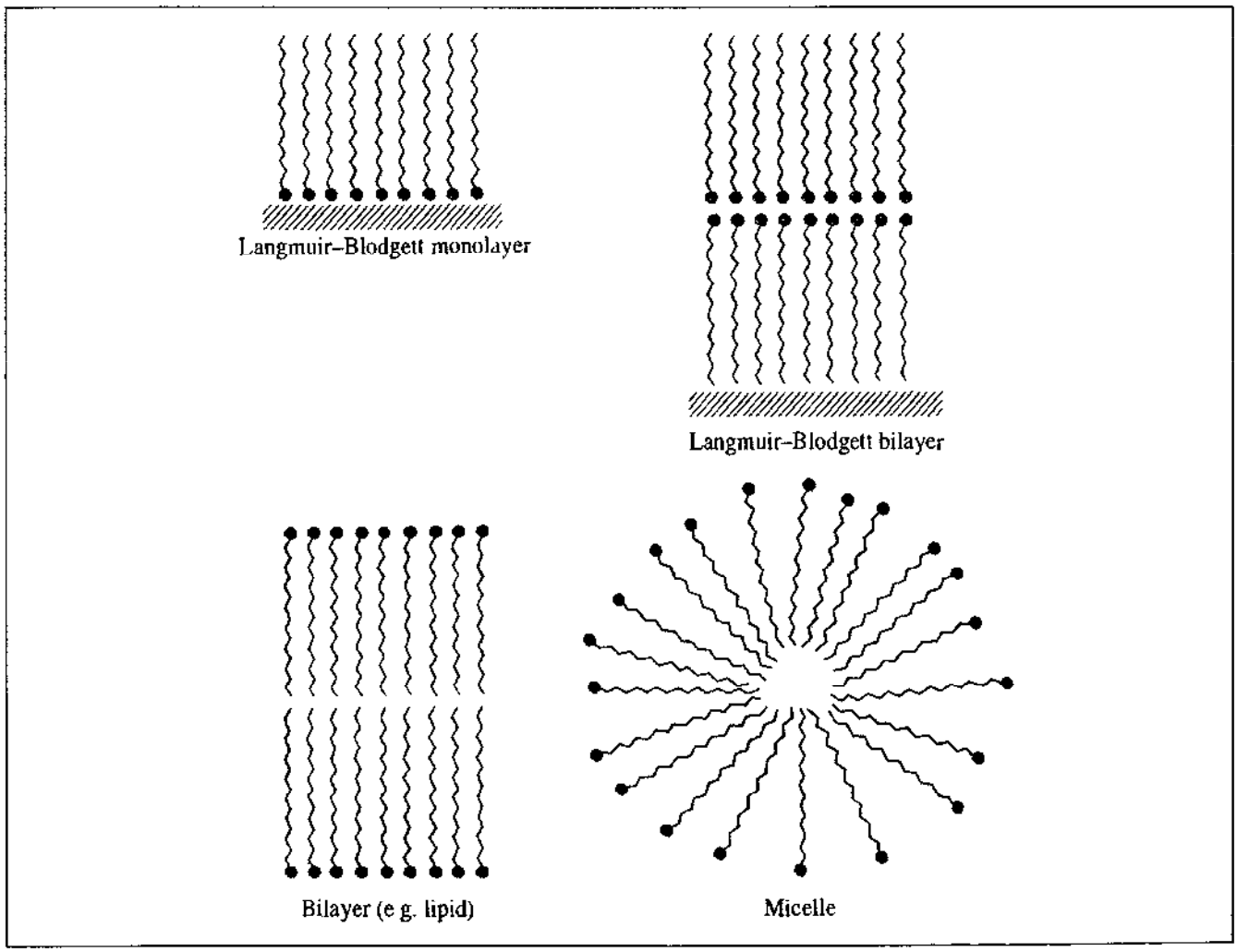


Fig 7 17 Some of the various phases that amphiphiles may form.

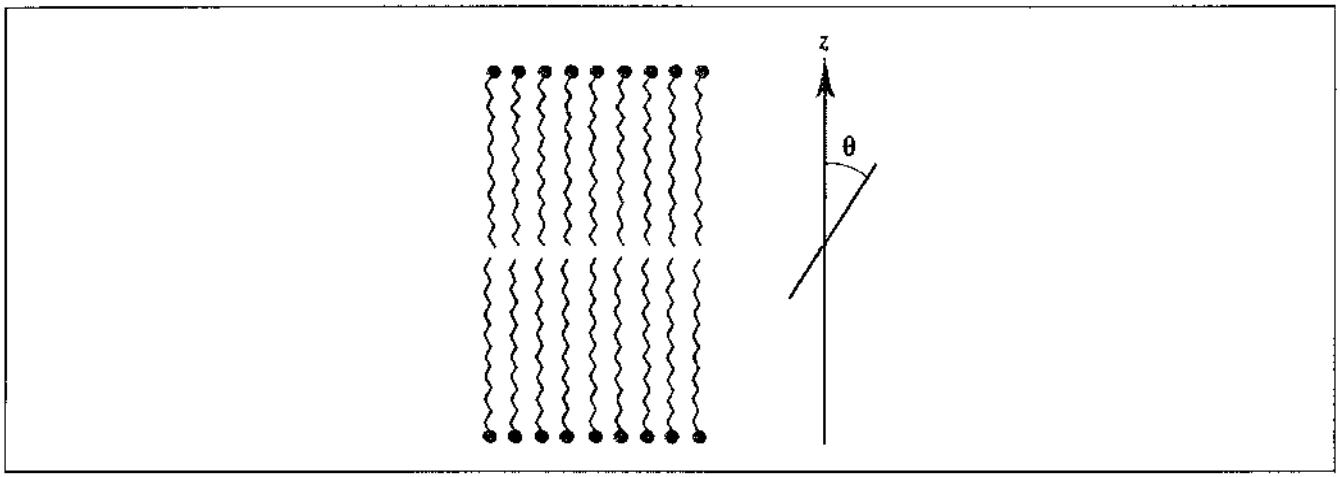


Fig 7.18 Definition of the order parameter

used to determine an *order parameter* for the carbon atoms on the hydrocarbon tail. The order parameter is defined as.

$$S = 0.5\langle 3\cos\theta_i \cos\theta_j - \delta_{ij}\rangle \tag{7.126}$$

θ_i is the angle between the ith molecular axis and the *director*, which is the average of the molecular axes over the sample. For a bilayer in the $L\alpha$ phase (the one present in cell membranes) the director is the same as the bilayer normal and is conventionally taken to be the z axis; see Figure 7.18. δ_{ij} is the Kronecker delta function ($\delta_{ij} = 1$ if $i = j$; $\delta_{ij} = 0$ if $i \neq j$) The expression for S is averaged over time and over molecules. The deuterium NMR experiment provides the order parameter S_{CD}, which indicates the average orientation of the C–D bond vector with respect to the bilayer normal. The experimental order parameters S_{CD} can range from 1 0 (indicating full order along the bilayer normal) to −0.5 (full order perpendicular to the bilayer normal) [Seelig and Seelig 1974]. A value of zero is considered to indicate full isotropic motion of the group. Experimental values are determined using molecules with deuterium-substituted methylene groups at positions along the hydrocarbon chain. Many simulations of amphiphiles are performed using united atom models for the hydrocarbon chains and it is therefore necessary to be able to relate the experimental order parameters to values that can be calculated from a simulation. This is done as follows [Essex *et al.* 1994]. Molecular axes are defined for each CH_2 unit in the chain as shown in Figure 7 19. These molecular axes are defined for the nth CH_2 unit as follows:

z: vector from C_{n-1} to C_{n+1}
y: vector perpendicular to z and in the plane through C_{n-1}, C_n and C_{n+1}
x: perpendicular to y and z

Using these definitions, components of the molecular order parameter tensor can be determined (for example, S_{zz} is determined by measuring the angle between the molecular z axis and the bilayer normal). The experimental order parameter can be related to the molecular order parameter using the equation:

$$S_{CD} = 2S_{xx}/3 + S_{yy}/3 \tag{7.127}$$

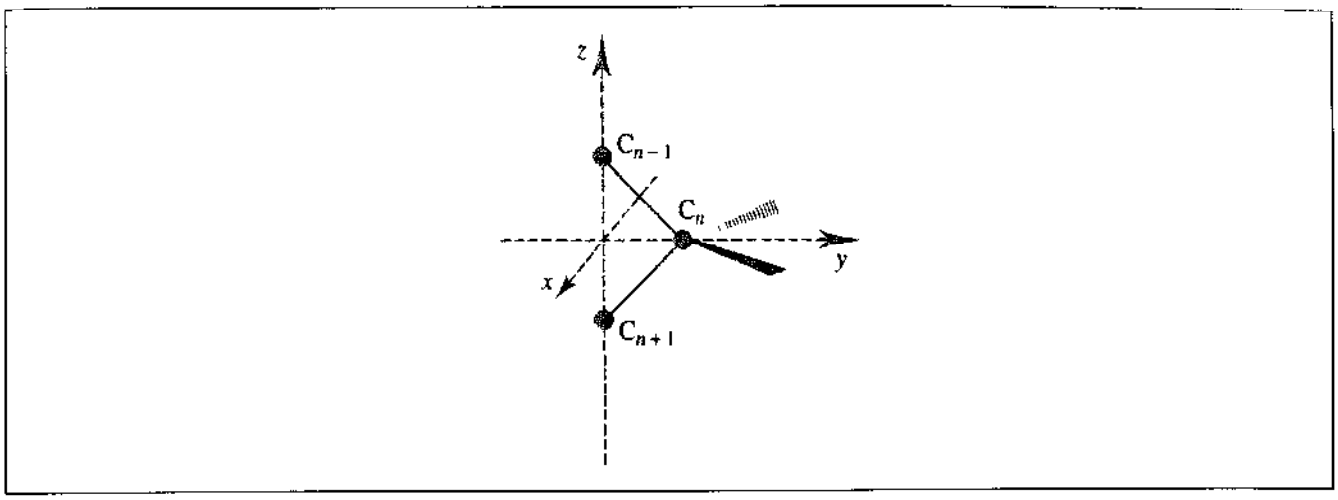


Fig 7 19. Calculation of the order parameter for united atom simulations.

With all-atom simulations the locations of the hydrogen atoms are known and so the order parameters can be calculated directly. Another structural property of interest is the ratio of *trans* conformations to *gauche* conformations for the CH_2-CH_2 bonds in the hydrocarbon tail. The *trans* : *gauche* ratio can be estimated using a variety of experimental techniques such as Raman, infrared and NMR spectroscopy.

7.10.1 Simulation of Lipids

There has been considerable interest in the simulation of lipid bilayers due to their biological importance. Early calculations on amphiphilic assemblies were limited by the computing power available, and so relatively simple models were employed. One of the most important of these is the mean field approach of Marcelja [Marcelja 1973, 1974], in which the interaction of a single hydrocarbon chain with its neighbours is represented by two additional contributions to the energy function. The energy of a chain in the mean field is given by:

$$\mathscr{V}_{\text{tot}} = \mathscr{V}_{\text{int}} + \mathscr{V}_{\text{disp}} + \mathscr{V}_{\text{rep}} \tag{7.128}$$

where $\mathscr{V}_{\text{int}}$ is the internal energy of a chain, which can be calculated using standard force field methods. $\mathscr{V}_{\text{disp}}$ simulates the van der Waals interactions with the neighbouring molecules. It is often modelled using a Maier-Saupe potential:

$$\mathscr{V}_{\text{disp}} = -\Phi \sum_{i=1}^{\text{carbons}} \tfrac{1}{2}(3\cos^2\theta_i - 1) \tag{7.129}$$

The summation runs over all carbon atoms in the chain. θ_i is the angle between the bilayer normal and the molecular axis, as discussed above. Φ is the field strength; this may be parametrised to reproduce appropriate experimental data such as the deuterium NMR order parameters or it may be obtained by a self-consistent protocol, as described below. In his work on lipid bilayers Marcelja used a slightly different expression for $\mathscr{V}_{\text{disp}}$ which

involved the fraction of *trans* bonds in the system:

$$\mathscr{V}_{\text{disp}} = -\Phi \frac{n_{\text{trans}}}{n} \sum_{i=1}^{\text{carbons}} \tfrac{1}{2}(3\cos^2\theta_i - 1) \tag{7.130}$$

This additional factor was introduced to ensure the proper behaviour over both liquid crystalline and solid phases. In simulations of the liquid crystalline phase alone this term may be omitted for computational efficiency.

The repulsive contribution, $\mathscr{V}_{\text{rep}}$, is due to lateral pressure on each chain. In Marcelja's original treatment, this was set equal to the product of the lateral pressure, γ, and the cross-sectional area of the chain. The cross-sectional area was approximated by:

$$A = A_0 l_0 / l \tag{7.131}$$

where l_0 and A_0 are the length and cross-sectional area, respectively, of the hydrocarbon chain in a fully extended conformation. l is the length of the chain in the current conformation, projected onto the bilayer normal. If the bilayer normal is along the z axis then l is taken to be the z coordinate of the last carbon atom in the hydrocarbon chain. In other mean field models [Pastor *et al.* 1988] the product $\gamma A_0/l_0$ is replaced with a single adjustable parameter Γ and so $\mathscr{V}_{\text{rep}}$ is given by:

$$\mathscr{V}_{\text{rep}} = \sum_{\text{chains}} \frac{\Gamma}{(z_n - z_0)} \tag{7.132}$$

where z_n is the z coordinate of the last carbon in the chain and z_0 is the coordinate of the surface of the monolayer or bilayer. This force acts to keep the last carbon away from the surface; the closer it gets the larger the force pulling it away.

In his calculations, Marcelja generated all possible conformations of the hydrocarbon chain, restricting each carbon–carbon bond to the *trans* and *gauche* conformations. The energy of each conformation was evaluated. From the ensemble of conformations a partition function can be computed:

$$Z = \sum_{\text{all conformations}} \exp[-\mathscr{V}_{\text{tot}}/k_{\text{B}}T] \tag{7.133}$$

The molecular field is related to the partition function:

$$\Phi = \sum_{\text{all conformations}} \left\{ \frac{\frac{n_{\text{trans}}}{n} \sum_{i=1}^{\text{carbons}} \tfrac{1}{2}(3\cos^2\theta_i - 1)\exp[-\mathscr{V}_{\text{tot}}/k_{\text{B}}T]}{Z} \right\} \tag{7.134}$$

The molecular field is thus related to the partition function and so it is possible to generate a self-consistent value of the molecular field, Φ. Thermodynamic properties can then be calculated from the partition function. For example, Marcelja calculated the pressure as a function of the area per polar head group for surface monolayers at a variety of temperatures. His results showed good qualitative agreement with experimental results for such systems.

The mean field approach can be incorporated into a molecular dynamics simulation. It is particularly useful when used in conjunction with Langevin dynamics, as very long simulations can be performed. For example, Pearce and Harvey were able to perform simulations of three unsaturated phospholipids for 100 ns (i.e. 0.1 μs) in single-molecule Langevin dynamics calculations [Pearce and Harvey 1993]. An extension of this strategy is to use a central 'core' containing one or more molecules that are simulated using molecular dynamics This core is surrounded by a shell of molecules that are simulated using Langevin dynamics with the mean field. In this way one attempts to simulate a more 'realistic' system without incurring the computational penalty of a full molecular dynamics simulation of the entire system [De Loof *et al.* 1991].

The first molecular dynamics simulations of a lipid bilayer which used an explicit representation of all the molecules was performed by van der Ploeg and Berendsen in 1982 [van der Ploeg and Berendsen 1982]. Their simulation contained 32 decanoate molecules arranged in two layers of sixteen molecules each. Periodic boundary conditions were employed and a united atom force potential was used to model the interactions. The head groups were restrained using a harmonic potential of the form:

$$v(z) = \frac{k_h}{2}(z - \langle z \rangle)^2 \tag{7 135}$$

By writing the restraint in terms of the average z coordinates of the head groups ($\langle z \rangle$) van der Ploeg and Berendsen ensured that the bilayer was able to change its thickness to reach its equilibrium value. This restraining potential was designed to reproduce the interactions between the head groups and the water layer, neither of which was explicitly included in the calculation. A key feature of the simulation was the long equilibration time required. By explicitly representing all the molecules in the system it was possible to determine the collective motion of the system as a whole. One distinct feature was a slowly fluctuating collective tilt of the molecules away from the normal to the bilayer surface (Figure 7.20). The degree to which the molecules were aligned with each other was also correlated with the tilt angle. When the average tilt angle reached a maximum the chains were much more likely to be well aligned, but when the average tilt angle was close to zero (i.e. such that the average orientation of the chains was almost normal to the bilayer surface) much less order was observed. In their original simulations this collective tilt phenomenon was observed to extend over the entire simulation cell, suggesting that the cell dimensions were too small and that the use of periodic boundary conditions was enhancing the long-range correlations. Simulations using a larger system subsequently showed that this collective tilt could be observed for subsets of the molecules.

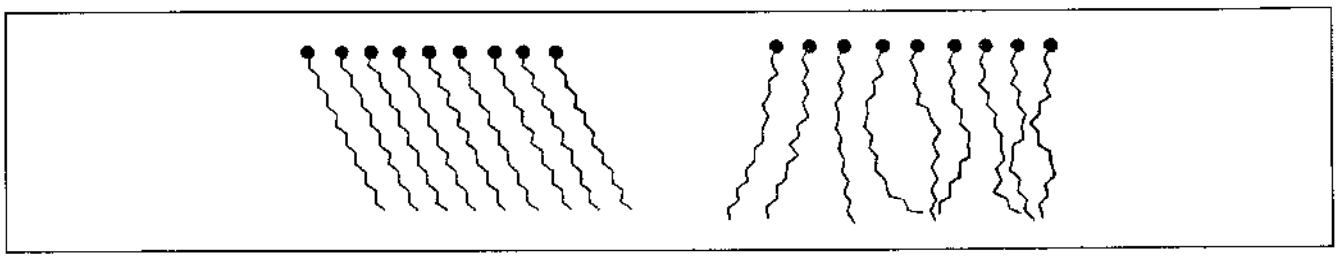


Fig. 7 20 Variation in alignment of chains in lipid simulation with tilt angle [van der Ploeg and Berendsen 1982]

Faster computers have enabled more realistic simulations of lipid bilayers to be performed, of larger systems, with more accurate models and for longer times [Stouch 1993; Tobias *et al.* 1997]. The trend is very much towards simulations that use full representations of all species present (i e 'all-atom' models with explicit solvent and counterions). The charged and highly polar nature of lipid head groups means that a proper representation of the long-range electrostatic forces can be critical, using a method such as the Ewald summation. Equilibration of such systems often requires hundreds of picoseconds and certain phenomena are only observed on a nanosecond timescale. In addition, molecules such as cholesterol and proteins may be included within the membrane These simulations have revealed many hitherto unknown features of the behaviour of such systems. For example, considerable conformational mobility of the hydrocarbon chains is often observed in the liquid crystalline phases. This is illustrated in Figure 7.21 (colour plate section), which shows a snapshot of a lipid bilayer after a molecular dynamics simulation of several hundred picoseconds. The considerable degree of disorder in the hydrocarbon chains near the middle of the bilayer is clear from this figure and is very different to the idealised, 'textbook' pictures in which the chains are perfectly aligned in completely extended conformations. The distribution of *gauche* conformations tends to be higher towards the end of the chain, though in some systems a *gauche* link is required near the head group to enable the chain to lie perpendicular to the interface. 'Kinks' are often observed in the chains; these are arrangements of three successive bonds with *gauche*(+)-*trans*-*gauche*(−) torsion angle, which enable the chain to remain perpendicular to the surface.

7.10.2 Simulations of Langmuir–Blodgett films

The simulations of Langmuir-Blodgett systems can be difficult due to the need to correctly model the solid support To illustrate the procedure we will describe the calculations of Kim, Moller, Tildesley and Quirke [Kim *et al.* 1994a] who simulated stearic acid ($CH_3(CH_2)_{16}COOH$) adsorbed onto graphite. The surface was modelled using a Lennard-Jones 9–3 potential that depends upon the height of the atom (α) above the surface (z_α):

$$v_{\alpha s}(z_\alpha) = \frac{2\pi\rho}{3}\varepsilon_{ss}\left[\frac{2}{15}\left(\frac{\sigma_{\alpha s}}{z_\alpha}\right)^9 - \left(\frac{\sigma_{\alpha s}}{z_\alpha}\right)^3\right] \tag{7.136}$$

where ρ is the density of the solid and ε_{xx} and δ_{ss} are its Lennard-Jones parameters. An image-change method was also applied to the acid head group with the interaction between a charge and its image being:

$$v_{ic}(z) = \frac{1}{2}\frac{(\varepsilon - \varepsilon')}{(\varepsilon + \varepsilon')}\left[\frac{q_\alpha^2}{8\pi\varepsilon_0(z - z_{ip})}\right] \tag{7.137}$$

where ε' is the relative permittivity of the solid (taken to be 4.0) and ε is the permittivity above the surface ($\varepsilon = 1.0$). The image plane is located at $z_{ip} = \sigma_{ss}/2$. Each charge interacts with its own image and with the images of other charges, but there are no interactions between the image charges themselves. The hydrocarbon chain of the stearic acid was modelled using an all-atom model, with explicit hydrogen atoms.

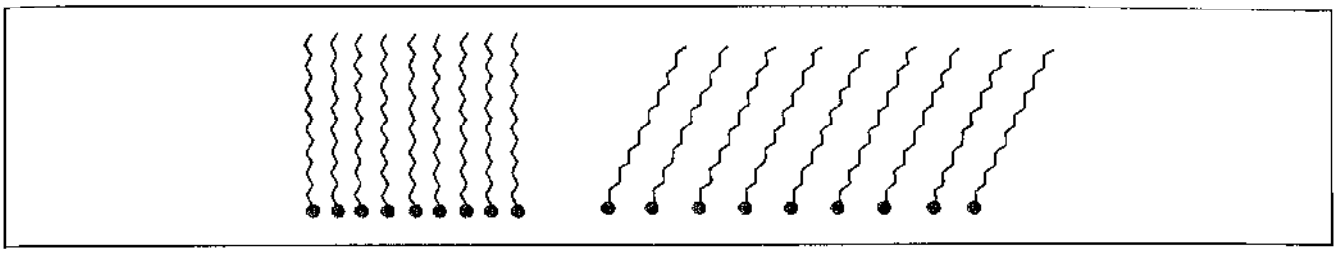


Fig 7.22 Simulations of a Langmuir–Blodgett film [Kim et al *1994a]: as the area per head group increases the chains tilt away from the normal*

A molecular dynamics simulation of 64 molecules with periodic boundary conditions confirmed the presence of a transition in which the collective tilt of the chains changed from being upright (i.e. perpendicular to the surface) to having an angle of around 20° (Figure 7.22). This transition was induced by increasing the area per head group. The proportion of molecules in the all-*trans* conformation decreased significantly as the head group area was increased (97.7% of molecules were fully extended for a head group area of 20.6 $Å^2$ but only 66.9% for an area of 21.2 $Å^2$). The bond linking the chain to the acid head growth showed a considerable degree of rotational disorder.

Bilayers of stearic acid were also simulated on a hydrophobic surface [Kim *et al.* 1994b]. In the bilayer the molecules are arranged head to head, with the hydrocarbon tail on the surface. In this arrangement hydrogen bonds form between the head groups (Figure 7.23). The bilayer also showed the tilt angle transition that was observed for the monolayer, though the degree of tilt was considerably less for the bilayer, suggesting that hydrogen bonding between the head groups was important in controlling the orientation of the molecules.

An extension of these calculations to cationic dialkylamide salts required an even more complex model [Adolf *et al.* 1995]. These molecules have the general formula $(CH_3)_2N^+[(CH_2)_{n-1}CH_3][(CH_2)_{m-1}CH_3]Cl^-$ and the isomer with $m = n = 18$ is one of the main active ingredients in commercial fabric softeners. The presence of two long alkyl

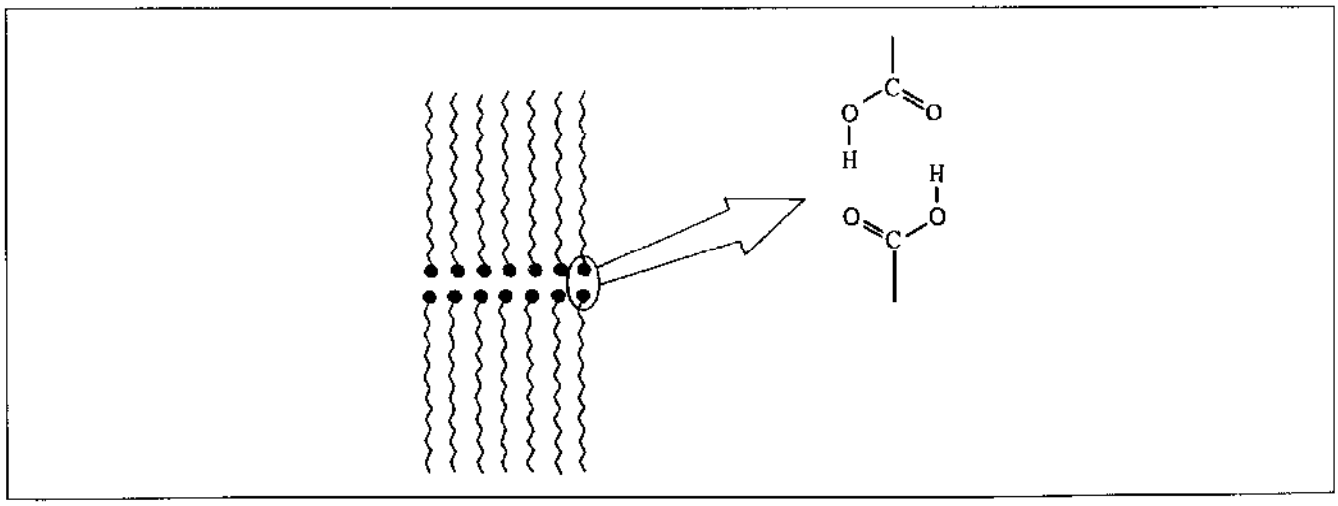


Fig 7.23 In simulations of stearic acid on a hydrophobic surface hydrogen bonding between the head groups is important in controlling the orientation of the molecules [Kim et al *1994b]*

chains and an ionic head group means that these molecules are also structurally similar to phospholipids. A modified Ewald method was used to calculate electrostatic interactions in the two dimensions parallel to the surface, and the anisotropic potential model of Toxvaerd (see Section 4.15) was employed to retain the computational savings of a united-atom model. This system also showed a variation in the tilt with head group area, though the results at the highest head group densities were not as 'solid-like' as was suggested by the experimental data. Nevertheless, there were some areas where the model could be improved, including the need to incorporate water molecules and use a more appropriate representation of the chloride anion.

7.10.3 Mesoscale Modelling: Dissipative Particle Dynamics

The molecular dynamics methods that we have discussed in this chapter, and the examples that have been used to illustrate them, fall into the category of 'atomistic simulations', in that all of the actual atoms (or at least the non-hydrogen atoms) in the core system are represented explicitly. Atomistic simulations can provide very detailed information about the behaviour of the system, but as we have discussed this typically limits a simulation to the nanosecond timescale. Many processes of interest occur over a longer timescale. In the case of processes which occur on a 'macroscopic' timescale (i.e. of the order of seconds) then rather simple models may often be applicable. Between these two extremes are phenomena that occur on an intermediate scale (of the order of microseconds). This is the realm of the *mesoscale* Dissipative particle dynamics (DPD) is particularly useful in this region, examples include complex fluids such as surfactants and polymer melts.

These three general regions (atomistic, mesoscopic and macroscopic) are not only characterised by different timescales but also varying length scales Indeed, there is a general inverse relationship between the time and the length In the case of the dissipative particle dynamics method the fast motion of the atoms is integrated out, leaving as the fundamental 'unit' a set of beads that interact with other beads via an appropriate potential [Koelman and Hoogerbrugge 1993]. Each bead represents a small 'droplet' of the fluid. The total force on each bead is due to a combination of direct interactions with other beads together with random and dissipative forces. The trajectory of the system is calculated by integrating Newton's laws of motion in the usual way, from which properties can be derived.

The underlying model in dissipative particle dynamics is usually developed in such a way that the mass, length and timescales are all unity. This is similar to the use of reduced units for the Lennard-Jones potential (Section 4.10.5). A particular advantage of such an approach is that a single simulation may often be able to explain the behaviour of many different systems. With a mass of 1 the force acting on a particle is equal to its acceleration. In DPD there are three forces on each bead [Groot and Warren 1997]:

$$\mathbf{f}_i = \sum_{j=1; j \neq i}^{N} (\mathbf{F}_{ij}^{\mathrm{C}} + \mathbf{F}_{ij}^{\mathrm{D}} + \mathbf{F}_{ij}^{\mathrm{R}}) \tag{7.138}$$

The summation is over all other particles j which are within a certain cutoff radius r_c of i. This cutoff radius becomes the unit of length in the subsequent treatment (i.e $r_c = 1$). The

first of these forces, the conservative force, $\mathbf{F}_{ij}^{C}$, is a soft repulsion that acts along the line joining i to j.

$$\mathbf{F}_{ij}^{C} = \begin{cases} a_{ij}(1 - r_{ij})\hat{\mathbf{r}}_{ij} & r_{ij} < 1 \\ 0 & r_{ij} > 1 \end{cases} \tag{7.139}$$

r_{ij} is the distance between beads i and j and $\hat{\mathbf{r}}_{ij}$ is the corresponding unit vector. The second force is a dissipative (or drag) force, which is given by:

$$\mathbf{F}_{ij}^{D} = \begin{cases} -\gamma w^{D}(r_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})\hat{\mathbf{r}}_{ij} & r_{ij} < 1 \\ 0 & r_{ij} > 1 \end{cases} \tag{7.140}$$

This dissipative force is proportional to the relative velocity of the two beads and acts so as to reduce their relative momentum. $\mathbf{v}_{ij}$ is the difference between the two velocities ($\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$) and $w^{D}(r_{ij})$ is a weight function that depends upon the distance r_{ij} and disappears for inter-bead distances greater than unity (i.e. r_c).

The third and final force acting between any pair of beads is a random force:

$$\mathbf{F}_{ij}^{R} = \begin{cases} \sigma w^{R}(r_{ij})\theta_{ij}\hat{\mathbf{r}}_{ij} & r_{ij} < 1 \\ 0 & r_{ij} > 1 \end{cases} \tag{7.141}$$

$w^{R}(r_{ij})$ is a distance-dependent weight function similar to that for the dissipative force. θ_{ij} is a function which ensures that the random force between each pair of particles averages to zero over time and is independent of the force between every other pair of particles. The random force can be more usefully expressed in terms of the timestep in the integration scheme:

$$\mathbf{F}_{ij}^{R} = \frac{\sigma w^{R}(r_{ij})\zeta_{ij}\hat{\mathbf{r}}_{ij}}{\sqrt{\delta t}} \tag{7.142}$$

ζ_{ij} is a random number with zero mean and unit variance, chosen independently for each pair of particles and at each time step in the integration.

Both the dissipative force and the random force act along the line joining the pair of beads and also conserve linear and angular momentum. The model thus has two unknown functions $w^{D}(r_{ij})$ and $w^{R}(r_{ij})$ and two unknown constants γ and σ. In fact, only one of the two weight functions can be chosen arbitrarily as they are related [Espanol and Warren 1995]. Moreover, the temperature of the system relates the two constants:

$$w^{D}(r) = [w^{R}(r)]^2 \tag{7.143}$$

$$\sigma^2 = 2\gamma k_B T \tag{7.144}$$

The usual choice for the weight functions is to make the random force the same as the conservative force:

$$w^{D}(r) = [w^{R}(r)]^2 = \begin{cases} (1 - r)^2 & r < 1 \\ 0 & r > 1 \end{cases} \tag{7.145}$$

The equations of motion are integrated using a modified velocity Verlet algorithm. The modification is required because the force depends upon the velocity, the extra step involves

a prediction followed by a correction. If, in addition to the use of units of mass and length, we assume that $k_B T$ is equal to 1, then the unit of time is:

$$\tau = r_c \sqrt{m/k_B T} \tag{7.146}$$

It remains to assign values to the noise amplitude, σ, the time step for the integration, δt, and the repulsion parameter, a_{ij}. The effects of the first two of these upon the stability of the simulation are also related to the integration method. Groot and Warren determined that when the noise amplitude was larger than 8 the integration scheme became unstable and that a value of 3 gave good results over a range of temperatures [Groot and Warren 1997]. The integration time step with the modified Verlet algorithm should have a value between 0.04 and 0.06; any larger and the temperature would artificially increase by an unacceptable amount. The repulsion parameter is the key determinant of the interactions between the beads. This can be achieved by relating the DPD model to bulk properties For example, to model the compressibility of water at room temperature the repulsion parameter is related to the density, ρ, by:

$$a_{ii}\rho = 75 k_B T \tag{7.147}$$

These interaction parameters between particles of the same type can be used to derive the values of a_{ij} between unlike beads. For polymers which involve beads of different types the repulsion between unlike beads is made larger than between like beads.

A good example of the use of DPD is the study by Groot and Madden of the microphase separation of diblock copolymer melts [Groot and Madden 1998; Groot *et al.* 1999]. Block copolymers are surfactants which are present in many consumer products such as foods (e.g. ice cream and margarine), detergents and personal care products (e g. shampoo). The properties of these materials are strongly dependent upon their bulk organisation (or *morphology*), which in turn depends upon the relative sizes of the head and the tail groups and how they interact. The diblock copolymers of interest can be represented by the general formula $A_m B_n$ where A and B represent amalgamations of the smaller building blocks from which the polymer is constructed. Of particular interest was the way in which the behaviour of the system varied as the ratio of A to B was changed, for a fixed polymer length. In this particular case the length was fixed at 10 beads and the entire simulation contained a total of 40 000 particles. A variety of systems were investigated, such as A_2B_8, A_3B_7 and A_5B_5. The beads in each polymer chain were kept together by adding an extra term to the force (Equation (7.139)) of the form Cr_{ij} if i is connected to j.

Due to the greater degree of repulsion between unlike beads the final configuration of the system contains domains which are rich in either the A or the B type of bead. Some regions are rich in A; others are rich in B. The organisation of the A-rich and B-rich domains can be visualised by plotting a three-dimensional contour that connects regions where the density is intermediate between purely A and purely B.

For the 1:1 polymer (A_5B_5) a lamellar phase was obtained, in which the A- and B-rich domains form parallel planes of alternating A and B beads (Figure 7.24(a), colour plate section). For other configurations, however, different structures were observed. The A_3B_7 system evolved to a hexagonal phase (Figure 7.24(b)) and the A_2B_8 structure produces a set of peanut-shaped micelles (Figure 7.24(c))

Appendix 7.1 Energy Conservation in Molecular Dynamics

The total energy is the sum of the kinetic $\mathscr{K}(t)$ and potential energies $\mathscr{V}(t)$:

$$E(t) = \mathscr{K}(t) + \mathscr{V}(t) \tag{7.148}$$

We want to derive an expression for the rate of change of the energy with time, dE/dt. First, we differentiate the kinetic energy term with respect to time:

$$\frac{d\mathscr{K}}{dt} = \sum_{i=1}^{N} \frac{d}{dt}\left(\tfrac{1}{2}m_i v_i^2\right) = \sum_{i=1}^{N} m_i v_i \frac{dv_i}{dt} \tag{7.149}$$

As $m_i\, dv_i/dt$ is equal to the force on the atom i, the result can be written:

$$\frac{d\mathscr{K}}{dt} = \sum_{i=1}^{N} v_i f_i \tag{7.150}$$

f_i is the force on atom i.

The potential energy is written as a series of pairwise interaction terms:

$$\mathscr{V}(t) = \sum_{i=1}^{N} \sum_{j=i+1}^{N} v(r_{ij}(t)) \tag{7.151}$$

The derivative of the potential energy with respect to time can be written:

$$\frac{d\mathscr{V}}{dt} = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{\partial \mathscr{V}}{\partial v(r_{ij})} \frac{dv(r_{ij})}{dt} \tag{7.152}$$

$\partial\mathscr{V}/\partial v(r_{ij})$ equals 1 for each pairwise combination i and j. Each term $v(r_{ij})$ is a function of the positions of atom i and j ($\mathbf{r}_i$ and $\mathbf{r}_j$) and we can then write:

$$\frac{dv(r_{ij})}{dt} = \frac{dv(r_{ij})}{d\mathbf{r}_i}\frac{d\mathbf{r}_i}{dt} + \frac{dv(r_{ij})}{d\mathbf{r}_j}\frac{d\mathbf{r}_j}{dt} \tag{7.153}$$

For a given atom i, there will be a total of $N-1$ terms of the form $v(r_{ij})$ in the expression for the potential energy due to the interactions between i and all other atoms j. Hence we can write $d\mathscr{V}/dt$ as follows:

$$\frac{d\mathscr{V}}{dt} = \sum_{i=1}^{N} \sum_{j=1,\, j\neq i}^{N} \frac{\partial v(r_{ij})}{\partial \mathbf{r}_i}\frac{d\mathbf{r}_i}{dt} = \sum_{i=1}^{N} \frac{d\mathbf{r}_i}{dt} \sum_{j=1;\, j\neq i}^{N} \frac{\partial v(r_{ij})}{\partial \mathbf{r}_i} \tag{7.154}$$

The force on atom i due to its interaction with atom j equals minus the gradient with respect to $\mathbf{r}_i$, or $-dv(r_{ij})/d\mathbf{r}_i$. Thus the total force on the atom is equal to

$$-\sum_{j=1;\, j\neq i}^{N} \frac{\partial v(r_{ij})}{\partial \mathbf{r}_i} \tag{7.155}$$

and so we have:

$$\frac{d\mathscr{V}}{dt} = -\sum_{i=1}^{N} \frac{d\mathbf{r}_i}{dt} f_i = -\sum_{i=1}^{N} v_i f_i \tag{7.156}$$

Thus $(d\mathscr{V}/dt) + (d\mathscr{K}/dt) = dE/dt = 0$, which implies that the energy is constant. In practice, the total energy fluctuates about a constant value.

Further Reading

Allen M P and D J Tildesley 1987 *Computer Simulation of Liquids*. Oxford, Oxford University Press.

Berendsen H C and W F van Gunsteren 1984. Molecular Dynamics Simulations Techniques and Approaches. In Barnes A J, W J Orville-Thomas and J Yarwood (Editors) *Molecular Liquids, Dynamics and Interactions*. NATO ASI Series C135, New York, Reidel, pp. 475–600

Berendsen H C and W F van Gunsteren 1986. Practical Algorithms for Dynamic Simulations. Molecular Dynamics Simulation of Statistical Mechanical Systems. *Proceedings of the Enrico Fermi Summer School Varenna Soc Italian di Fiscia* Bologna, pp 43–65

Brooks C L III, M Karplus and B M Pettitt 1988 Proteins. A Theoretical Perspective of Dynamics, Structure and Thermodynamics. *Advances in Chemical Physics* Volume LXXI. New York, John Wiley & Sons

Goldstein H 1980. *Classical Mechanics* (2nd Edition). Reading, MA, Addison-Wesley.

Haile J M 1992. *Molecular Dynamics Simulation Elementary Methods* New York, John Wiley & Sons

McCammon J A and S C Harvey 1987. *Dynamics of Proteins and Nucleic Acids*. Cambridge, Cambridge University Press

van Gunsteren W F 1994 Molecular Dynamics and Stochastic Dynamics Simulations: A Primer. In van Gunsteren W F, P K Weiner and A J Wilkinson (Editors) *Computer Simulations of Biomolecular Systems* Volume 2 Leiden, ESCOM

van Gunsteren W F and H J C Berendsen 1990 Computer Simulation of Molecular Dynamics: Methodology, Applications and Perspectives in Chemistry *Angewandte Chemie International Edition in English* **29**.992–1023

References

Adolf D B, D J Tildesley, M R S Pinches, J B Kingdon, T Madden and A Clark 1995. Molecular Dynamics Simulations of Dioctadecyldimethylammonium Chloride Monolayers. *Langmuir* **11**:237–246.

Alder B J and T E Wainwright 1957 Phase Transition for a Hard-sphere System. *Journal of Chemical Physics* **27**:1208–1209

Alder B J and T E Wainwright 1970. Decay of the Velocity Autocorrelation Function *Physical Review* **A1**:18–21.

Allen M P and D J Tildesley 1987 *Computer Simulation of Liquids*. Oxford, Oxford University Press

Anderson H C 1980. Molecular Dynamics Simulations at Constant Pressure and/or Temperature. *Journal of Chemical Physics* **72**:2384–2393.

Anderson H C 1983. Rattle A 'Velocity' Version of the Shake Algorithm for Molecular Dynamics Calculations. *Journal of Computational Physics* **54**:24–34.

Beeman D 1976 Some Multistep Methods for Use in Molecular Dynamics Calculations. *Journal of Computational Physics* **20** 130–139.

Berendsen H J C, J P M Postma, W F van Gunsteren, A Di Nola and J R Haak 1984 Molecular Dynamics with Coupling to an External Bath. *Journal of Chemical Physics* **81**:3684-3690

Brunger A, C B Brooks and M Karplus 1984 Stochastic Boundary Conditions for Molecular Dynamics Simulations of ST2 Water. *Chemical Physics Letters* **105**:495–500.

Dauber-Osguthorpe P and D J Osguthorpe 1990 Analysis of Intramolecular Motions by Filtering Molecular Dynamics Trajectories. *Journal of the American Chemical Society* **112**:7921-7935

Dauber-Osguthorpe P and D J Osguthorpe 1993 Partitioning the Motion in Molecular Dynamics Simulations into Characteristic Modes of Motion *Journal of Computational Chemistry* **14** 1259–1271.

De Loof H, S C Harvey, J P Segrest and R W Pastor 1991. Mean Field Stochastic Boundary Molecular Dynamics Simulation of a Phospholipid in a Membrane *Biochemistry* **30** 2099–2113

Espanol P and P B Warren 1995 Statistical Mechanics of Dissipative Particle Dynamics. *Europhysics Letters* **30**.191–196.

Essex J W, M M Hann and W G Richards 1994. Molecular Dynamics of a Hydrated Phospholipid Bilayer *Philosophical Transactions of the Royal Society of London* **B344**.239–260

Fincham D and Heyes D M 1982 Integration Algorithms in Molecular Dynamics. *CCP5 Quarterly* **6**.4–10

Gear C W 1971 *Numerical Initial Value Problems in Ordinary Differential Equations* Englewood Cliffs, NJ, Prentice Hall

Groot R D and T J Madden 1998. Dynamic Simulation of Diblock Copolymer Microphase Separation. *Journal of Chemical Physics* **108**:8713–8724

Groot R D, T J Madden and D J Tildesley 1999 On the Role of Hydrodynamic Interactions in Block Copolymer Microphase Separation. *Journal of Chemical Physics* **110** 9739–9749.

Groot R D and P B Warren 1997. Dissipative Particle Dynamics: Bridging the Gap Between Atomistic and Mesoscopic Simulation *Journal of Chemical Physics* **107** 4423–4435.

Helfand E 1984 Dynamics of Conformational Transitions in Polymers *Science* **226**:647–650.

Hockney R W 1970. The Potential Calculation and Some Applications *Methods in Computational Physics* **9** 136–211

Hoover W G 1985. Canonical Dynamics. Equilibrium Phase-space Distributions. *Physical Review* **A31**:1695–1697.

Humphreys D D, R A Friesner and B J Berne 1994. A Multiple Time-step Molecular Dynamics Algorithm for Macromolecules *Journal of Physical Chemistry* **98**:6885–6892.

Humphreys D D, R A Friesner and B J Berne 1995. Simulated Annealing of a Protein in a Continuum Solvent by Multiple Time-step Molecular Dynamics. *Journal of Physical Chemistry* **99**:10674–10685.

Humphreys, D D, R A Friesner and B J Berne 1996 A Multiple Time-step Molecular Dynamics Algorithm for Macromolecules *Journal of Physical Chemistry* **98**:6885–6892.

Jorgensen W L, R C Binning Jr and B Bigot 1981 Structures and Properties of Organic Liquids: *n*-Butane and 1,2-Dichloroethane and Their Conformational Equilibria *Journal of the American Chemical Society* **103**.4393–4399.

Kim K S, M A Moller, D J Tildesley and N Quirke 1994a. Molecular Dynamics Simulations of Langmuir-Blodgett Monolayers with Explicit Head-group Interactions *Molecular Simulation* **13**:77–99.

Kim K S, D J Tildesley and N Quirke 1994b. Molecular Dynamics of Langmuir-Blodgett Films: II. Bilayers. *Molecular Simulation* **13**.101–114

Koelman J M V A and P J Hoogerbrugge 1993 Dynamic Simulations of Hard-sphere Suspensions Under Steady Shear *Europhysics Letters* **21** 363–368.

Lavery R and H Sklenar 1988 The Definition of Generalized Helicoidal Parameters and of Axis Curvature for Irregular Nucleic Acids *Journal of Biomolecular Structure and Dynamics* **6**.63–91

Leach A R and T E Klein 1995 A Molecular Dynamics Study of the Inhibitors of Dihydrofolate Reductase by a Phenyl Triazine. *Journal of Computational Chemistry* **16**:1378–1393

Marcelja S 1973. Molecular Model for Phase Transition in Biological Membranes *Nature* **241**:451–453

Marcelja S 1974. Chain Ordering in Liquid Crystals. II. Structure of Bilayer Membranes. *Biochimica et Biophysica Acta* **367**:165–176

Nosé S 1984 A Molecular Dynamics Method for Simulations in the Canonical Ensemble. *Molecular Physics* **53**:255–268.

Pastor R W, R M Venable and M Karplus 1988. Brownian Dynamics Simulation of a Lipid Chain in a Membrane Bilayer *Journal of Chemical Physics* **89**:1112–1127.

Pearce L L and S C Harvey 1993. Langevin Dynamics Studies of Unsaturated Phospholipids in a Membrane Environment. *Biophysical Journal* **65**:1084–1092

Procacci P and B Berne 1994 Computer Simulation of Solid C_{60} Using Multiple Time-step Algorithms *Journal of Chemical Physics* **101**:2421–2431.

Rahman A 1964. Correlations in the Motion of Atoms in Liquid Argon. *Physical Review* **A136**:405–411.

Rahman A and F H Stillinger 1971. Molecular Dynamics Study of Liquid Water. *Journal of Chemical Physics* **55** 3336–3359

Robinson A J, W G Richards, P J Thomas and M M Hann 1994. Head Group and Chain Behaviour in Biological Membranes—A Molecular Dynamics Simulation *Biophysical Journal* **67**.2345–2354.

Rubinstein R Y 1981. *Simulation and Monte Carlo Methods* New York, John Wiley & Sons.

Ryckaert J P, G Cicotti and H J C Berensden 1977 Numerical Integration of the Cartesian Equations of Motion of a System with Constraints Molecular Dynamics of *n*-Alkanes *Journal of Computational Physics* **23**:327–341

Seelig A and J Seelig 1974 The Dynamics Structure of Fatty Acyl Chains in a Phospholipid Bilayer Measured by Deuterium Magnetic Resonance *Biochemistry* **13**:4839–4845

Stouch T R 1993 Lipid Membrane Structure and Dynamics Studied by All-atom Molecular Dynamics Simulations of Hydrated Phospholipid Bilayers. *Molecular Simulation* **10**:335–362.

Streett W B, D Tildesley and G Saville 1978 Multiple Time-step Methods in Molecular Dynamics *Molecular Physics* **35**:639–648.

Swindoll R D and J M Haile 1984. A Multiple Time-step Method for Molecular Dynamics Simulations of Fluids of Chain Molecules. *Journal of Compututational Physics* **53**.289–298

Swope W C, H C Anderson, P H Berens and K R Wilson 1982. A Computer Simulation Method for the Calculation of Equilibrium Constants for the Formation of Physical Clusters of Molecules: Application to Small Water Clusters *Journal of Chemical Physics* **76**.637–649

Tobias D J and C L Brooks III 1988 Molecular Dynamics with Internal Coordinate Constraints. *Journal of Chemical Physics* **89**:5115–5126

Tobias D J, K Tu and M L Klein 1997. Atomic-scale Molecular Dynamics Simulations of Lipid Membranes *Current Opinion in Colloid and Interface Science* **2**:15–26

Tuckerman M, B J Berne and G J Martyna 1992. Reversible Multiple Time Scale Molecular Dynamics. *Journal of Chemical Physics* **97**.1990–2001

van der Ploeg P and H J C Berendsen 1982. Molecular Dynamics Simulation of a Bilayer Membrane. *Journal of Chemical Physics* **76**:3271–3276

van Gunsteren W F and H J C Berendsen 1982 Algorithms for Brownian Dynamics. *Molecular Physics* **45**.637–547.

van Gunsteren W F, H J C Berendsen and J A C Rullmann 1981. Stochastic Dynamics for Molecules with Constraints. Brownian Dynamics of *n*-Alkanes. *Molecular Physics* **44**:69–95.

Verlet L 1967. Computer 'Experiments' on Classical Fluids. I Thermodynamical Properties of Lennard-Jones Molecules *Physical Review* **159**:98–103

Watanabe M and M Karplus 1993 Dynamics of Molecules with Internal Degrees of Freedom by Multiple Time-step Methods. *Journal of Chemical Physics* **99**:8063–8074

Widmalm G and R W Pastor 1992. Comparison of Langevin and Molecular Dynamics Simulations. *Journal of the Chemical Society Faraday Transactions* **88**:1747–1754

Woodcock L V 1971 Isothermal Molecular Dynamics Calculations for Liquid Salts *Chemical Physics Letters* **10**:257–261.

Yun-Yu S, W Lu and W F van Gunsteren 1988 On the Approximation of Solvent Effects on the Conformation and Dynamics of Cyclosporin A by Stochastic Dynamics Simulation Techniques *Molecular Simulation* **1** 369–383

Zhou R and B J Berne 1995 A New Molecular Dynamics Method Combining the Reference System Propagator Algorithm with a Fast Multipole Method for Simulating Proteins and Other Complex Systems. *Journal of Chemical Physics* **103** 9444–9459.

CHAPTER EIGHT

Monte Carlo Simulation Methods

8.1 Introduction

The Monte Carlo simulation method occupies a special place in the history of molecular modelling, as it was the technique used to perform the first computer simulation of a molecular system. A Monte Carlo simulation generates configurations of a system by making random changes to the positions of the species present, together with their orientations and conformations where appropriate. Many computer algorithms are said to use a 'Monte Carlo' method, meaning that some kind of random sampling is employed. In molecular simulations 'Monte Carlo' is almost always used to refer to methods that use a technique called *importance sampling*. Importance sampling methods are able to generate states of low energy, as this enables properties to be calculated accurately We can calculate the potential energy of each configuration of the system, together with the values of other properties, from the positions of the atoms. The Monte Carlo method thus samples from a $3N$-dimensional space of the positions of the particles. There is no momentum contribution in a Monte Carlo simulation, in contrast to a molecular dynamics simulation. How then can a Monte Carlo simulation be used to calculate thermodynamic quantities, given that phase space is $6N$-dimensional?

To resolve this difficulty, let us return to the canonical ensemble partition, Q, which for a system of N identical particles of mass m can be written:

$$Q_{NVT} = \frac{1}{N!}\frac{1}{h^{3N}}\iint d\mathbf{p}^N\, d\mathbf{r}^N \exp\left[-\frac{\mathscr{H}(\mathbf{p}^N,\mathbf{r}^N)}{k_B T}\right] \tag{8.1}$$

The factor $N!$ disappears when the particles are no longer indistinguishable. $\mathscr{H}(\mathbf{p}^N,\mathbf{r}^N)$ is the Hamiltonian that corresponds to the total energy of the system. The value of the Hamiltonian depends upon the $3N$ positions and $3N$ momenta of the particles in the system (one position and one momentum for each of the three coordinates of each particle). The Hamiltonian can be written as the sum of the kinetic and potential energies of the system:

$$\mathscr{H}(\mathbf{p}^N,\mathbf{r}^N) = \sum_{i=1}^{N}\frac{|\mathbf{p}_i|^2}{2m} + \mathscr{V}(\mathbf{r}^N) \tag{8.2}$$

The crucial point to recognise is that the double integral in Equation (8.1) can be separated into two separate integrals, one over positions and the other over the momenta:

$$Q_{NVT} = \frac{1}{N!}\frac{1}{h^{3N}}\int d\mathbf{p}^N \exp\left[-\frac{|\mathbf{p}|^2}{2mk_B T}\right]\int d\mathbf{r}^N \exp\left[-\frac{\mathscr{V}(\mathbf{r}^N)}{k_B T}\right] \tag{8.3}$$

This separation is possible only if the potential energy function, $\mathscr{V}(\mathbf{r}^N)$, is not dependent upon the velocities (this is a safe assumption for almost all potential functions in common use). The integral over the momenta can now be performed analytically, the result being:

$$\int d\mathbf{p}^N \exp\left[-\frac{|\mathbf{p}|^2}{2mk_\mathrm{B}T}\right] = (2\pi mk_\mathrm{B}T)^{3N/2} \tag{8.4}$$

The partition function can thus be written:

$$Q_{NVT} = \frac{1}{N!}\left(\frac{2\pi mk_\mathrm{B}T}{h^2}\right)^{3N/2} \int d\mathbf{r}^N \exp\left(-\frac{\mathscr{V}(\mathbf{r}^N)}{k_\mathrm{B}T}\right) \tag{8.5}$$

The integral over the positions is often referred to as the *configurational integral*, Z_{NVT}:

$$Z_{NVT} = \int d\mathbf{r}^N \exp\left(-\frac{\mathscr{V}(\mathbf{r}^N)}{k_\mathrm{B}T}\right) \tag{8.6}$$

In an ideal gas there are no interactions between the particles and so the potential energy function, $\mathscr{V}(\mathbf{r}^N)$, equals zero. $\exp(-\mathscr{V}(\mathbf{r}^N)/k_\mathrm{B}T)$ is therefore equal to 1 for every gas particle in the system. The integral of 1 over the coordinates of each atom is equal to the volume, and so for N ideal gas particles the configurational integral is given by V^N ($V \equiv$ volume). This leads to the following result for the canonical partition function of an ideal gas:

$$Q_{NVT} = \frac{V^N}{N!}\left(\frac{2\pi k_\mathrm{B}Tm}{h^2}\right)^{3N/2} \tag{8.7}$$

This is often written in terms of the *de Broglie thermal wavelength*, Λ:

$$Q_{NVT} = \frac{V^N}{N!\Lambda^{3N}} \tag{8.8}$$

where $\Lambda = \sqrt{h^2/2\pi k_\mathrm{B}Tm}$.

By combining Equations (8.4) and (8.6) we can see that the partition function for a 'real' system has a contribution due to ideal gas behaviour (the momenta) and a contribution due to the interactions between the particles. Any deviations from ideal gas behaviour are due to interactions within the system as a consequence of these interactions. This enables us to write the partition function as:

$$Q_{NVT} = Q_{NVT}^{\text{ideal}} Q_{NVT}^{\text{excess}} \tag{8.9}$$

The excess part of the partition function is given by:

$$Q_{NVT}^{\text{excess}} = \frac{1}{V^N}\int d\mathbf{r}^N \exp\left[-\frac{\mathscr{V}(\mathbf{r}^N)}{k_\mathrm{B}T}\right] \tag{8.10}$$

A consequence of writing the partition function as a product of a real gas and an ideal gas part is that thermodynamic properties can be written in terms of an ideal gas value and an excess value. The ideal gas contributions can be determined analytically by integrating over the momenta. For example, the Helmholtz free energy is related to the canonical partition function by:

$$A = -k_\mathrm{B}T \ln Q_{NVT} \tag{8.11}$$

Writing the partition function as the product, Equation (8.9), leads to:

$$A = A^{\text{ideal}} + A^{\text{excess}} \tag{8.12}$$

The important conclusion is that all of the deviations from ideal gas behaviour are due to the presence of interactions between the atoms in the system, as calculated using the potential energy function. This energy function is dependent only upon the positions of the atoms and not their momenta, and so a Monte Carlo simulation is able to calculate the excess contributions that give rise to deviations from ideal gas behaviour.

8.2 Calculating Properties by Integration

Having established that we can indeed explore configurational phase space and derive useful thermodynamic properties, let us consider how we might achieve this in practice. For example, the average potential energy can, in principle at least, be determined by evaluating the integral:

$$\langle \mathscr{V}(\mathbf{r}^N) \rangle = \int d\mathbf{r}^N \mathscr{V}(\mathbf{r}^N) \rho(\mathbf{r}^N) \tag{8.13}$$

This is a multidimensional integral over the $3N$ degrees of freedom of the N particles in the system. $\rho(\mathbf{r}^N)$ is the probability of obtaining the configuration $\mathbf{r}^N$ and is given by

$$\rho(\mathbf{r}^N) = \frac{\exp[-\mathscr{V}(\mathbf{r}^N)/k_B T]}{Z} \tag{8.14}$$

The denominator, Z, is the configurational integral (Equation (8.6)). For the potential functions commonly used in molecular modelling, it is not possible to evaluate these integrals analytically. However, we could attempt to obtain values for the integrals using numerical methods. One simple numerical integration method is the trapezium rule. This approximates the integral as a series of trapeziums between the two limits, as illustrated for a one-dimensional problem in Figure 8.1. In this case we have divided the integral into ten trapeziums, which requires eleven function evaluations. Simpson's rule involves a similar procedure and may provide a more accurate value of the integral [Stephenson 1973]. For a function of two variables ($f(x,y)$), it is necessary to square the number of function

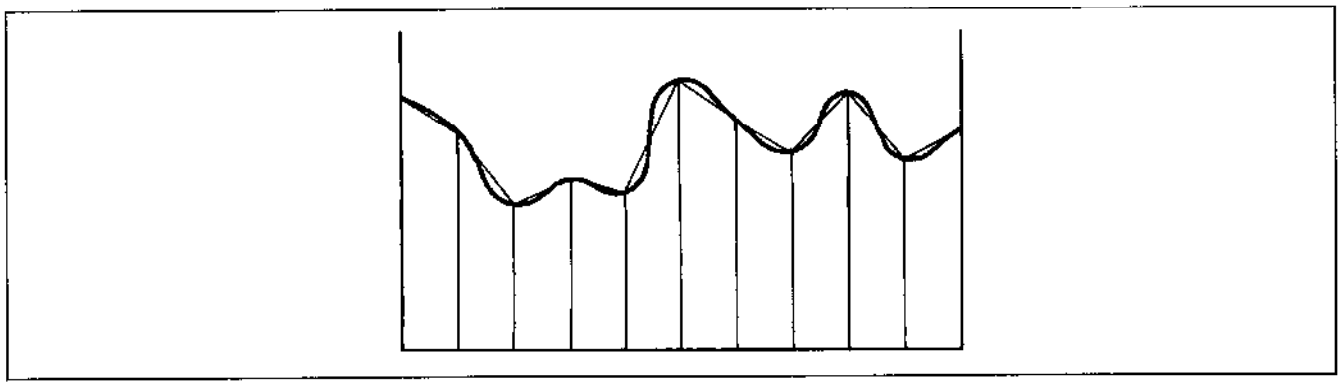


Fig 8 1 Evaluation of a one-dimensional integral using the trapezium rule The area under the curve is approximated as the sum of the areas of the trapeziums

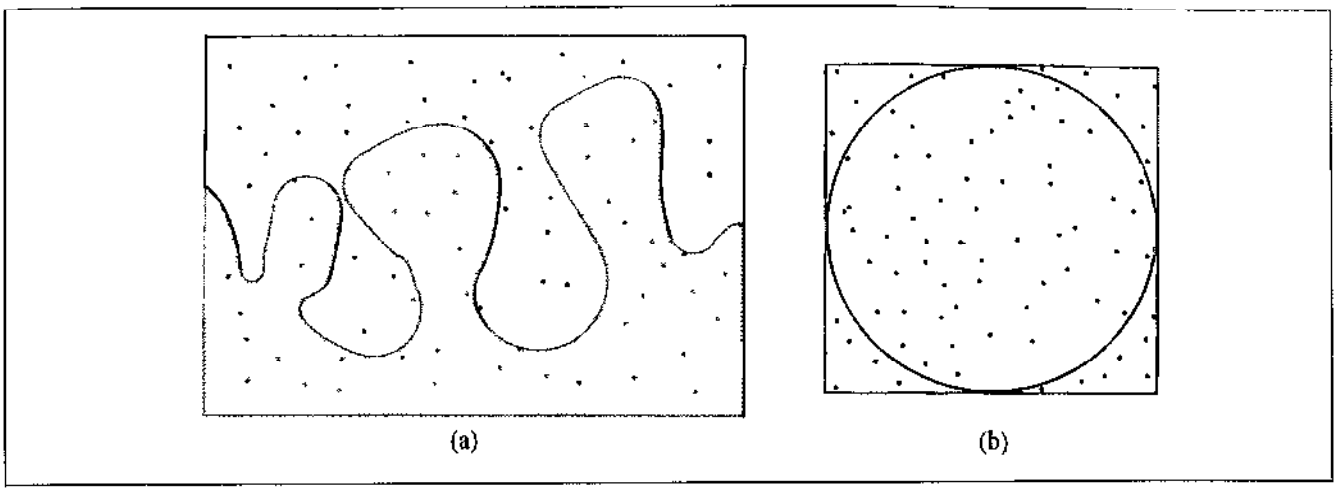


Fig. 8 2 Simple Monte Carlo integration (a) The shaded area under the irregular curve equals the ratio of the number of random points under the curve to the total number of points, multiplied by the area of the bounding area (b) An estimate of π can be obtained by generating random numbers within the square. π then equals the number of points within the circle divided by the total number of points within the square, multiplied by 4

evaluations required. For a $3N$-dimensional integral the total number of function evaluations required to determine the integral would be m^{3N}, where m is the number of points needed to determine the integral in each dimension. This number is enormous even for very small numbers of particles. For example, with just 50 particles and three points per dimension, a total of 3^{150} ($\sim 10^{71}$) evaluations would be required. Integration using the trapezium rule or Simpson's rule is clearly not a feasible approach.

We could consider a random method as a possible alternative. The general principle can be illustrated using the function shown in Figure 8.2. To determine the area under the curve in Figure 8.2 a series of random points would be generated within the bounding area. The area under the curve is then calculated by multiplying the bounding area A by the ratio of the number of trial points that lie under the curve to the total number of points generated. An estimate of π can be determined in this way, as illustrated in Figure 8.2.

To calculate the partition function for a system of N atoms using this simple Monte Carlo integration method would involve the following steps:

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, $\mathscr{V}(\mathbf{r}^N)$.
3. From the potential energy, calculate the Boltzmann factor, $\exp(-\mathscr{V}(\mathbf{r}^N)k_B T)$.
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. After a number, N_{trial}, of iterations, the mean value of the potential energy would be calculated using:

$$\langle \mathscr{V}(\mathbf{r}^N) \rangle = \frac{\sum_{i=1}^{N_{\text{trial}}} \mathscr{V}_i(\mathbf{r}^N) \exp[-\mathscr{V}_i(\mathbf{r}^N)/k_B T]}{\sum_{i=1}^{N_{\text{trial}}} \exp[-\mathscr{V}_i(\mathbf{r}^N)/k_B T]} \tag{8 15}$$

Unfortunately, this is not a feasible approach for calculating thermodynamic properties due to the large number of configurations that have extremely small (effectively zero) Boltzmann

factors caused by high-energy overlaps between the particles. This reflects the nature of the phase space, most of which corresponds to non-physical configurations with very high energies. Only a very small proportion of the phase space corresponds to low-energy configurations where there are no overlapping particles and where the Boltzmann factor has an appreciable value. These low-energy regions coincide with the physically observed phases such as solid, liquid, etc

One way around this impasse is to generate configurations that make a large contribution to the integral (8 15), which is the strategy adopted in importance sampling and which is the essence of the method described by Metropolis, Rosenbluth, Rosenbluth, Teller and Teller in 1953 [Metropolis *et al.* 1953]. For many thermodynamic properties of a molecular system, those states with a high probability ρ are also the ones that make a significant contribution to the integral (there are some notable exceptions to this, such as the free energy). The Metropolis method has become so widely adopted that in the simulation and molecular modelling communities it is usually referred to as 'the Monte Carlo method'. Fortunately, there is rarely any confusion with the simple Monte Carlo methods. The crucial feature of the Metropolis approach 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 $\exp(-\mathscr{V}(\mathbf{r}^N)/k_B T)$ and then counts each of them equally. By contrast, the simple Monte Carlo integration method generates states with equal probability (both high- and low-energy) and then assigns them a weight $\exp(-\mathscr{V}(\mathbf{r}^N)/k_B T)$.

8.3 Some Theoretical Background to the Metropolis Method

The Metropolis algorithm generates a *Markov chain* of states. A Markov chain satisfies the following two conditions:

1. The outcome of each trial depends only upon the preceding trial and not upon any previous trials.
2. Each trial belongs to a finite set of possible outcomes.

Condition (1) provides a clear distinction between the molecular dynamics and Monte Carlo methods, for in a molecular dynamics simulation all of the states are connected in time. Suppose the system is in a state m. We denote the probability of moving to state n as π_{mn}. The various π_{mn} can be considered to constitute an $N \times N$ matrix $\boldsymbol{\pi}$ (the transition matrix), where N is the number of possible states. Each row of the transition matrix sums to 1 (i.e. the sum of the probabilities π_{mn} for a given m equals 1). The probability that the system is in a particular state is represented by a probability vector $\boldsymbol{\rho}$:

$$\boldsymbol{\rho} = (\rho_1, \rho_2, \ldots, \rho_m, \rho_n, \ldots, \rho_N) \tag{8.16}$$

Thus ρ_1 is the probability that the system is in state 1 and ρ_m the probability that the system is in state m. If $\boldsymbol{\rho}(1)$ represents the initial (randomly chosen) configuration, then the probability of the second state is given by:

$$\boldsymbol{\rho}(2) = \boldsymbol{\rho}(1)\boldsymbol{\pi} \tag{8.17}$$

The probability of the third state is:

$$\boldsymbol{\rho}(3) = \boldsymbol{\rho}(2)\boldsymbol{\pi} = \boldsymbol{\rho}(1)\boldsymbol{\pi}\boldsymbol{\pi} \tag{8.18}$$

The equilibrium distribution of the system can be determined by considering the result of applying the transition matrix an infinite number of times. This limiting distribution of the Markov chain is given by $\boldsymbol{\rho}_{\text{limit}} = \lim_{N\to\infty} \boldsymbol{\rho}(1)\boldsymbol{\pi}^N$.

One feature of the limiting distribution is that it is independent of the initial guess $\rho(1)$. The limiting or equilibrium distribution for a molecular or atomic system is one in which the probabilities of each state are proportional to the Boltzmann factor. We can illustrate the use of the probability distribution and the transition matrix by considering a two-level system in which the energy levels are such that the ratio of the Boltzmann factors is 2 : 1. The expected limiting distribution thus corresponds to a configuration vector $(\frac{2}{3}, \frac{1}{3})$. The following transition matrix enables the limiting distribution to be achieved.

$$\boldsymbol{\pi} = \begin{pmatrix} 0.5 & 0.5 \\ 1 & 0 \end{pmatrix} \tag{8.19}$$

We can illustrate the use of this transition matrix as follows. Suppose the initial probability vector is $(1, 0)$ and so the system starts with a 100% probability of being in state 1 and no probability of being in state 2. Then the second state is given by:

$$\boldsymbol{\rho}(2) = (1 \quad 0)\begin{pmatrix} 0.5 & 0.5 \\ 1 & 0 \end{pmatrix} = (0.5 \quad 0.5) \tag{8.20}$$

The third state is $\boldsymbol{\rho}(3) = (0.75, 0.25)$. Successive applications of the transition matrix give the limiting distribution $(2/3, 1/3)$.

When the limiting distribution is reached then application of the transition matrix must return the same distribution back:

$$\boldsymbol{\rho}_{\text{limit}} = \boldsymbol{\rho}_{\text{limit}}\boldsymbol{\pi} \tag{8.21}$$

Thus, if an ensemble can be prepared that is at equilibrium, then one Metropolis Monte Carlo step should return an ensemble that is still at equilibrium A consequence of this is that the elements of the probability vector for the limiting distribution must satisfy:

$$\sum_m \rho_m \pi_{mn} = \rho_n \tag{8.22}$$

This can be seen to hold for our simple two-level example:

$$(2/3 \quad 1/3)\begin{pmatrix} 1/2 & 1/2 \\ 1 & 0 \end{pmatrix} = (2/3 \quad 1/3) \tag{8.23}$$

We will henceforth use the symbol ρ to refer to the limiting distribution

Closely related to the transition matrix is the *stochastic matrix*, whose elements are labelled α_{mn}. This matrix gives the probability of choosing the two states m and n between which the move is to be made. It is often known as the *underlying matrix* of the Markov chain. If the probability of accepting a trial move from m to n is p_{mn} then the probability of making a transition from m to n (π_{mn}) is given by multiplying the probability of choosing states m

and n (α_{mn}) by the probability of accepting the trial move (p_{mn}):

$$\pi_{mn} = \alpha_{mn} p_{mn} \tag{8.24}$$

It is often assumed that the stochastic matrix $\boldsymbol{\alpha}$ is symmetrical (i.e. the probability of choosing the states m and n is the same whether the move is made from m to n or from n to m) If the probability of state n is greater than that of state m in the limiting distribution (i.e. if the Boltzmann factor of n is greater than that of m because the energy of n is lower than the energy of m) then in the Metropolis recipe, the transition matrix element π_{mn} for progressing from m to n equals the probability of selecting the two states in the first place (i.e. $\pi_{mn} = \alpha_{mn}$ if $\rho_n \geq \rho_m$). If the Boltzmann weight of the state n is less than that of state m, then the probability of permitting the transition is given by multiplying the stochastic matrix element α_{mn} by the ratio of the probabilities of the state n to the previous state m. This can be written:

$$\pi_{mn} = \alpha_{mn} \quad (\rho_n \geq \rho_m) \tag{8.25}$$

$$\pi_{mn} = \alpha_{mn}(\rho_n/\rho_m) \quad (\rho_n < \rho_m) \tag{8.26}$$

These two conditions apply if the initial and final states m and n are different. If m and n are the same state, then the transition matrix element is calculated from the fact that the rows of the stochastic matrix sum to 1:

$$\pi_{mm} = 1 - \sum_{m \neq n} \pi_{mn} \tag{8.27}$$

Let us now try to reconcile the Metropolis algorithm as outlined in Section 6 1.3 with the more formal approach that we have just developed. We recall that in the Metropolis method a new configuration n is accepted if its energy is lower than the original state m. If the energy is higher, however, then we would like to choose the move with a probability according to Equation (8 24) This is achieved by comparing the Boltzmann factor $\exp(-\Delta\mathscr{V}(\mathbf{r}^N)/k_B T)$ ($\Delta\mathscr{V}(\mathbf{r}^N) = [\mathscr{V}(\mathbf{r}^N)_n - \mathscr{V}(\mathbf{r}^N)_m]$) to a random number between 0 and 1. If the Boltzmann factor is greater than the random number then the new state is accepted. If it is smaller then the new state is rejected. Thus if the energy of the new state (n) is very close to that of the old state (m) then the Boltzmann factor of the energy difference will be very close to 1, and so the move is likely to be accepted. If the energy difference is very large, however, then the Boltzmann factor will be close to zero and the move is unlikely to be accepted.

The Metropolis method is derived by imposing the condition of microscopic reversibility: at equilibrium the transition between two states occurs at the same rate. The rate of transition from a state m to a state n equals the product of the population (ρ_m) and the appropriate element of the transition matrix (π_{mn}). Thus, at equilibrium we can write:

$$\pi_{mn}\rho_m = \pi_{nm}\rho_n \tag{8 28}$$

The ratio of the transition matrix elements thus equals the ratio of the Boltzmann factors of the two states:

$$\frac{\pi_{mn}}{\pi_{nm}} = \exp[-(\mathscr{V}(\mathbf{r}^N)_n - \mathscr{V}(\mathbf{r}^N)_m)/k_B T] \tag{8.29}$$

8.4 Implementation of the Metropolis Monte Carlo Method

A Monte Carlo program to simulate an atomic fluid is quite simple to construct 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 If the random number generator produces numbers (ξ) in the range 0 to 1, moves in both positive and negative directions are possible if the coordinates are changed as follows:

$$x_{\text{new}} = x_{\text{old}} + (2\xi - 1)\delta r_{\max} \tag{8.30}$$

$$y_{\text{new}} = y_{\text{old}} + (2\xi - 1)\delta r_{\max} \tag{8.31}$$

$$z_{\text{new}} = z_{\text{old}} + (2\xi - 1)\delta r_{\max} \tag{8.32}$$

A unique random number is generated for each of the three directions x, y and z. $\delta r_{\max}$ is the maximum possible displacement in any direction The energy of the new configuration is then calculated; this 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. As a consequence, the neighbour list used by a Monte Carlo simulation must contain *all* the neighbours of each atom, because it is necessary to identify all the atoms which interact with the moving atom (recall that in molecular dynamics the neighbour list for each atom contains only neighbours with a higher index). Proper account should be taken of periodic boundary conditions and the minimum image convention when generating new configurations and calculating their energies. 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, $\exp(-\Delta\mathscr{V}/k_{\text{B}}T)$, 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:

$$\text{rand}(0,1) \leq \exp(-\Delta\mathscr{V}(\mathbf{r}^N)/k_{\text{B}}T) \tag{8.33}$$

The size of the move at each iteration is governed by the maximum displacement, $\delta r_{\max}$. 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 explored very slowly. Too large a value of $\delta r_{\max}$ and many trial moves will be rejected because they lead to unfavourable 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 $\delta r_{\max}$ is reduced.

As an alternative to the random selection of particles it is possible to move the atoms sequentially (this requires one fewer call to the random number generator per iteration) Alternatively, 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 molecular dynamics simulation, a Monte Carlo simulation comprises an equilibration phase followed by a production phase During equilibration, appropriate thermodynamic and structural quantities such as the total energy (and the partitioning of the energy among the various components), mean square displacement and order parameters (as appropriate) are monitored until they achieve stable values, whereupon the production phase can commence. In a Monte Carlo simulation from the canonical ensemble, the temperature and volume are, of course, fixed. In a constant pressure simulation the volume will change and should therefore also be monitored to ensure that a stable system density is achieved.

8.4.1 Random Number Generators

The random number generator at the heart of every Monte Carlo simulation program is accessed a very large number of times, not only to generate new configurations but also to decide whether a given move should be accepted or not. Random number generators are also used in other modelling applications; for example, in a molecular dynamics simulation the initial velocities are normally assigned using a random number generator. The numbers produced by a random number generator are not, in fact, truly random; the same sequence of numbers should always be generated when the program is run with the same initial conditions (if not, then a serious error in the hardware or software must be suspected!). The sequences of numbers are thus often referred to as 'pseudo-random' numbers as they possess the statistical properties of 'true' sequences of random numbers Most random number generators are designed to generate different sequences of numbers if a different 'seed' is provided. In this way, several independent runs can be performed using different seeds One simple strategy is to use the time and/or date as the seed; this is information that can often be obtained automatically by the program from the computer's operating system

The numbers produced by a random number generator should satisfy certain statistical properties. This requirement usually supersedes the need for a computationally very fast algorithm as other parts of a Monte Carlo simulation take much more time (such as calculating the change in energy). One useful and simple test of a random number generator is to break a sequence of random numbers into blocks of k numbers, which are taken to be coordinates in a k-dimensional space. A good random number should give a random distribution of points. Many of the common generators do not satisfy this test because the points lie on a plane or because they show clear correlations [Sharp and Bays 1992].

The *linear congruential* method is widely used for generating random numbers. Each number in the sequence is generated by taking the previous number, multiplying by a constant (the multiplier, a), adding a second constant (the increment, b), and taking the remainder when dividing by a third constant (the modulus, m) The first value is the seed, supplied by the user. Thus:

$$\xi[1] = \text{seed} \tag{8.34}$$

$$\xi[i] = \text{MOD}\{(\xi[i-1] \times a + b), m\} \tag{8.35}$$

The MOD function returns the remainder when the first argument is divided by the second (for example, MOD(14,5) equals 4). If the constants are chosen carefully, the linear congruential method generates all possible integers between 0 and $m-1$, and the period (i.e. the number of iterations before the sequence starts to repeat itself) will be equal to

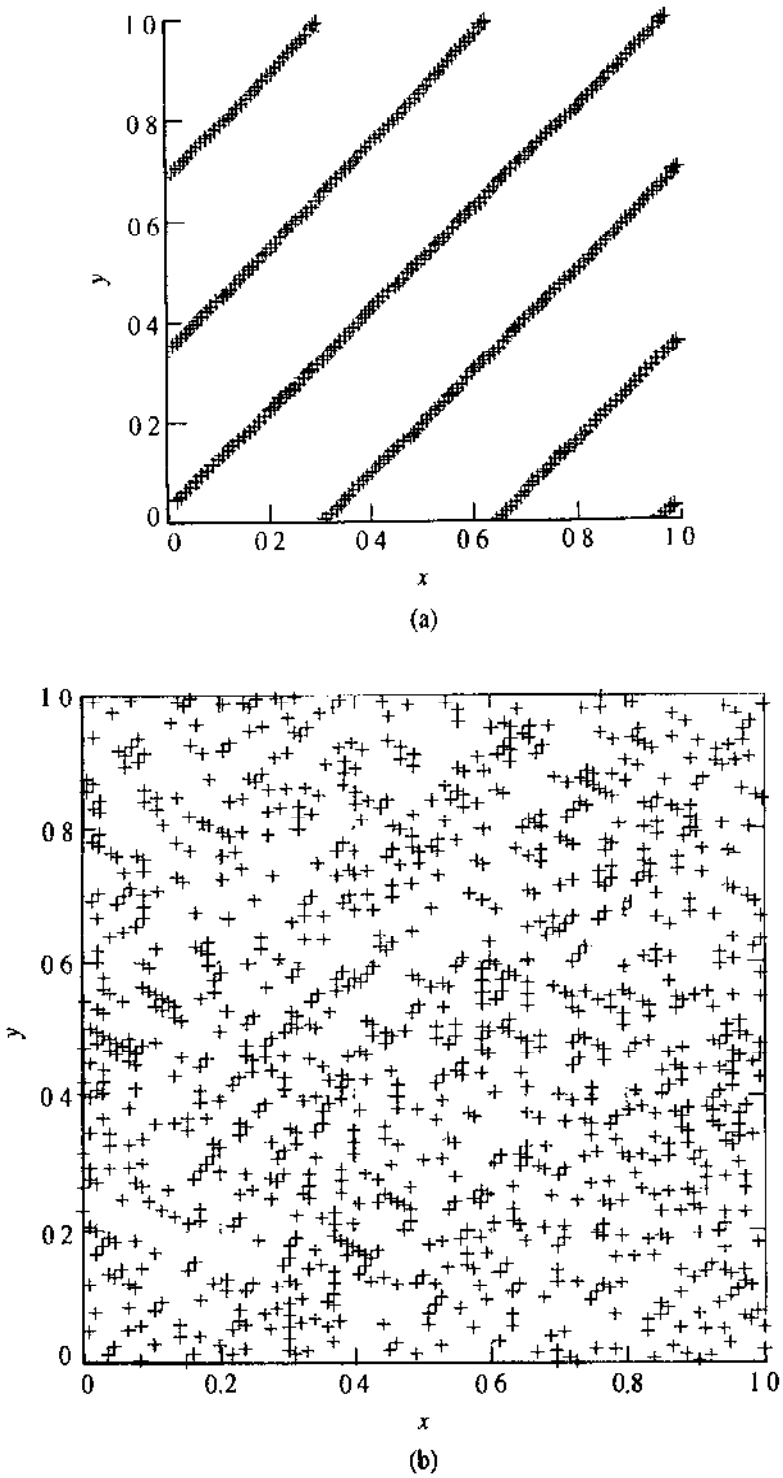


Fig. 8 3 Two 'random' distributions obtained by plotting pairs of values from a linear congruential random generator The distribution (a) was obtained using $m = 32\,769$, $a = 10\,924$, $b = 11\,830$ The distribution (b) was obtained using $m = 6075$, $a = 106$, $b = 1283$. Data from [Sharp and Bays 1992]

the modulus. The period cannot of course be greater than m. The linear congruential method generates integral values, which can be converted to real numbers between 0 and 1 by dividing by m. The modulus is often chosen to be the largest prime number that can be represented in a given number of bits (usually chosen to be the number of bits per word; $2^{31} - 1$ is thus a common choice on a 32-bit machine).

Although popular, by virtue of the ease with which it can be programmed, the linear congruential method does not satisfy all of the requirements that are now regarded as important in a random number generator. For example, the points obtained from a linear congruential generator lie on $(k-1)$-dimensional planes rather than uniformly filling up the space Indeed, if the constants a, b and m are chosen inappropriately then the linear congruential method can give truly terrible results, as shown in Figure 8.3. One random number generator that is claimed to perform well in all of the standard tests is that of G Marsaglia, which is described in Appendix 8 1.

8.5 Monte Carlo Simulation of Molecules

The Monte Carlo 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 systems, 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 neighbours and thus a high rejection rate.

8.5.1 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 Monte Carlo step Translations are usually described in terms of the position of the centre 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 (x, y or z) and to rotate about the chosen axis by a randomly chosen angle $\delta\omega$, chosen to lie within the maximum angle variation, $\delta\omega_{\max}$ [Barker and Watts 1969]. The rotation is achieved by applying routine 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 calculated as follows:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\delta\omega & \sin\delta\omega \\ 0 & -\sin\delta\omega & \cos\delta\omega \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{8 36}$$

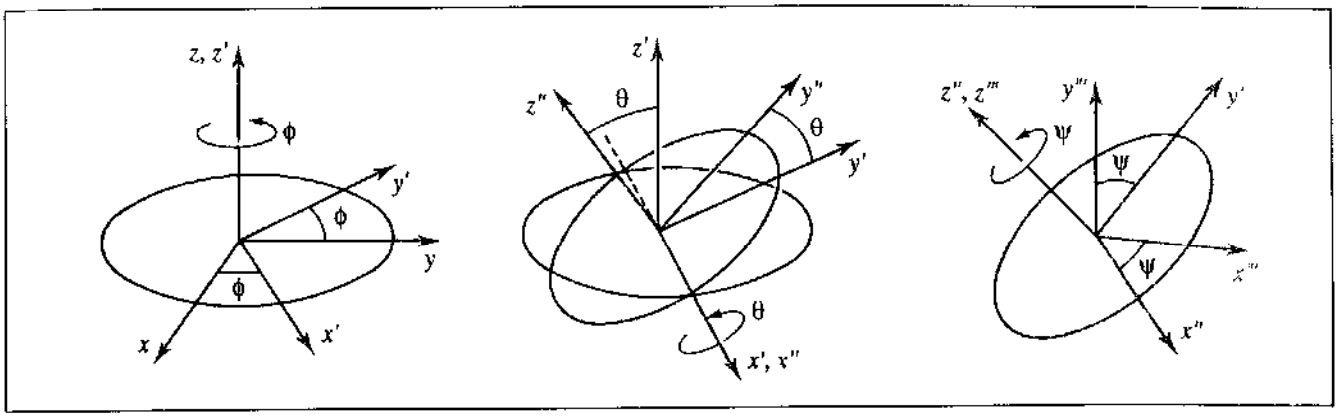


Fig. 8.4: The Euler angles ϕ, θ and ψ

The *Euler angles* are often used to describe the orientations of a molecule. There are three Euler angles; ϕ, θ and ψ. ϕ is a rotation about the Cartesian z axis; this has the effect of moving the x and y axes. θ is a rotation about the new x axis. Finally, ψ is a rotation about the new z axis (Figure 8.4). If the Euler angles are randomly changed by small amounts $\delta\phi$, $\delta\theta$ and $\delta\psi$ then a vector $\mathbf{v}_{\text{old}}$ is moved according to the following matrix equation:

$$\mathbf{v}_{\text{new}} = \mathbf{A}\mathbf{v}_{\text{old}} \tag{8.37}$$

where the matrix $\mathbf{A}$ is

$$\begin{pmatrix} \cos\delta\phi\cos\delta\psi - \sin\delta\phi\cos\delta\theta\sin\delta\psi & \sin\delta\phi\cos\delta\psi + \cos\delta\phi\cos\delta\theta\sin\delta\psi & \sin\delta\theta\sin\delta\psi \\ -\cos\delta\phi\sin\delta\psi - \sin\delta\phi\cos\delta\theta\cos\delta\psi & -\sin\delta\phi\sin\delta\psi + \cos\delta\phi\cos\delta\theta\cos\delta\psi & \sin\delta\theta\cos\delta\psi \\ \sin\delta\phi\sin\delta\theta & -\cos\delta\phi\sin\delta\theta & \cos\delta\theta \end{pmatrix} \tag{8.38}$$

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 8.5).

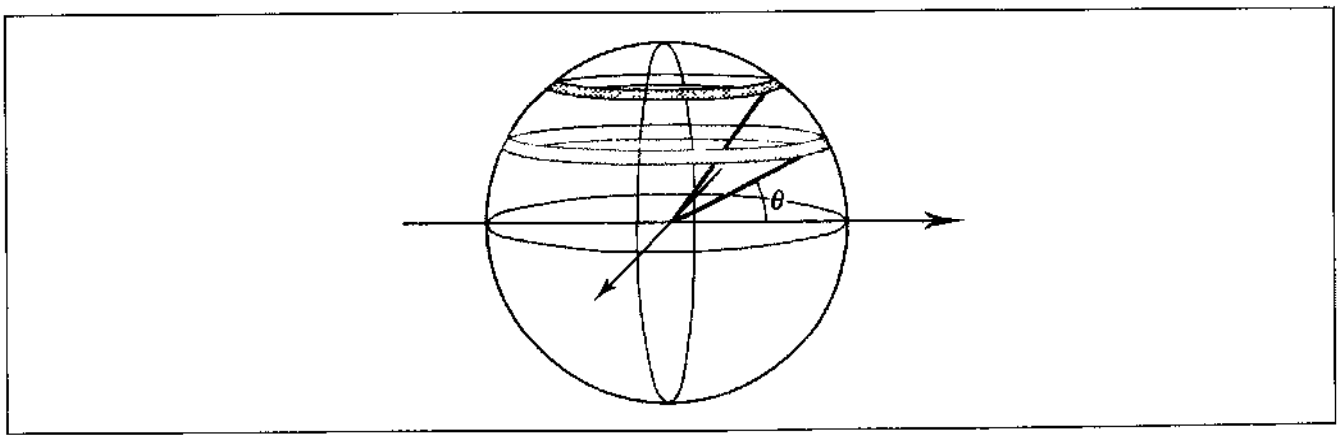


Fig. 8.5: To achieve a uniform distribution of points over the surface of a sphere it is necessary to sample from $\cos\theta$ rather than θ. If the sampling is uniform in θ then the number of points per unit area increases with θ, leading to an uneven distribution over the sphere.

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

$$\phi_{\text{new}} = \phi_{\text{old}} + 2(\xi - 1)\delta\phi_{\max} \tag{8.39}$$

$$\cos\theta_{\text{new}} = \cos\theta_{\text{old}} + (2\xi - 1)\delta(\cos\theta)_{\max} \tag{8.40}$$

$$\psi_{\text{new}} = \psi_{\text{old}} + 2(\xi - 1)\delta\psi_{\max} \tag{8.41}$$

The alternative is to sample in θ and to modify the acceptance or rejection criteria as follows:

$$\theta_{\text{new}} = \theta_{\text{old}} + (2\xi - 1)\delta\theta_{\max} \tag{8.42}$$

$$\frac{\rho_{\text{new}}}{\rho_{\text{old}}} = \exp(-\Delta\mathscr{V}/k_{\text{B}}T)\frac{\sin\theta_{\text{new}}}{\sin\theta_{\text{old}}} \tag{8.43}$$

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 (sine and cosine for each of the three Euler angles). 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:

$$q_0 = \cos\tfrac{1}{2}\theta\cos\tfrac{1}{2}(\phi + \psi) \tag{8.44}$$

$$q_1 = \sin\tfrac{1}{2}\theta\cos\tfrac{1}{2}(\phi + \psi) \tag{8.45}$$

$$q_2 = \sin\tfrac{1}{2}\theta\sin\tfrac{1}{2}(\phi + \psi) \tag{8.46}$$

$$q_3 = \cos\tfrac{1}{2}\theta\sin\tfrac{1}{2}(\phi + \psi) \tag{8.47}$$

The Euler angle rotation matrix can then be written

$$\mathbf{A} = \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix} \tag{8.48}$$

To generate a new 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 [Vesely 1982]:

1. Generate pairs of random numbers (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.

The introduction of an orientational component as well as a translational component increases the number of maximum displacement parameters that determine the acceptance ratio. It is important to check that the desired acceptance ratio is achieved, and also that an appropriate proportion of orientational and translational moves are made. Trial and error is often the most effective way to find the best combination of parameters.

8.5.2 Monte Carlo Simulations of Flexible Molecules

Monte Carlo simulations of flexible molecules are often difficult to perform successfully unless the system is small, or some of the internal degrees of freedom are frozen out, or special models or methods are employed. The simplest way to generate a new configuration of a flexible molecule is to perform random changes to the Cartesian coordinates of 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. For example, even small movements away from an equilibrium bond length will cause a large increase in the energy. One obvious tactic is to freeze out some of the internal degrees of freedom, usually the 'hard' degrees of freedom such as the bond lengths and the bond angles. Such algorithms have been extensively used to investigate small molecules such as butane. However, 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 8.6. The rigid bond and rigid angle approximation must be used with care, for freezing out some of the internal degrees of freedom can affect the distributions of other internal degrees of freedom.

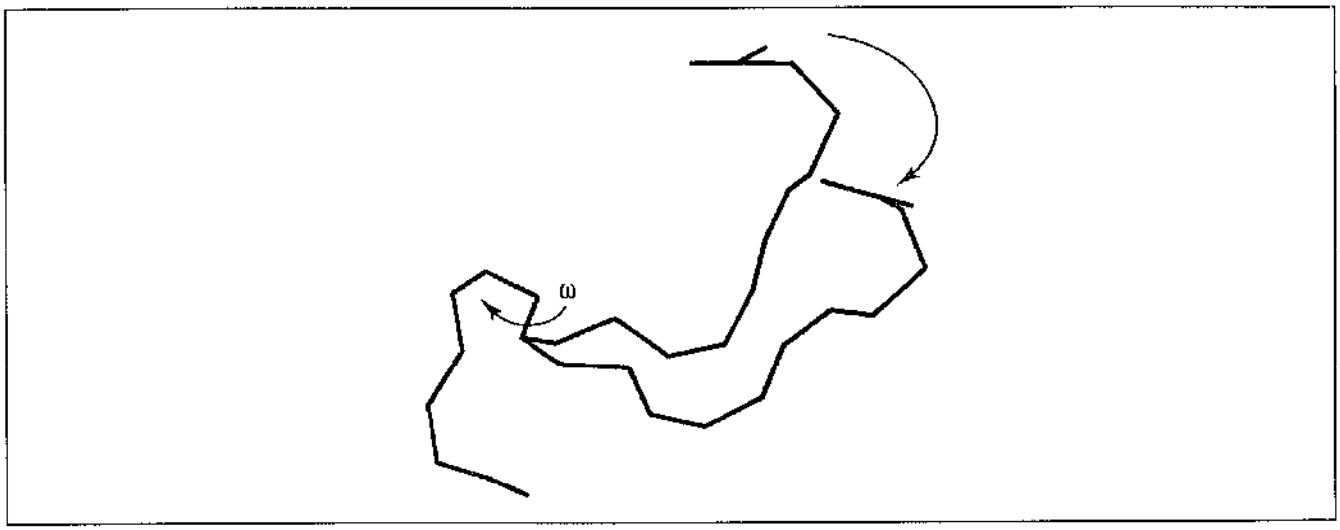


Fig 8.6 A bond rotation in the middle of a molecule may lead to a large movement at the end

8.6 Models Used in Monte Carlo Simulations of 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 modelling 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 behaviour. 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 ångstroms. Many kinds of model 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. One famous example is Flory's rotational isomeric state model [Flory 1969]. Increasing computer performance now makes it possible to use techniques such as molecular dynamics and Monte Carlo simulations to study polymer systems.

Most simulations on polymers are performed using empirical energy models (though with faster computers and new methods it is becoming possible 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 centres, 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). Our aim here is to give a flavour of the way in which Monte Carlo methods can be used to investigate polymeric systems. We divide the discussion into lattice and continuum models but recognise that there is a spectrum of models from the simplest to the most complex.

8.6.1 Lattice Models of Polymers

Lattice models have provided many insights into the behaviour 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 (Figure 8.7) 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.

More complex models have been developed in which the lattice representation is closer to the 'true' geometry of the molecule. For example, in Figure 8.8 we show the bond fluctuation model of polyethylene, in which the 'bond' between successive monomers on the lattice

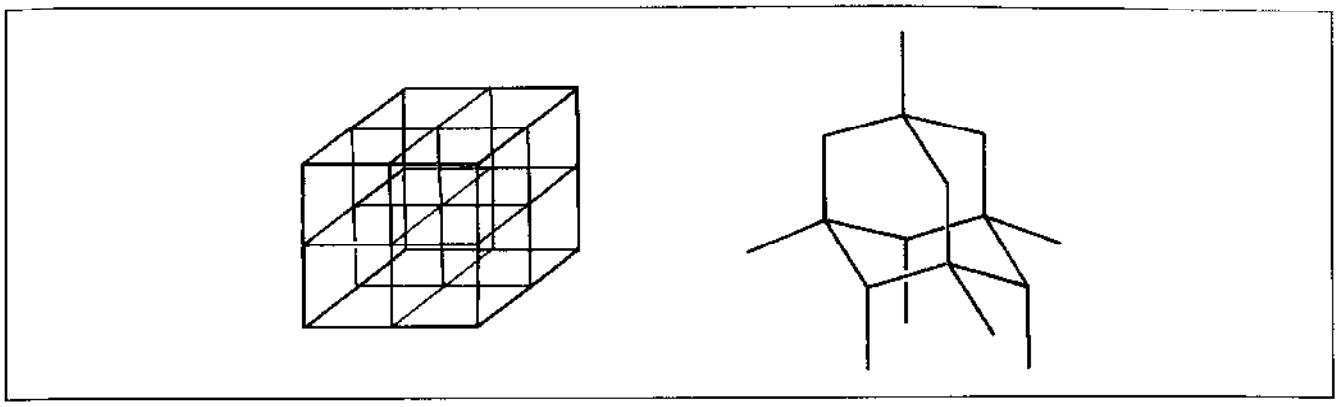


Fig 8 7 Cubic and tetrahedral (diamond) lattices, which are commonly used for lattice simulations of polymers

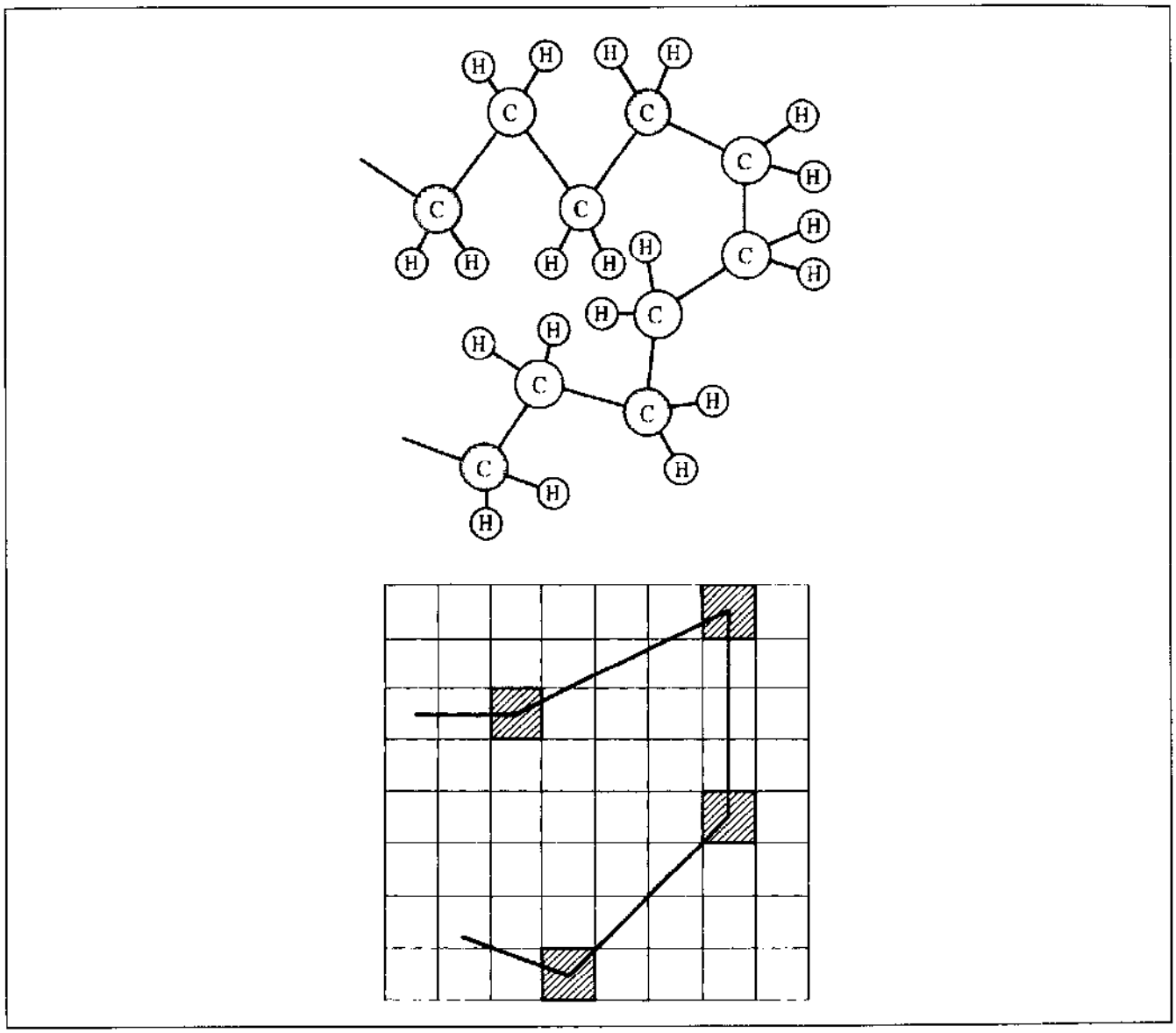


Fig 8 8· The bond fluctuation model In this example three bonds in the polymer are incorporated into a single 'effective bond' between 'effective monomers' (Figure adapted from Baschnagel J, K Binder, W Paul, M Laso, U Suter, I Batoulis, W Jilge and T Bürger 1991 On the Construction of Coarse-Grained Models for Linear Flexible Polymer-Chains – Distribution-Functions for Groups of Consecutive Monomers Journal of Chemical Physics 95 6014–6025)

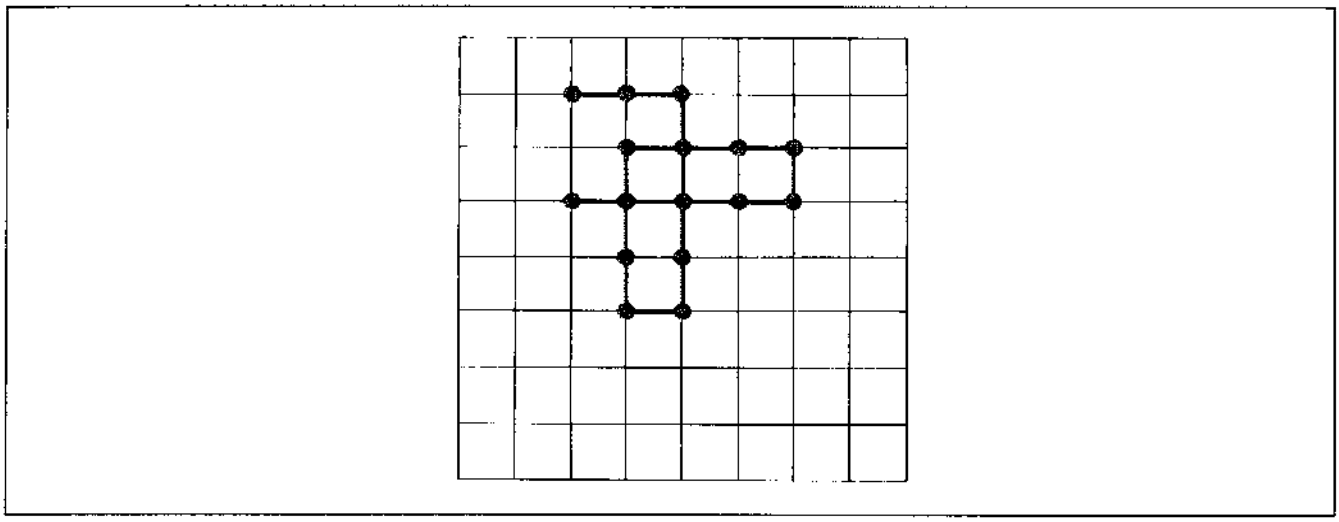


Fig 8.9· In a random walk on a square lattice the chain can cross itself.

represent three bonds in the actual molecule [Baschnagel *et al.* 1991]. In this model each monomer is positioned at the centre of a cube within the lattice and five different distances are possible for the monomer–monomer bond lengths.

Lattices can be used to study a wide variety of polymeric systems, from single polymer chains to dense mixtures. The simplest type of simulation is a 'random walk', in which the chain is randomly grown in the lattice until it contains the desired number of bonds (Figure 8.9). In this model the chain is free to cross itself (i.e. excluded volume effects are ignored). Various properties can be calculated from such simulations, by averaging the results over a large number of trials. For example, a simple measure of the size of a polymer is the mean square end-to-end distance, $\langle R_n^2 \rangle$. For the random walk model $\langle R_n^2 \rangle$ is related to the number of bonds (n) and the length of each bond (l) by:

$$\langle R_n^2 \rangle = nl^2 \tag{8.49}$$

The radius of gyration is another commonly calculated property; this is the root mean square distance of each atom (or monomer) from the centre of mass. For the random walk model the radius of gyration $\langle s^2 \rangle$ is given in the asymptotic limit by:

$$\langle s^2 \rangle = \langle R_n^2 \rangle / 6 \tag{8.50}$$

The ability of the chain to cross itself in the random walk may seem to be a serious limitation, but it is found to be valid under some circumstances. When excluded volume effects are not important (also known as 'theta' conditions) then a subscript '0' is often added to properties such as the mean square end-to-end distance, $\langle R_n^2 \rangle_0$). Excluded volume effects can be taken into account by generating a 'self-avoiding walk' of the chain in the lattice (Figure 8.10). In this model 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

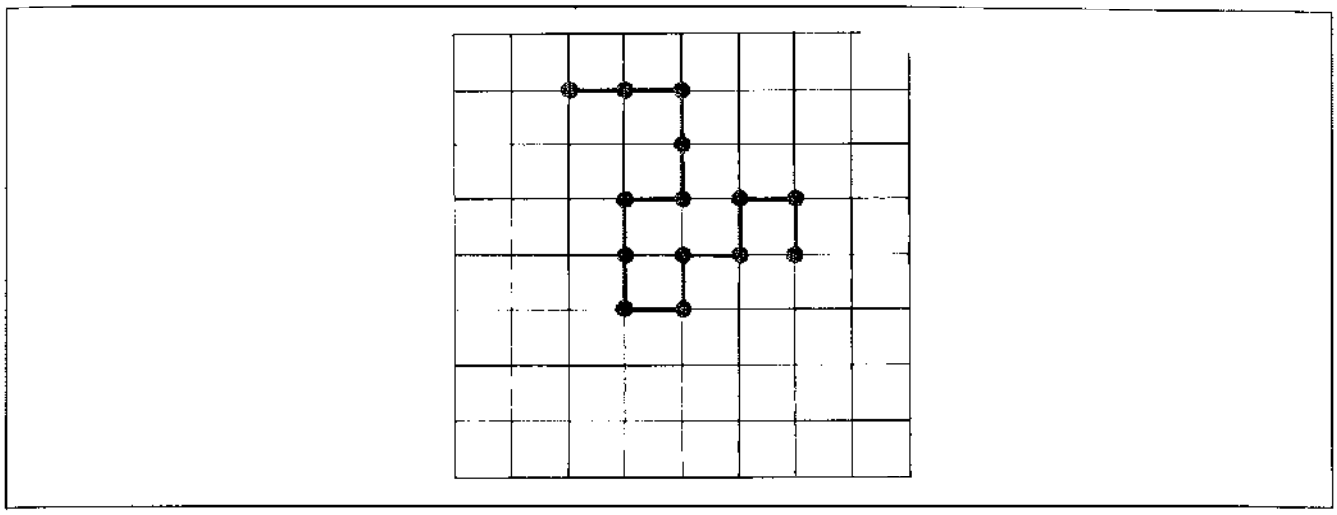


Fig 8 10· Self-avoiding walk only one monomer can occupy each lattice site

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 variants of the Verdier–Stockmayer algorithm [Verdier and Stockmayer 1962] new configurations are generated using combinations of 'crankshaft', 'kink jump' and 'end rotation' moves (Figure 8.11). Another widely used algorithm in Monte Carlo 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 8.12. 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.

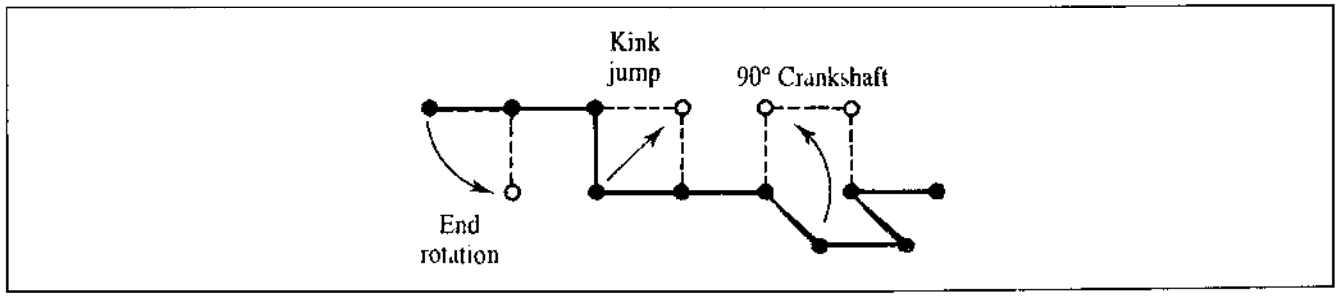


Fig 8 11· The 'crankshaft', 'kink jump' and 'end rotation' moves used in Monte Carlo simulations of polymers

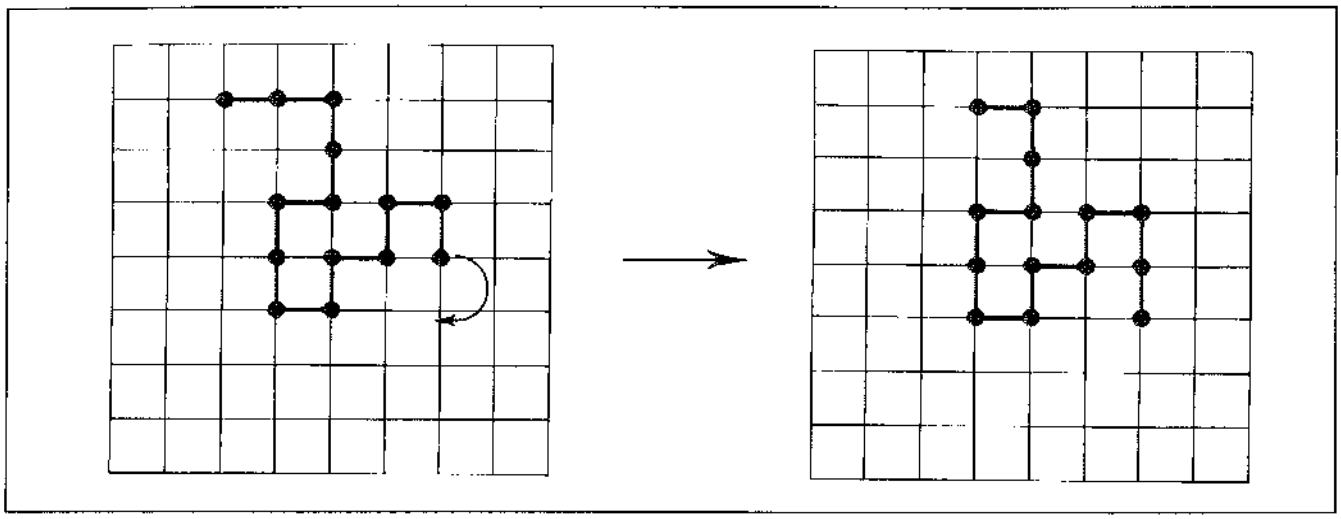


Fig 8 12· The 'slithering snake' algorithm

8.6.2 'Continuous' Polymer Models

The simplest of the continuous polymer models consists of a string of connected beads (Figure 8.13). 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 ('effective monomers' is a more appropriate term). Similarly, the links between the beads should not be thought of as bonds. The links may be modelled as rods of a fixed and invariant length or may be permitted to vary using a harmonic potential function.

In Monte Carlo 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 8.13. For isolated polymer chains the pivot algorithm can give a good sampling of the configurational/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°

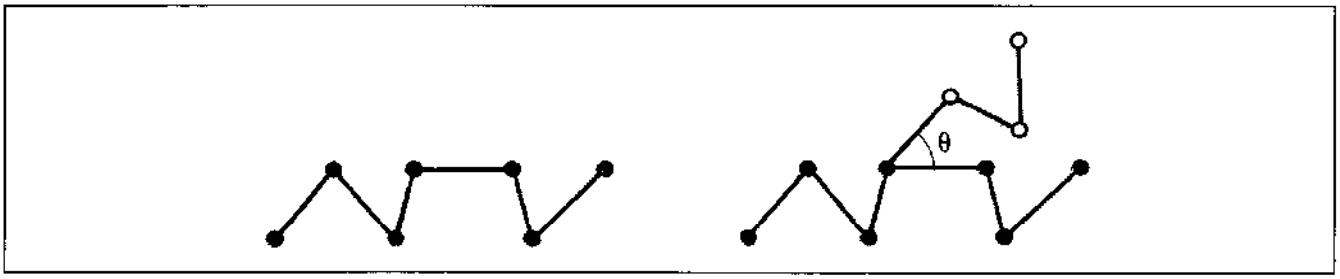


Fig 8 13 The bead model for polymer simulations The beads may be connected by stiff rods or by harmonic springs

and 360° is equally likely. Fixing the bond angles in this way obviously affects the properties of the chain when compared to the freely jointed chain; one way to quantify this is via the characteristic ratio C_n, which is defined as:

$$C_n = \frac{\langle R_n^2 \rangle_0}{nl^2} \tag{8.51}$$

The characteristic ratio approximately indicates how extended the chain is. For the freely rotating chain the characteristic ratio is given by:

$$C_n = \frac{1+\cos\theta'}{1-\cos\theta'} - \frac{2\cos\theta'}{n}\frac{1-\cos^n\theta'}{(1-\cos\theta')^2} \tag{8.52}$$

where θ' is the supplement of the normal bond angle (i.e. $\theta' = 180° - \theta$). For an infinitely long chain the characteristic ratio becomes:

$$C_\infty = \frac{1+\cos\theta'}{1-\cos\theta'} \tag{8.53}$$

To move up the scale of complexity one now needs to consider the energetics of rotation about each bond. The simplest approach is to assume that each bond can be treated independently and that the total energy of the chain is the sum of the individual torsional energies for each bond. However, this particular model has some serious shortcomings arising from the assumption of independence.

The *rotational isomeric state model* (RIS) developed by Flory [Flory 1969] is probably the best known of the 'approximate' approaches to modelling polymer chains. Each bond is assumed to adopt one of a small number of discrete rotational states, which usually correspond to minima in the potential energy. For example, one might use three rotational states for a typical polyalkane, corresponding to the trans, gauche(+) and gauche(−) conformations. A key part of the RIS approach is its elegant use of various matrices to simplify the calculation. *Generator matrices* are used to establish certain conformation-dependent properties. Thus for a property A one would write:

$$A(\tau_1 \ldots \tau_n) = \prod_{i=1}^{n} \mathbf{F}_i \tag{8.54}$$

where $\mathbf{F}_i$ is the generator matrix for the particular property for bond i (with torsion angle τ_i). An example is the generator matrix for the square end-to-end distance R^2, which takes the following form:

$$\mathbf{G}_i = \begin{bmatrix} 1 & 2\mathbf{l}^{\mathrm{T}}\mathbf{T} & l^2 \\ 0 & \mathbf{T} & \mathbf{1} \\ 0 & 0 & 1 \end{bmatrix} \tag{8.55}$$

The vector $\mathbf{l}$ is the bond vector for bond i and $\mathbf{T}$ is the 3×3 matrix that transforms coordinates in the reference frame for bond $(i+1)$ to those in the frame of bond i. In this case the square end-to-end distance can be calculated from:

$$R^2 = \mathbf{G}_{[1}\mathbf{G}_2^{n-2}\mathbf{G}_{n]} \tag{8.56}$$

The nomenclature is such that $\mathbf{G}_{[1}$ represents the first row of the matrix $\mathbf{G}_1$ and $\mathbf{G}_{n]}$ represents the last column of G_n.

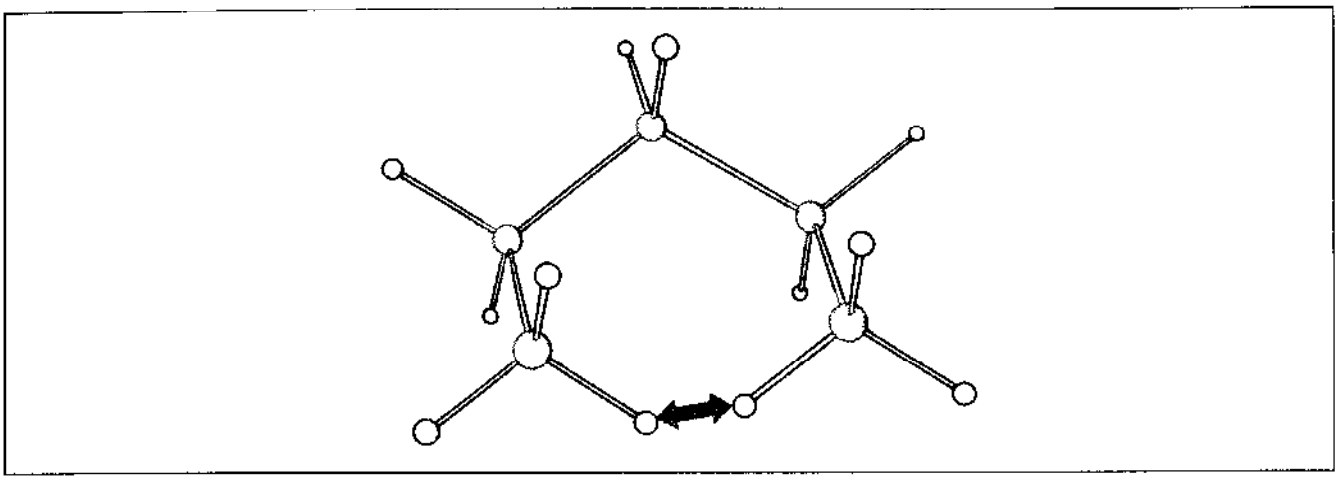


Fig 8 14. Pentane violation

In order to calculate average properties of the polymer chain one uses a standard statistical mechanical approach involving a summation over all possible conformations, with each term multiplied by the appropriate Boltzmann factor. This involves the use of a *statistical weights matrix,* which Flory introduced to deal with the influence a bond has on its neighbours. The pentane violation is the most important of these, wherein a sequence of gauche(+) and gauche(−) bonds gives rise to an unfavourable high-energy interaction (Figure 8.14). The statistical weights matrix associated with bond i has v_{i-1} rows and v_i columns, which correspond to the v_{i-1} rotational states of bond $(i-1)$ and the v_i rotational states of bond i. For example, for a typical polymer with the trans, gauche(+) and gauche(−) rotational states the statistical weights matrix is:

$$\mathbf{U}_i = \begin{bmatrix} u_{tt} & u_{tg^+} & u_{tg^-} \\ u_{g^+t} & u_{g^+g^+} & u_{g^+g^-} \\ u_{g^-t} & u_{g^-g^+} & u_{g^-g^-} \end{bmatrix} \tag{8.57}$$

Some typical values for the elements of this matrix would be:

$$\mathbf{U}_i = \begin{bmatrix} 1.0 & 0.54 & 0.54 \\ 1.0 & 0.54 & 0.05 \\ 1.0 & 0.05 & 0.54 \end{bmatrix} \tag{8.58}$$

The key point to note is the small weight (0.05) for adjacent gauche(+)–gauche(−) bonds. By combining the generator and statistical weights matrices it is possible to reduce the problem of calculating the average value of a property from a set of complex integrals to a series of straightforward matrix multiplications. Some of the properties that can be determined from the RIS model include the mean square end-to-end distance, the mean square radius of gyration and the mean square dipole moment.

The RIS model can be combined with the Monte Carlo simulation approach to calculate a wider range of properties than is available from the simple matrix multiplication method. In the RIS Monte Carlo method the statistical weight matrices are used to generate chain conformations with a probability distribution that is implied in their statistical weights.

Each conformation is generated by starting at one end of the chain and setting the backbone torsion angles one at a time until the entire chain has been constructed. The probability that a particular torsional state is selected for a given bond depends upon the *a priori* probabilities of each state and also upon the torsional state selected for the previous bond in the chain. These probabilities are used at each step by the Monte Carlo procedure to generate the whole chain. A large number of such chains is grown, calculating for each the properties of interest, which are then averaged. Properties which can be determined by the RIS-MC approach include the pair correlation function (which gives the relative probability of finding two atoms within the same chain separated by a distance r), the scattering function (which indicates how the polymer may scatter neutrons or X-rays) and the force-elongation relation (which gives the mean end-to-end distance of a chain subjected to an external force).

The ultimate level of detail in polymer modelling is achieved with the atomistic models, which as the name implies explicitly represent the atoms in the system. An atomistic model is clearly the closest to 'reality' and is necessary if one wishes to calculate accurately certain properties. One of the major problems with simulations of polymers that is particularly pertinent to the atomistic models is how to generate an initial configuration of the system. Amorphous polymers by definition do not adopt a characteristic and reproducible three-dimensional structure. It is important that the properties of the initial configuration are similar to the state one wishes to simulate else the computer time needed to move to the required state can be prohibitive. For short chains containing approximately 20–30 backbone bonds it is feasible to start from a regular crystalline structure, which can then be melted, but to 'melt' a long chain may require a prohibitive amount of computer time. For longer chains an initial configuration may be generated using a random walk and periodic boundary conditions (Figure 8.15). Such an arrangement will inevitably contain high-energy overlaps. These unfavourable interactions may be removed by relaxing the system using minimisation and/or computer simulation, during which the force field potentials are gradually turned on.

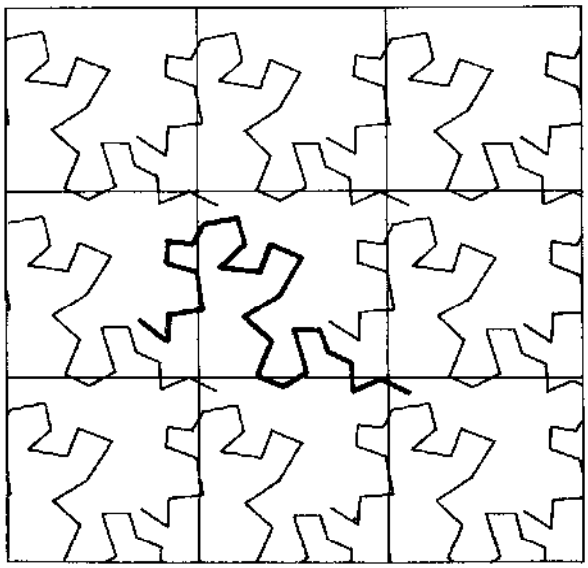


Fig 8 15 Generation of an initial configuration of a polymer using periodic boundary conditions

8.7 'Biased' Monte Carlo Methods

In some situations one is particularly interested in the behaviour of just a part of the system. For example, if we were simulating a solute–solvent system that contained a single solute molecule surrounded by a large number of solvent molecules then the behaviour of the solute and its interactions with the solvent would be of most interest. Solvent molecules far from the solute would be expected to behave almost like bulk solvent. A variety of techniques have been developed which can enhance the ability of the Monte Carlo method to explore the most important regions of phase space in such cases. One relatively simple procedure is *preferential sampling*, where the molecules in the vicinity of the solute are moved more frequently than those further away. This can be implemented by defining a cutoff region around the solute; molecules outside the cutoff region are moved less frequently than those inside the region as determined by a probability parameter p. At each Monte Carlo iteration a molecule is randomly chosen. If the molecule is inside the cutoff region then it is moved; if it is outside the region then a random number is generated between 0 and 1 and compared to the probability p. If p is greater than the random number a trial move is attempted; otherwise no move is made, no averages are accumulated, and a new molecule is randomly selected. The closer p is to zero, the more often 'closer' molecules are moved than 'further' molecules.

An alternative to the use of a fixed cutoff region is to relate the probability of choosing a solvent molecule to its distance from the solute, usually by some inverse power of the distance

$$p \propto r^{-n} \tag{8.59}$$

In preferential sampling it is necessary to ensure that the correct procedures are followed when deciding whether to accept or reject a move in a manner that is consistent with the principle of microscopic reversibility. Suppose a molecule inside the cutoff region is moved outside the cutoff region. In the preferential sampling scheme the chances of the molecule now being selected for an out → in move are less than for the original in → out move and this must be taken into account when determining the acceptance criteria

The *force-bias* Monte Carlo method [Pangali *et al.* 1978; Rao and Berne 1979] biases the movement according to the direction of the forces on it. Having chosen an atom or a molecule to move, the force on it is calculated. The force corresponds to the direction in which a 'real' atom or molecule would move. In the force-bias Monte Carlo method the random displacement is chosen from a probability distribution function that peaks in the direction of this force. The *smart Monte Carlo method* [Rossky *et al.* 1978] also requires the forces on the moving atom to be calculated. The displacement of an atom or molecule in this method has two components; one component is the force, and the other is a random vector $\delta\mathbf{r}^G$:

$$\delta\mathbf{r}_i = \frac{A\mathbf{f}_i}{k_B T} + \delta\mathbf{r}_i^G \tag{8.60}$$

where $\mathbf{f}_i$ is the force on the atom and A is a parameter The random displacement $\delta\mathbf{r}_i^G$ is chosen from a normal distribution with zero mean and variance equal to $2A$.

The main difference between the force-bias and the smart Monte Carlo methods is that the latter does not impose any limit on the displacement that an atom may undergo. The displacement in the force-bias method is limited to a cube of the appropriate size centred on the atom. However, in practice the two methods are very similar and there is often little to choose between them. In suitable cases they can be much more efficient at covering phase space and are better able to avoid bottlenecks in phase space than the conventional Metropolis Monte Carlo algorithm. The methods significantly enhance the acceptance rate of trial moves, thereby enabling larger moves to be made as well as simultaneous moves of more than one particle. However, the need to calculate the forces makes the methods much more elaborate, and comparable in complexity to molecular dynamics.

8.8 Tackling the Problem of Quasi-ergodicity: J-walking and Multicanonical Monte Carlo

If there are high-energy barriers between the potential energy minima in a system then a normal Metropolis Monte Carlo simulation may become trapped in just a few of the low-energy regions and fail to properly sample large regions of the thermally accessible space. Such a simulation may appear to possess all the qualities of a good simulation, in terms of its convergence, yet may give results that are completely incorrect. Such a simulation is often referred to as *quasi-ergodic*. This problem arises when studying systems as diverse as rare gas clusters near their melting temperature or protein folding, but it can also be demonstrated in even the simplest of model systems, such as the one-dimensional double-well potential (Figure 8.16). At low temperatures the simulation is unable to cross the high-energy barriers because of the favourable Boltzmann factor. A variety of methods have been suggested for tackling this problem, of which we shall consider two: J-walking and the multicanonical Monte Carlo method.

8.8.1 J-walking

In the J-walking (or Jump-walking) method [Frantz *et al.* 1990] a low-temperature Monte Carlo simulation is permitted occasionally to attempt jumps to regions of space that are accessible to a simulation run at a high temperature. The simplest way to implement this method is to perform the two simulations (at the high and low temperatures) in tandem. The low-temperature simulation is periodically permitted to attempt a jump to the configuration of the high-temperature simulation (the J-walker). The same Metropolis criteria are applied when deciding whether or not to accept the move. The high-temperature simulation will still in principle tend to be biased towards the low-energy regions so there will still be a reasonable chance that one of these special attempted jumps will be accepted.

In practice, it is found that this simple implementation is not the most effective approach. There are two particular problems First, when the two simulations are run in tandem then significant correlations can arise, which results in large systematic errors. There are a number of ways to avoid these correlations, such as moving the J-walker an extra number

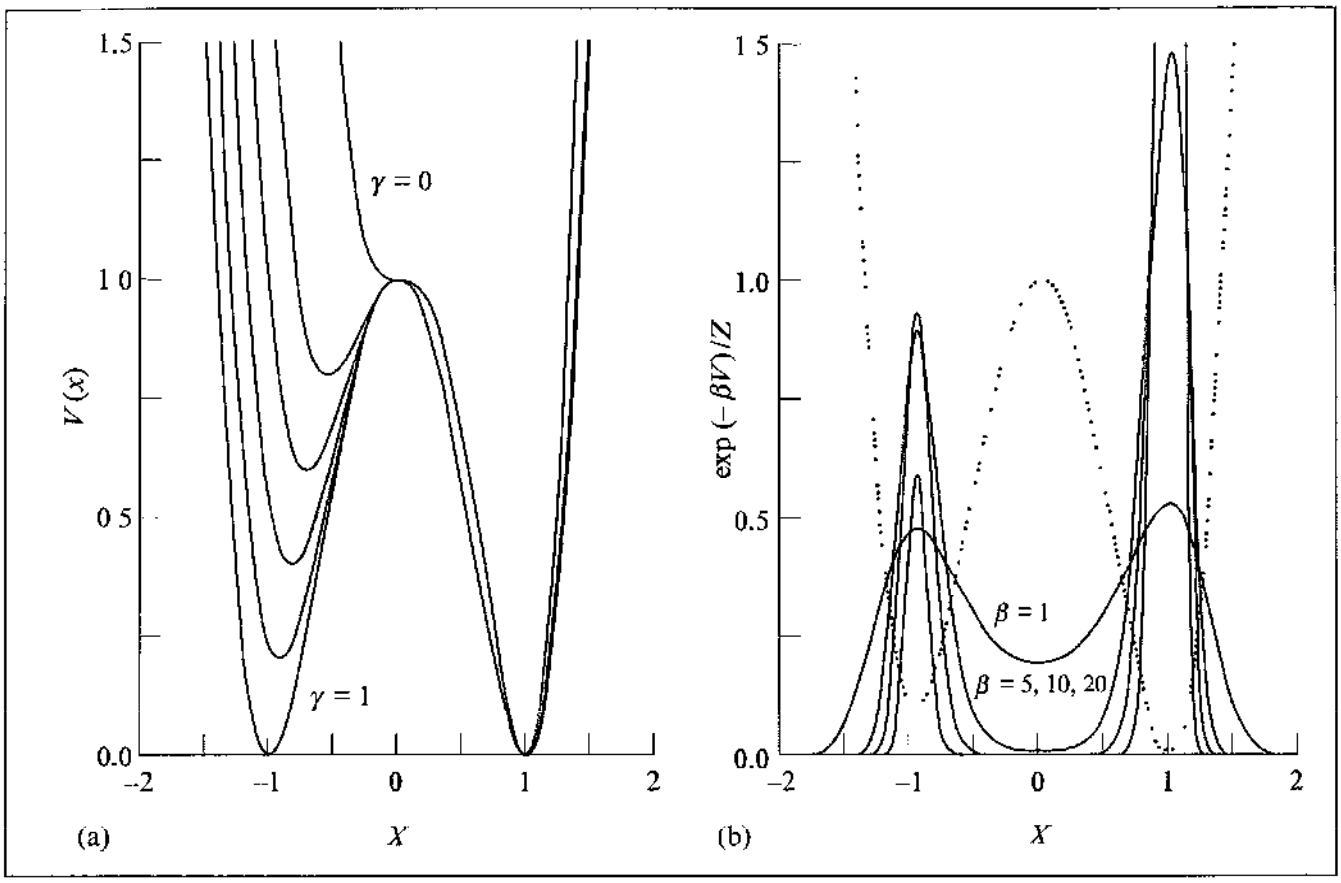


Fig 8 16 Double-well potential which shows quasi-ergodicity The potential is described by the quartic $\mathcal{V}(x) = 3\delta x^4 + 4\delta(\alpha - 1)x^3 - 6\delta\alpha x^2 + 1$, *where* $\delta = 1/(2\alpha + 1)$, *and the potential is characterised by a parameter* γ $(= (\alpha^4 + 2\alpha^3)/(2\alpha + 1))$ *When* $\gamma = 0$ *there is just a single well and when* $\gamma = 1$ *the potential is symmetrical On the right are Boltzmann distribution functions for the* $\gamma = 0\ 9$ *potential for various temperatures, expressed in terms of* β $(= 1/k_B T)$ *(Figure redrawn from Frantz, D D, D L Freeman and J D Doll 1990 Reducing quasi-ergodic behavior in Monte Carlo simulations by J-walking applications to atomic clusters* Journal of Chemical Physics **93** *2769–2784)*

of steps whenever a jump is attempted or running several high-temperature J-walkers and selecting jumps from them at random. However, it is found that a more effective approach is to perform the high-temperature simulation first The low-temperature simulation first reads in and stores the configurations from the high-temperature simulation, which are then selected at random for each special jump. This approach obviously requires that sufficient storage is available to save the high-temperature configurations; however, it is not necessary to store every configuration (as there will be a high degree of correlation between successive configurations) but rather a representative sample.

One of the first applications of J-walking was to argon clusters [Frantz *et al.* 1990]. These clusters (containing up to 30 atoms) are experimentally known to exhibit strange melting behaviour, wherein the melting temperature is very heavily dependent upon the number of atoms in the cluster. Thus argon clusters Ar_n with seven, thirteen or nineteen atoms have particularly high melting temperatures, whereas those clusters with eight, fourteen or twenty atoms have particularly low melting temperatures Moreover, certain clusters have different melting and freezing temperatures, implying that there is a range of

temperatures where both solid-like and liquid-like forms coexist Simulation of these clusters in this transition region is difficult because of the problems of quasi-ergodicity and in obtaining satisfactory convergence. When applying the J-walking method to these clusters it was found necessary to generate the J-walker distribution in stages. The objective was to study an Ar_{13} cluster over a temperature range of 24–41 K using a J-walker at 50 K. However, the distribution of potential energies for the 50 K J-walker did not overlap with the distribution for a 20 K walker, which would mean that at 20 K hardly any of the attempted jumps to the J-walker distribution would be accepted. A series of simulations was thus performed. First, a 50 K J-walker was used to generate a distribution at 40 K, which in turn was used to produce a distribution at 30 K. Finally, the 30 K distribution acted as the J-walker for the 20 K simulation. Significantly better convergence for the cluster configurational energy and the heat capacity when compared with an equivalent standard Metropolis Monte Carlo procedure were noted, even when the simulations were started from a random configuration rather than the icosohedral low-energy geometry.

A related approach designed to be used for conformationally flexible molecules is the 'jumping between wells' (JBW) approach [Senderowitz *et al.* 1995]. Here a conformational analysis (see Chapter 9) is first performed on the molecule in order to identify its thermally accessible minimum-energy conformations (e g. all within 5 kcal/mol of the global minimum-energy conformation). These minimum-energy conformations are stored in a list. The changes in internal coordinates that would be required to interconvert each pair of minimum-energy conformations are then determined. At each stage of the iterative cycle the minimum-energy conformation that is closest to the current structure is identified. A minimum-energy conformation is then selected at random from the conformation list and the appropriate transformation applied to the current structure. Small random changes in this new structure are then made to give a new trial structure, which is then accepted or rejected using the Metropolis criterion with reference to the initial starting structure. The process is then repeated. It is important with this method to avoid oversampling some of the potential energy wells which can occur if the combination of a conformational jump and the subsequent randomisation enters the space of a different minimum-energy conformation. This problem can be overcome by only including minimum-energy conformations that are significantly different from each other and by making the randomisation step small relative to the distance between the conformations.

8.8.2 The Multicanonical Monte Carlo Method

In a canonical ensemble the probability $P_{\text{canon}}(T, E)$ of visiting a point in phase space with an energy E is proportional to the Boltzmann factor, $w_B = \exp(-E/k_B T)$, multiplied by the density of states, $n(E)$, where the number of states between E and $E + dE$ is given by $n(E)\delta E$. Thus:

$$P_{\text{canon}}(T, E) \propto n(E) w_B(E) \tag{8 61}$$

The density of states increases rapidly with energy but the Boltzmann factor decreases exponentially, meaning that $P_{\text{canon}}(T, E)$ is bell-shaped, with values that can vary by many orders of magnitude as the energy changes. In the multicanonical method the simulation

is performed in an artificial multicanonical ensemble in which the probability of visiting a state is independent of its energy over a certain energy range. This is equivalent to replacing the Boltzmann factor by a multicanonical weight factor, $w_{mu}(E)$:

$$P_{mu}(E) \propto n(E)w_{mu}(E) = \text{constant} \tag{8.62}$$

This implies that the multicanonical weight factor is proportional to $n(E)^{-1}$. A simulation performed in the multicanonical ensemble is able to overcome any energy barrier, in contrast to the situation in a normal, canonical simulation. The main task in performing a multicanonical simulation is to determine the weight factor, which is not known *a priori* (unlike the case for the canonical ensemble, where it is equal to the Boltzmann factor). The multicanonical weight factor is usually determined in an iterative fashion from a series of short simulations. One approach is as follows [Okamoto and Hansmann 1995]. First, a simulation is carried out at a temperature (T_0) which is high enough (e.g. 1000 K) to ensure that all energy barriers can be overcome. An array $S(E)$ is established with all its elements initially set to zero. Each element of this array refers to a particular energy range E to $E + \delta E$, where δE might be (say) 1 kcal/mol. From this initial simulation a histogram is constructed which gives the number of times a state with an energy in the range E to $E + \delta E$ is determined. These histogram values are stored in an array $H(E)$. Each of the values in this array ($H(E)$) should thus initially approximate the energy distribution at the temperature T_0:

$$H(E) \propto n(E)\exp(-E/k_B T_0) \tag{8.63}$$

During this simulation the minimum and maximum energies visited during the simulation are recorded (E_{min} and E_{max}, respectively) The array $S(E)$ is now updated according to the following formula:

$$S(E) = S(E) + \ln H(E) \tag{8.64}$$

for all energy values E where the energy is between E_{min} and E_{max} and where the value of the appropriate array element $H(E)$ is greater than some minimum value (e.g. 20). In other words, that particular energy level must have been visited at least twenty times during the simulation. The following two parameters are now calculated:

$$\beta(E) = \begin{cases} 1/k_B T_0 & E \geq E_{max} \\ 1/k_B T_0 + \dfrac{S(E') - S(E)}{E' - E} & E_{min} \leq E \leq E_{max} \\ \beta(E_{min}) & E < E_{min} \end{cases} \tag{8.65}$$

$$\alpha(E) = \begin{cases} 0 & E \geq E_{max} \\ \alpha(E') + (\beta(E') - \beta(E))E' & E < E_{max} \end{cases} \tag{8.66}$$

E' refers to that element in the array $S(E)$ which succeeds E (i.e. E and E' are adjacent elements). Having determined the parameters $\alpha(E)$ and $\beta(E)$, a multicanonical weight factor is calculated as:

$$w_{mu}(E) = \exp[-\beta(E)E - \alpha(E)] \tag{8.67}$$

A new simulation is now initiated using this multicanonical weight factor (rather than the Boltzmann factor), from which new values of $S(E)$ and thus $H(E)$ can be determined. This cycle is continued until the distribution in $H(E)$ is reasonably flat in the energy range being considered.

Once the final multicanonical weight factor has been derived it provides the distribution for the production simulation in which high-energy configurations will be sampled adequately and high-energy barriers can be crossed with ease. Moreover, from this single simulation it is possible to derive the canonical distribution $P_{\text{canon}}(T, E)$ at any temperature (hence the name 'multicanonical'):

$$P_{\text{canon}}(T, E) \propto P_{\text{mu}}(E) w_{\text{mu}}^{-1} e^{-E/k_B T} \tag{8.68}$$

The average value of any property A at a temperature T can be determined from the multicanonical simulation using the following formula:

$$\langle A \rangle_T = \frac{\int A(E) P_{\text{canon}}(T, E)\, dE}{\int P_{\text{canon}}(T, E)\, dE} \tag{8.69}$$

In practice, one is restricted to energies between E_{min} and E_{max}, and a range of temperatures $T_{\text{min}} \leq T \leq T_{\text{max}}$ where the range of permitted temperatures is determined by calculating the expectation value of the energy at temperature T:

$$E_{\text{min}} \leq \langle E \rangle_T \leq E_{\text{max}} \tag{8.70}$$

The array $S(E)$ is intimately related to the entropy of the system as can be demonstrated by expanding the logarithm of $H(E)$ as:

$$\ln H(E) = \ln n(E) - E/k_B T_0 + \text{constant} \tag{8.71}$$

and recalling that the entropy is related to the density of states by $S(E) = \ln n(E)$.

The multicanonical Monte Carlo method can be used to study a wide range of systems and is particularly useful where traditional Metropolis Monte Carlo methods encounter difficulties. In addition to the simulation of systems such as clusters of rare gas atoms, the multicanonical method has been used to study the properties of macromolecular systems. There has been particular interest in using the approach to study the properties of amino acid polymers, and in particular their ability to form certain types of regular structure such as alpha helices. These structures will be discussed in more detail in Chapter 10. It suffices for now to note that certain types of amino acid confer a greater propensity to adopt a helical structure. Traditional simulation techniques can be used to study the equilibrium between the helical and 'random' (or coil) structures but typically have to start from the regular (i.e. helical) structure, which is then 'unfolded' using molecular dynamics or Monte Carlo simulations. The large number of minima on the potential energy surface means that it is not practical to start from the unfolded structure and observe helix formation. However, the multicanonical Monte Carlo method does provide a mechanism for more completely exploring the energy surface (including the helical conformations), starting from a random structure. In one such study, Okamoto and Hansmann were able to compare the thermodynamics of the equilibrium between the helix and coil structures for three amino acids (alanine, valine and glycine) which have different observed propensities to form helix structures [Okamoto and Hansmann 1995].

One of the drawbacks of the multicanonical method is that, during the simulations to derive the weight factor, the energy distribution in $H(E)$ can oscillate rather than steadily approaching a limiting distribution. Another drawback is that it can fail to properly

sample the low-energy regions adequately. The multicanonical method samples energies within a certain range with approximately equal probability, but at the ends of this range the probability drops dramatically. Thus little sampling is done from the low-energy regions These low-energy regions are proportionally more important at low temperatures, leading to poor statistics. To some extent the problems with J-walking are complementary to the limitations of the multicanonical method, and so attempts have been made to combine the two [Xu and Berne 1999]. In the combined method (termed 'multicanonical jump walking') the multicanonical weight factor is first derived and then used to perform a long multicanonical simulation. The configurations generated by this multicanonical simulation are saved and used as the 'high-temperature' component in the subsequent J-walking simulation. The key modification is that the standard acceptance criterion for the jump steps during this last phase must be multiplied by the ratio of the weight factors for the two energies (i.e. rather than comparing the usual Boltzmann factor, $\exp(-\Delta\mathscr{V}/k_BT)$, to the random number a modified factor $\exp(-\Delta\mathscr{V}/k_BT)[w(\mathscr{V}_{old})/w(\mathscr{V}_{new})]$ is used). This combined approach appears to provide more efficient sampling of phase space for a given number of Monte Carlo steps when compared to the regular J-walking or multicanonical sampling method, on both low-dimensional trial potentials and for systems such as rare gas clusters.

8.9 Monte Carlo Sampling from Different Ensembles

A Monte Carlo simulation traditionally samples from the constant *NVT* (canonical) ensemble, but the technique can also be used to sample from different ensembles. A common alternative is the isothermal–isobaric, or constant *NPT*, ensemble. To simulate from this ensemble, it is necessary to have a scheme for changing the volume of the simulation cell in order to keep the pressure constant. This is done by combining random displacements of the particles with random changes in the volume of the simulation cell. The size of each volume change is governed by the maximum volume change, δV_{max}. Thus a new volume is generated from the old volume as follows:

$$V_{new} = V_{old} + \delta V_{max}(2\xi - 1) \tag{8.72}$$

As usual, ξ is a random number between 0 and 1. When the volume is changed, it is in principle necessary to recalculate the interaction energy of the entire system, not just the interactions involving the one atom or molecule that has been displaced. However, for simple interatomic potentials the change in energy associated with a volume change can be calculated very rapidly by using *scaled coordinates*. For a set of particles that are modelled by a Lennard-Jones potential in a cubic box of length L_{old}, the potential energy can be written:

$$\mathscr{V}_{old}(\mathbf{r}^N) = 4\varepsilon \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(\frac{\sigma}{L_{old}s_{ij}}\right)^{12} - 4\varepsilon \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(\frac{\sigma}{L_{old}s_{ij}}\right)^{6} \tag{8.73}$$

where s_{ij} is a scaled coordinate which is related to the actual interatomic distance by $s_{ij} = L_{old}^{-1} r_{ij}$. It is necessary to write the energy as the sum of two components, one from

the repulsive part of the Lennard-Jones potential and the other from the attractive part:

$$\mathscr{V}_{\text{old}}(\mathbf{r}^N) = \mathscr{V}_{\text{old}}(12) + \mathscr{V}_{\text{old}}(6) \tag{8.74}$$

The advantage of using scaled coordinates is that they are independent of the size of the simulation box. Thus the energy of the configuration in a different-sized box (with side L_{new}) is:

$$\mathscr{V}_{\text{new}}(\mathbf{r}^N) = 4\varepsilon \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(\frac{\sigma}{L_{\text{new}} s_{ij}}\right)^{12} - 4\varepsilon \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(\frac{\sigma}{L_{\text{new}} s_{ij}}\right)^{6} \tag{8.75}$$

The energy $\mathscr{V}_{\text{new}}(\mathbf{r}^N)$ is related to the energy $\mathscr{V}_{\text{old}}(\mathbf{r}^N)$ as follows:

$$\mathscr{V}_{\text{new}}(\mathbf{r}^N) = \mathscr{V}_{\text{old}}(12)\left\{\frac{L_{\text{old}}}{L_{\text{new}}}\right\}^{12} + \mathscr{V}_{\text{old}}(6)\left\{\frac{L_{\text{old}}}{L_{\text{new}}}\right\}^{6} \tag{8.76}$$

The change in energy from the old to the new system is thus:

$$\Delta\mathscr{V}(\mathbf{r}^N) = \mathscr{V}_{\text{old}}(12)\left\{\frac{L_{\text{old}}}{L_{\text{new}}} - 1\right\}^{12} + \mathscr{V}_{\text{old}}(6)\left\{\frac{L_{\text{old}}}{L_{\text{new}}} - 1\right\}^{6} \tag{8.77}$$

Any long-range corrections to the potential must also be taken into account when the volume changes. One way to deal with these is to assume that the non-bonded cutoff scales with the box length. Under such circumstances, the long-range corrections to both the repulsive and attractive parts of the potential scale in exactly the same manner as the short-range interactions. However, the use of this assumption can give rise to serious problems, particularly for techniques such as the Gibbs ensemble Monte Carlo simulation (see Section 8.12) where two coupled simulation boxes of different dimensions are involved. The boxes contain identical particles, but this would be compromised by the use of different non-bonded cutoffs and long-range corrections.

This simple scaling method cannot be used when simulating molecules, for a change in the scaled coordinates would have the effect of introducing large and energetically unfavourable changes in the internal coordinates, such as the bond lengths. It is therefore necessary to recalculate the total interaction energy of the system each time a volume change is made. This is computationally expensive to do, but it is in any case advisable to change the volume relatively infrequently compared to the rate at which the particles are moved. One way to speed up the energy calculation associated with a volume change is to write the potential energy change as a Taylor series expansion of the box size.

The criterion used to accept or reject a new configuration is slightly different for the isothermal–isobaric simulation than for a simulation in the canonical ensemble. The following quantity is used:

$$\Delta H(\mathbf{r}^N) = \mathscr{V}_{\text{new}}(\mathbf{r}^N) - \mathscr{V}_{\text{old}}(\mathbf{r}^N) + P(V_{\text{new}} - V_{\text{old}}) - Nk_{\text{B}}T \ln\left(\frac{V_{\text{new}}}{V_{\text{old}}}\right) \tag{8.78}$$

If ΔH is negative then the move is accepted; otherwise, $\exp(-\Delta H/k_{\text{B}}T)$ is compared to a random number between 0 and 1 and the move accepted according to:

$$\text{rand}(0,1) \leq \exp(-\Delta H/k_{\text{B}}T) \tag{8.79}$$

To check that an isothermal–isobaric simulation is working properly, the pressure can be calculated from the virial as outlined in Section 6.2.3, including the appropriate long-range correction (which will not, of course, be constant as the volume of the box changes). Its value should be equal to the input pressure that appears in Equation (8.78).

8.9.1 Grand Canonical Monte Carlo Simulations

In the grand canonical ensemble the conserved properties are the chemical potential, the volume and the temperature. It can sometimes be more convenient to perform a grand canonical simulation at constant activity, z, which is related to the chemical potential μ by.

$$\mu = k_B T \ln \Lambda^3 z \tag{8.80}$$

where Λ is the de Broglie wavelength given by $\Lambda = \sqrt{h^2/2\pi m k_B T}$.

The key feature about the grand canonical Monte Carlo method is that the number of particles may change during the simulation. There are three basic moves in a grand canonical Monte Carlo simulation:

1. A particle is displaced, using the usual Metropolis method.
2. A particle is destroyed.
3. A particle is created at a random position.

The probability of creating a particle should be equal to the probability of destroying a particle To determine whether to accept a destruction move the following quantity is calculated:

$$\Delta D = \frac{[\mathscr{V}_{\text{new}}(\mathbf{r}^N) - \mathscr{V}_{\text{old}}(\mathbf{r}^N)]}{k_B T} - \ln\left(\frac{N}{zV}\right) \tag{8.81}$$

For a creation step the equivalent quantity is:

$$\Delta C = \frac{[\mathscr{V}_{\text{new}}(\mathbf{r}^N) - \mathscr{V}_{\text{old}}(\mathbf{r}^N)]}{k_B T} - \ln\left(\frac{zV}{N+1}\right) \tag{8.82}$$

If ΔD or ΔC is negative then the move is accepted; if positive, then the exponential $\exp(-\Delta D/k_B T)$ or $\exp(-\Delta C/k_B T)$ as appropriate is calculated and compared with a random number between 0 and 1 in the usual way.

It is important that the possibility of creating a new particle is the same as the probability of destroying an old one. The ratio of particle creation/destruction moves to translation/rotation moves can vary, but the most rapid convergence is often achieved if all types of move occur with approximately equal frequency.

In grand canonical Monte Carlo simulations of liquids there can be some practical problems in achieving statistically accurate results. This is because the probability of achieving a successful creation or destruction step is often very small. Creation steps fail because the fluid is so dense that it is difficult to insert a new particle without causing significant

overlaps with neighbouring particles. Destruction steps fail because the particles in a fluid often experience significant attractive interactions, which are lost when the particle is removed. These problems are particularly acute for long-chain molecules. However, some of the newer Monte Carlo techniques such as the configurational bias Monte Carlo method do enable such systems to be simulated and accurate results obtained. These techniques will be discussed in Section 8.11.

8.9.2 Grand Canonical Monte Carlo Simulations of Adsorption Processes

One application of the grand canonical Monte Carlo simulation method is in the study of the adsorption and transport of fluids through porous solids. Mixtures of gases or liquids can be separated by the selective adsorption of one component in an appropriate porous material. The efficacy of the separation depends to a large extent upon the ability of the material to adsorb one component in the mixture much more strongly than the other component. The separation may be performed over a range of temperatures and so it is useful to be able to predict the adsorption isotherms of the mixtures.

A typical example of such a calculation is the simulation of Cracknell, Nicholson and Quirke [Cracknell *et al.* 1994] who studied the adsorption of a mixture of methane and ethane onto a microporous graphite surface. Four types of move were employed in their simulations: particle moves, particle deletions, particle creations and attempts to exchange particles. Methane was modelled as a single Lennard-Jones particle and ethane as two Lennard-Jones particles separated by a fixed bond length. The graphite surfaces were modelled as Lennard-Jones atoms with a slit-shaped pore being constructed from two layers of graphite separated by an appropriate distance. Triangle-shaped pores can also be used. The simulations were used to calculate the selectivity of the solid for the two components as the ratio of the mole fractions in the pore to the ratio of the mole fractions in the bulk. The selectivity was determined as a function of the pressure for different pore sizes to give some indication of the effect of changing the physical nature of the solid. The pressure can be calculated directly from the input chemical potential using the following standard relationship (for an ideal gas):

$$P = \{\exp(\mu/k_B T)k_B T\}/\Lambda^3 \tag{8.83}$$

The selectivity showed a complicated dependence upon the pore size (Figure 8.17).

The selectivity is best interpreted by considering the interactions between ethane molecules and the walls of the pore. For the smallest pore sizes, the molecules are restricted to the centre of the pore and the ethane molecules are forced to lie flat. As the pore size increases, it becomes possible for ethane to adopt a particularly favourable orientation perpendicular to the walls, with each methyl group being in a potential energy minimum for interaction with the pore atoms. This particular pore size ($2.5\sigma_{CH_4}$) thus has the greatest selectivity for ethane over methane. As the pore size increases further the distribution of ethane becomes more complex, with some layers of ethane lying flat on the pore wall and some in the centre of the pore, with ethane molecules spanning the space between. These arrangements are shown in Figure 8.18.

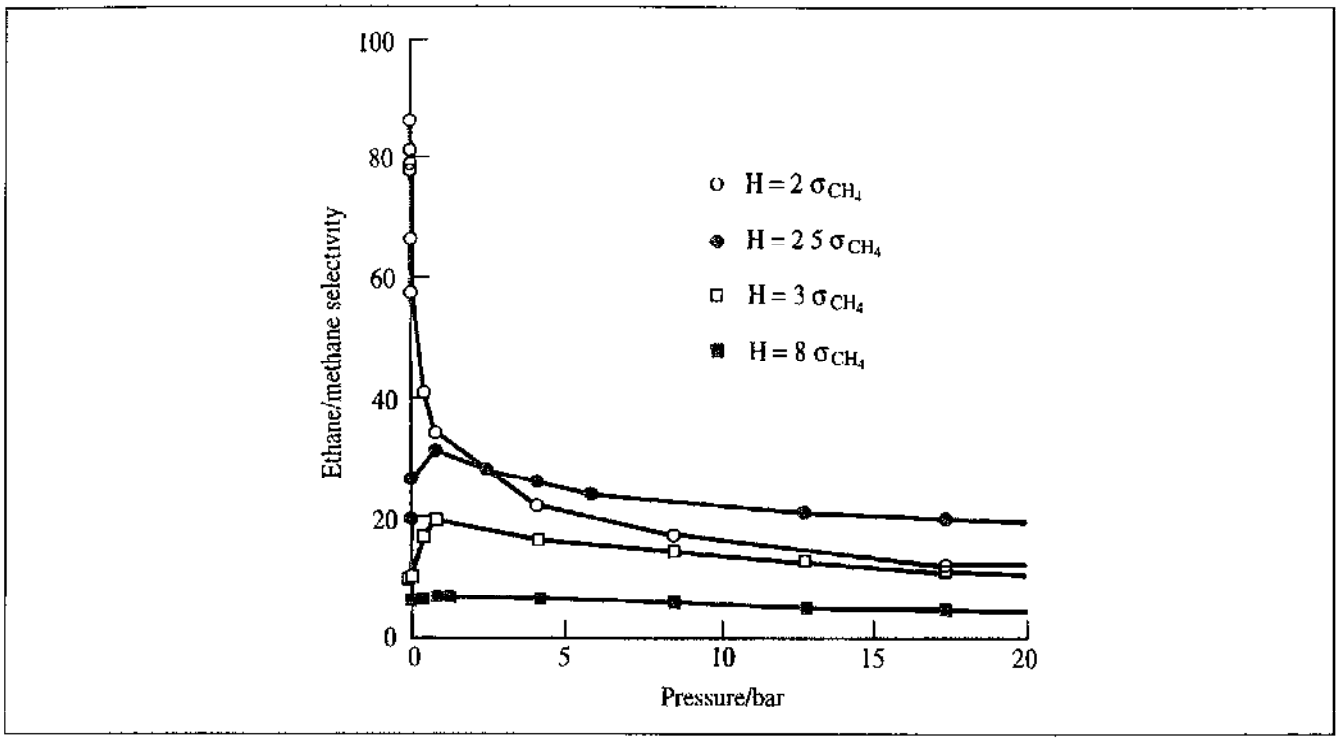


Fig 8 17 Ethane/methane selectivity calculated from grand canonical Monte Carlo simulations of mixtures in slit pores at a temperature of 296 K The selectivity is defined as the ratio of the mole fractions in the pore to the ratio of the mole fractions in the bulk H is the slit width defined in terms of the methane collision diameter σ_{CH_4} (Figure redrawn from Cracknell R F, D Nicholson and N Quirke 1994. A Grand Canonical Monte Carlo Study of Lennard-Jones Mixtures in Slit Pores; 2 Mixtures of Two-Centre Ethane with Methane Molecular Simulation *13 161–175)*

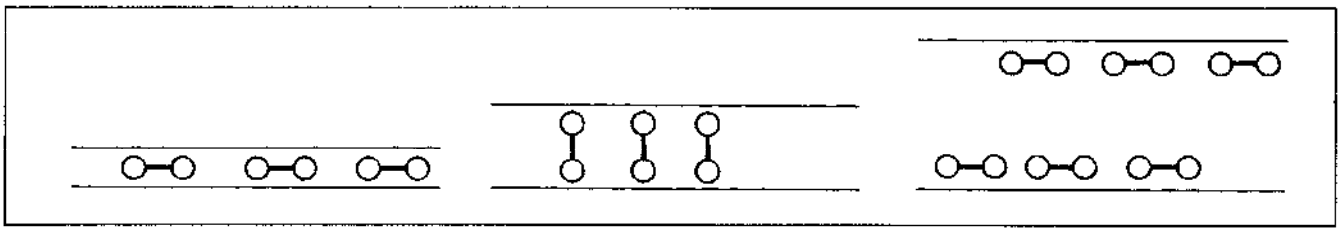


Fig 8 18 Schematic illustration of the arrangements of ethane molecules in slits of varying sizes In the slit of width (2 5σ_{CH_4}) each methyl group is able to occupy a potential minimum from the pore (middle)

8.10 Calculating the Chemical Potential

In a grand canonical simulation the chemical potential is constant. One may also wish to determine how the chemical potential varies during a simulation. The chemical potential is usually determined using an approach due to Widom [Widom 1963], in which a 'test' particle is inserted into the system and the resulting change in potential energy is calculated. The Widom approach is applicable to both molecular dynamics and Monte Carlo simulations Consider a system containing $N-1$ particles, into which we insert another particle at a random position. The inserted particle causes the internal potential energy to change by an amount $\mathscr{V}(\mathbf{r}^{\text{test}})$, i.e. $\mathscr{V}(\mathbf{r}^N) = \mathscr{V}(\mathbf{r}^{N-1}) + \mathscr{V}(\mathbf{r}^{\text{test}})$. Then the configurational

integral for the $N\mathscr{V}$ particle system is given by:

$$Z_N = \int d\mathbf{r}^N \exp[-\mathscr{V}(\mathbf{r}^N)/k_B T] \tag{8.84}$$

or

$$Z_N = \int d\mathbf{r}^N \exp[-\mathscr{V}(\mathbf{r}^{test})/k_B T] \exp[-\mathscr{V}(\mathbf{r}^{N-1})/k_B T] \tag{8.85}$$

By substituting unity in the form Z_{N-1}/Z_{N-1} it is possible to show that $Z_N = Z_{N-1} V \langle \exp[-\mathscr{V}(\mathbf{r}^{test})/k_B T] \rangle$.

The excess chemical potential, that is the difference between the actual value and that of the equivalent ideal gas system, is given by:

$$\mu_{excess} = -k_B T \ln \langle \exp[-\mathscr{V}(\mathbf{r}^{test})/k_B T] \rangle \tag{8.86}$$

The excess chemical potential is thus determined from the average of $\exp[-\mathscr{V}(\mathbf{r}^{test})/k_B T]$. In ensembles other than the canonical ensemble the expressions for the excess chemical potential are slightly different. The ghost particle does not remain in the system and so the system is unaffected by the procedure. To achieve statistically significant results many Widom insertion moves may be required. However, practical difficulties are encountered when applying the Widom insertion method to dense fluids and/or to systems containing molecules, because the proportion of insertions that give rise to low values of $\mathscr{V}(\mathbf{r}^{test})$ falls dramatically. This is because it is difficult to find a 'hole' of the appropriate size and shape.

8.11 The Configurational Bias Monte Carlo Method

Various techniques have been developed to tackle the problem of calculating the chemical potential in cases where the routine Widom method does not give converged results. Of these methods, the configurational bias Monte Carlo (CBMC) method, which was originally introduced by Siepmann [Siepmann 1990], is particularly exciting as it can be applied to assemblies of chain molecules. The configurational bias Monte Carlo method also provides a way to overcome the problems associated with Monte Carlo simulations of assemblies of chain molecules, where many proposed moves are rejected because of high-energy overlaps. The problem of calculating the chemical potential in such cases is clear from the following example. The probability of successfully inserting a single monomer into a fluid of typical liquid density is of the order of 0.5%, or 1 in 200. If one wishes to insert a molecule consisting of n such monomers, the probability is thus approximately 1 in 200^n. For an eight-segment molecule, this probability is less than 1 in 10^{18}, making such calculations impractical The configurational bias Monte Carlo simulation technique can dramatically improve the chances of making a successful insertion.

The essence of the configurational bias Monte Carlo method is that a growing molecule is preferentially directed (i.e. biased) towards acceptable structures The effects of these biases can then be removed by modifying the acceptance rules The configurational bias methods are based upon work published in 1955 by Rosenbluth and Rosenbluth

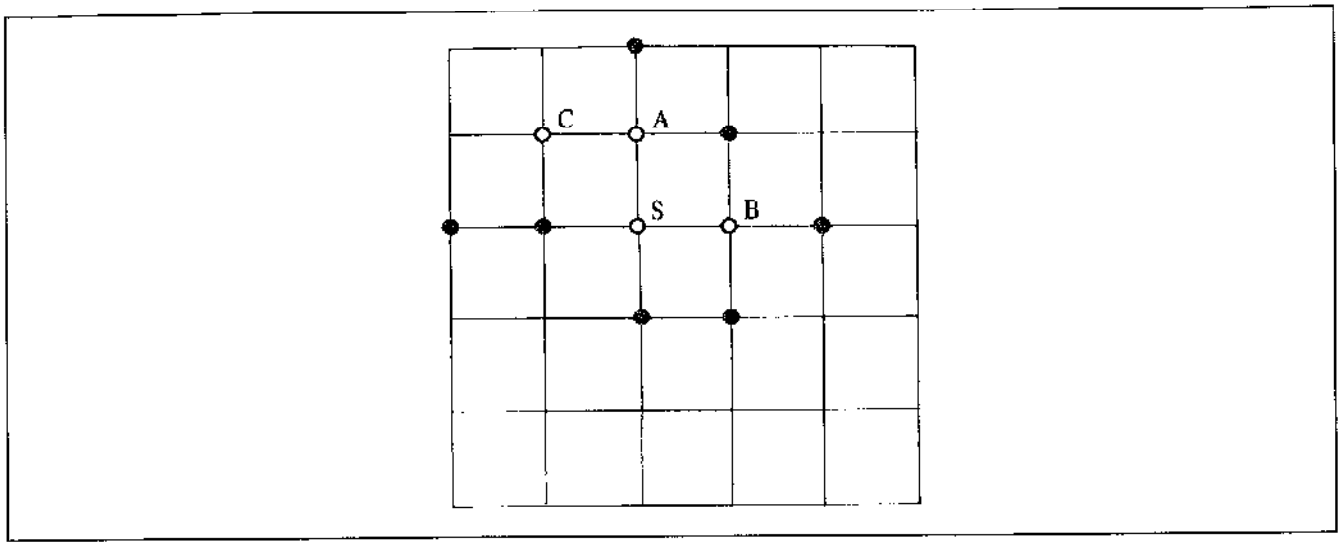


Fig. 8 19 The insertion of a three-unit molecule onto the lattice shown, starting at point S, can be achieved in only one way (see text). (Figure adapted from Siepmann J I 1990 A Method for the Direct Calculation of Chemical Potentials for Dense Chain Systems. Molecular Physics *70 1145–1158.)*

[Rosenbluth and Rosenbluth 1955] and can be applied to both lattice models and to systems with arbitrary molecular potentials and conformations. The method is most easily explained using a two-dimensional lattice model Suppose we wish to insert a three-unit molecule onto the lattice shown in Figure 8.19. First we consider how the conventional approach would tackle this problem The initial step is to select a lattice point at random. Suppose we select the lattice point labelled S in Figure 8.19. We then choose one of the four neighbours of S at random. Of the four neighbouring sites, two are occupied and two are free (A and B in Figure 8 19). There is thus a 50% probability that the move will be rejected at this stage. If we select site B then the molecule can be grown no further as all of the adjacent sites are filled. Were we to grow onto A then we would select one of the three neighbouring sites at random. Of these only site C is available. On average, only one trial in twelve will successfully grow a molecule from S using a conventional Monte Carlo algorithm.

Let us now consider how the configurational bias Monte Carlo method would deal with this problem. Again, the first site (S) is chosen at random. We next consider where to place the second unit. The sites adjacent to S are examined to see which are free. In this case, only two of the four sites are free. One of these free sites is chosen at random. Note that the conventional Monte Carlo procedure selected from all four adjoining sites at random, irrespective of whether it is occupied or not. A *Rosenbluth weight* for the move is then calculated. The Rosenbluth weight for each step i is given by:

$$W_i = \frac{n'}{n} W_{i-1} \tag{8.87}$$

where W_{i-1} is the weight for the previous step ($W_0 = 1$), n' is the number of available sites and n is the total number of neighbouring sites (not including the one occupied by the previous unit) In the case of our lattice, $W_1 = 2/4 = 1/2$. If site B is chosen then there is no site available for the third unit, and so the attempt has to be abandoned If site A is chosen, its adjacent sites are examined to see which are free. In this case there is only one

free site where the third and final unit can be placed. The Rosenbluth weight for this step is $1/3 \times 1/2 = 1/6$. The overall statistical weight for the move is obtained by multiplying the number of successful trials by the Rosenbluth weight of each trial; as half the trials succeed, the statistical weight is therefore $1/2 \times 1/6 = 1/12$. This is exactly the same result that would be obtained with a conventional sampling scheme, though recall that in a conventional scheme only one trial in twelve results in a successful insertion. By contrast, with the configurational bias method the proportion of successful trials is one in two.

The configurational bias algorithm can be extended to take account of intermolecular interactions between the growing chain and its lattice neighbours. If the energy of segment i when occupying a particular site Γ is $v_\Gamma(i)$ then that site is chosen with a probability given by:

$$p_\Gamma(i) = \frac{\exp[-v_\Gamma(i)/k_B T]}{Z_i} \tag{8.88}$$

Z_i is the sum of the Boltzmann factors for all of the b positions considered:

$$Z_i = \sum_{\Gamma=1}^{b} \exp[-v_\Gamma(i)/k_B T] \tag{8.89}$$

The site can be chosen using a biased roulette wheel algorithm, in which the interval between 0 and 1 is divided into b adjacent segments each with a size proportional to the probabilities $p_1(i), p_2(i), \ldots, p_b(i)$ (Figure 8.20). The site within whose interval a random number between 0 and 1 lies is the one chosen. The chain is thus biased towards those sites with a higher Boltzmann weighting; the sum of the Boltzmann factors plays the role of n' in Equation (8.87). The Rosenbluth weight for the entire chain (of length l) can be calculated as:

$$W_l = \exp[-\mathscr{V}_{\text{tot}}(l)/k_B T] \prod_{i=2}^{l} \frac{Z_i}{b} \tag{8.90}$$

where $\mathscr{V}_{\text{tot}}(l)$ is the total energy of the chain, equal to the sum of the individual segment energies $v_\Gamma(i)$. The average Rosenbluth weight is directly related to the excess chemical potential:

$$\mu_{\text{ex}} = k_B T \ln\langle W_l \rangle \tag{8.91}$$

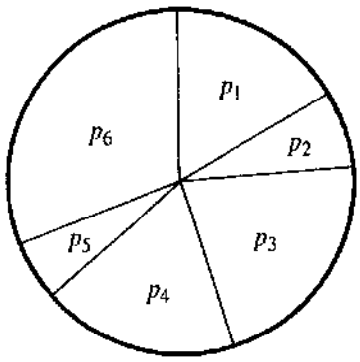


Fig 8 20: A biased roulette wheel chooses states according to their probabilities

If a segment has a zero Rosenbluth weight then growth of the chain is terminated. However, such chains must still be included in the averaging used to determine the excess chemical potential.

So far, we have only considered a fixed number of neighbouring sites for each segment. The method can be extended to cover fully flexible chains, where the set of possible neighbouring positions is infinite [De Pablo *et al.* 1992, 1993]. When growing each segment, a subset containing k random directions is chosen. These trial directions need not be uniformly distributed in space. For each of these orientations the energy $v_\Gamma(i)$ is calculated and so is the Boltzmann factor. An orientation is then chosen with probability:

$$p_\Gamma(i) = \frac{\exp[-v_\Gamma(i)/k_B T]}{\sum_{\Gamma=1}^{k} \exp[-v_\Gamma(i)/k_B T]} \tag{8.92}$$

The Rosenbluth factor is accumulated as follows:

$$W_i = W_{i-1} \frac{1}{k} \sum_{\Gamma=1}^{k} \exp[-v_\Gamma(i)/k_B T] \tag{8.93}$$

To implement this method it is necessary to determine the appropriate number of trial directions, k. If $k = 1$ then the method is equivalent to the original Widom particle insertion method. If k is too large then too much time is taken calculating the Rosenbluth factors for trial positions that are very close in phase space. Frenkel and colleagues have investigated how the choice of k influences the accuracy of the results and the efficiency with which those results were obtained [Frenkel *et al.* 1991] The system they examined was of a flexible chain containing up to 20 segments in a moderately dense atomic fluid. The conventional particle insertion method failed completely for this system. Not surprisingly, the results showed that as the length of the chain increases so the number of random orientations that need to be considered also increases. At least four trial orientations were used at each step and k was chosen to increase logarithmically with the number of segments to be grown. The limiting value of k was considered to be reached when so many trials were required that the configurational bias method was no more efficient than alternative methods of regrowing chains, such as reptation algorithms. For example, for a 6-segment chain the proportion of accepted configurations (once the initial monomer had been inserted successfully) was 0.000 01% for $k = 1$, 3.2% for $k = 10$ and 35% for $k = 50$. For a 20-segment chain the proportion of accepted configurations was 0.0001% for $k = 20$, 0.66% for $k = 50$ and 2.0% for $k = 100$.

The Rosenbluth algorithm can also be used as the basis for a more efficient way to perform Monte Carlo sampling for fully flexible chain molecules [Siepmann and Frenkel 1992], which, as we have seen, is difficult to do as bond rotations often give rise to high energy overlaps with the rest of the system.

The configurational bias Monte Carlo method involves three types of move. Two of these are translational or rotational moves of the entire molecule, which are performed in the conventional way. The third type of move is a conformational change. A chain is selected at random and one of the segments within it is also randomly chosen. That part of the chain that lies above or below the segment (chosen with equal probability) is discarded and an

attempt is made to regrow the discarded portion. Let us consider first the case where each segment is restricted to a given number of discrete orientations, either because the chain is restricted to a lattice or because the model discretely samples the conformational space (e.g. it only permits *gauche* and *trans* conformations to a hydrocarbon chain). At each stage, the Boltzmann weights of the b discrete conformations are determined and one of the sites is chosen with a probability given by Equation (8.92). The Rosenbluth weight is determined for the growing chain using Equation (8.93).

Having generated a trial conformation it must be decided whether to accept it or not. To do this a random number is generated in the range 0–1 and compared with the ratio of the Rosenbluth weights for the trial conformation ($W_{l,\text{trial}}$) and the old conformation ($W_{l,\text{old}}$). The new chain is then accepted using the following criterion:

$$\text{rand}(0,1) \leq \frac{W_{l,\text{trial}}}{W_{l,\text{old}}} \tag{8.94}$$

A similar approach can be adopted with continuous chains. Here it is also possible to enhance the sampling by guiding the choice of trial sites towards those with a particularly favourable intramolecular energy. This can be achieved by generating random vectors on the surface of a sphere of unit radius for each segment. The potential energy (angle-bending and torsional) for a bond directed along this vector is calculated. The vector is then accepted or rejected using the Metropolis criterion. If it is accepted, the vector is scaled to the appropriate bond length. This procedure continues until the desired number of trial sites have been generated. A trial site is then selecting using Boltzmann factors, which only consider the intra- and intermolecular non-bonded interactions of the sites with the chain and with the rest of the system. The Rosenbluth weights are similarly calculated and the move is accepted according to the ratio of the new and old Rosenbluth weights. Again, the correct choice of the number of trial sites is crucial to the efficiency of the method.

For branched molecules some modifications are required to the configurational bias method as described so far. This is because there may be bond angles which share the same central atom and torsion angles which have the same central two atoms in common. Thus in 2-methylalkanes the bond angles to the two terminal methyl groups share the 2-carbon atom. In 3,4-dimethylhexane there is a potential torsion problem. In the 'standard' configurational bias method one of the methyl groups would be grown, followed by the second. What is sometimes observed, however, is that the distributions of these bond angles is not equal (as it should be, as they are equivalent). Two possible ways to deal with this problem are to grow both atoms simultaneously [Dijkstra 1997] or to use a small Monte Carlo simulation to generate the trial positions [Vlugt *et al.* 1999]. When there are multiple torsion angles these two methods are not suitable; indeed, for a molecule such as 2,3-dimethylbutane the entire molecule must be generated in a single step. Martin and Siepmann suggested that it was possible to decouple the selection of the different energy terms [Martin and Siepmann 1999]. Suppose that the Lennard-Jones, torsional and bond-angle terms are decoupled Then the probability of generating a particular configuration is given by:

$$P = \prod_{n=1}^{n_{\text{step}}} \left[\frac{\exp(-v_{\text{LJ}}(i)/k_{\text{B}}T)}{W_{\text{L}}(n)}\right]\left[\frac{\exp(-v_{\text{tor}}(j)/k_{\text{B}}T)}{W_{\text{T}}(n)}\right]\left[\frac{\exp(-v_{\text{bend}}(k)/k_{\text{B}}T)}{W_{\text{B}}(n)}\right] \tag{8.95}$$

The relevant Rosenbluth weights are:

$$W_L(n) = \sum_{i=1}^{n_{LJ}} \exp(-v_{LJ}(i)/k_B T) \tag{8.96}$$

$$W_T(n) = \sum_{j=1}^{n_{tor}} \exp(-v_{tor}(j)/k_B T) \tag{8.97}$$

$$W_B(n) = \sum_{k=1}^{n_{bend}} \exp(-v_{bend}(k)/k_B T) \tag{8.98}$$

where n_{LJ}, n_{tor} and n_{bend} are the number of trial sites for the Lennard-Jones, torsional and angle-bending interactions, respectively. Under these conditions, the move is accepted with a probability:

$$P_{acc} = \min\left[1, \frac{\prod_{n=1}^{n_{step}} W_L(n)_{new} W_T(n)_{new} W_B(n)_{new}}{\prod_{n=1}^{n_{step}} W_L(n)_{old} W_T(n)_{old} W_B(n)_{old}}\right] \tag{8.99}$$

The advantage of this decoupling method is that a large number of trial sites can be chosen for the computationally less expensive bond angle selection without increasing the cost of performing the other selections. Once the bond angle distribution has been chosen by this biased method, it is used as input to a biased selection of the torsional and Lennard-Jones interactions. An extension to this decoupling procedure involves grouping the torsional and Lennard-Jones together and having each biased selection of bond angles send many possible conformations forward to the next step. This coupling and decoupling of terms is claimed to provide a great deal of flexibility when designing a configurational bias scheme for any particular molecule and would also be applicable to force field models that included additional terms such as bond stretching or cross terms.

8.11.1 Applications of the Configurational Bias Monte Carlo Method

The CBMC method has been used to investigate a number of systems involving long-chain alkanes. Siepmann and McDonald examined a monolayer of 90 $CH_3(CH_2)_{15}SH$ molecules chemisorbed onto a gold surface [Siepmann and McDonald 1993a, b]. The thiol group forms a bond with the gold surface atoms, thus producing a high degree of surface ordering of the adsorbed molecules. Spectroscopic experiments indicated that the chains were tilted relative to the surface and adopted a predominantly *trans* conformation for the alkyl links. Both discrete and continuous versions of the configurational bias Monte Carlo method were employed; in the discrete model, each CH_2CH_2 segment was restricted to the *trans* and two *gauche* conformations. In the continuous simulation, six trial sites were used for each segment. The molecules were initially placed on a triangular lattice in an extended conformation perpendicular to the surface.

In the structure obtained at the end of the simulation the chains were ordered in an approximately hexagonal pattern. During equilibration *gauche* conformations were introduced into the alkyl chains, causing the system to tilt. However, once the molecules had all tilted the

gauche defects were gradually squeezed out to give chains with predominantly *trans* links. The final configuration is shown in Figure 8 21 (colour plate section)

The configurational bias Monte Carlo method has also been used to investigate the adsorption of alkanes in zeolites. Such systems are of especial interest in the petrochemical industry. One interesting experimental result obtained for the zeolite silicalite was that short-chain alkanes (C_1 to C_5) and long-chain alkanes (C_{10}) have simple adsorption isotherms but hexane and heptane show kinked isotherms. Such systems are obvious candidates for theoretical investigations because the experimental data is difficult and time-consuming to obtain. Moreover, the simulation can often provide a detailed molecular explanation for the observed behaviour. The simulation of such systems is difficult using conventional methods; the Monte Carlo method suffers from the problems of low acceptance ratios or a very slow exploration of phase space, and long simulation times would be required with molecular dynamics as the diffusion of long-chain alkanes is very slow. The configurational bias Monte Carlo method enabled effective and efficient simulations to be performed, providing both thermodynamic properties and the spatial distribution of the molecules within the zeolite [Smit and Siepmann 1994; Smit and Maesen 1995]. The adsorption isotherms (i.e. the number of molecules adsorbed as a function of the applied pressure) were calculated using grand canonical simulations in which the zeolite was coupled to a reservoir at constant temperature and chemical potential.

Silicalite has both straight and zig-zag channels, which are connected via intersections (Figure 8.22). An analysis of the configurations showed that the distribution of a short alkane, such as butane, was approximately equal between the two types of channel. However, as the length of the alkane chain was increased, so there was a greater probability of finding it in a straight channel than in a zig-zag channel. Hexane is an interesting case, for its length is almost equal to the period of the zig-zag channels. At low pressure the hexane molecules move freely in the zig-zag channels and occupy the intersections for part of the time. To fill the zeolite with hexane it is first necessary for the alkane molecules to occupy just the zig-zag channels and not the intersections. This is accompanied by a loss of entropy, which must be compensated for by a higher chemical potential and so gives the kinked isotherm. The straight channels can then be filled with hexane. Different behaviour

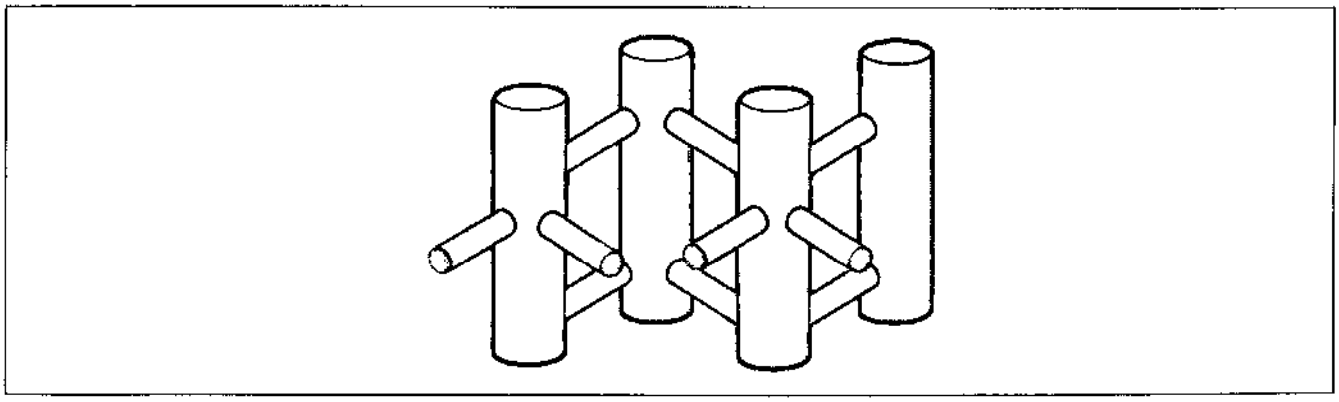


Fig 8 22 Schematic structure of the zeolite silicalite showing the straight and zig-zag channels (Figure adapted from Smit B and J I Siepmann 1994 Simulating the Adsorption of Alkanes in Zeolites Science **264** *1118–1120)*

is observed for smaller alkanes because more than one molecule can occupy the zig-zag channels. Longer alkanes always partially occupy the intersection and so there is no benefit from freezing the molecules in the zig-zag channels. It is also possible to simulate the behaviour of branched alkanes [Vlugt *et al.* 1998] and compare these with their linear equivalents. Thus, whereas *n*-butane has an equal probability of being in either channel, isobutane has a preference for the intersection. Moreover, once all the intersections are full it requires considerable energy to place isobutane elsewhere in the zeolite. Thus requires a much higher pressure, giving rise to an inflection in the adsorption isotherms

8.12 Simulating Phase Equilibria by the Gibbs Ensemble Monte Carlo Method

The most 'obvious' way to investigate phase equilibria is to set up an appropriate system with a conventional simulation technique. Unfortunately, simulations of systems with more than one phase usually require inordinate amounts of computer time. There are several reasons why the use of conventional simulation methods to investigate phase equilibria is difficult. First, it would take a very long time to equilibrate such a system, which would need to separate into its two phases (e g. liquid and vapour). The properties of the fluid in the interfacial region differ substantially from the properties in the bulk and so to obtain a 'bulk' measurement all of the interfacial atoms must be ignored. Smit has calculated the percentage of the number of particles in the interfacial region for systems of varying sizes, these percentages range from 10% in the interfacial region for a system of 50 000 particles to 95% for a system of 100 particles [Smit 1993]. To simulate phase equilibria directly would thus require long simulations to be performed on systems containing many particles.

The Gibbs ensemble Monte Carlo simulation method, invented by Panagiotopoulos [Panagiotopoulos 1987], enables phase equilibria to be studied directly using small numbers of particles. Rather than trying to form an interface within a single simulation, two simulation boxes are used, each representing one of the two phases. There is no physical interface between the two boxes, which are subject to the usual periodic boundary conditions (Figure 8.23). Three types of move are possible. The first type of move comprises particle displacements within each box, as in a conventional Monte Carlo simulation. The second type of move involves volume changes of the two boxes by equal and opposite amounts so that the total volume of the system remains constant. The third type of move involves the removal of a particle from one box and its attempted placement in the other box. This is identical to the Widom insertion method for calculating the chemical potential. Indeed, as the energy of the inserted particle must be calculated, it is possible to determine the chemical potential in the Gibbs ensemble without any additional computational cost. These three types of move are often performed in strict order, but it may be better to choose each type of move at random, ensuring that, on average, the appropriate numbers of each type of move are made.

The properties of the Gibbs ensemble Monte Carlo simulation method have been examined in great detail using simple systems such as the Lennard-Jones fluid and simple gases. A

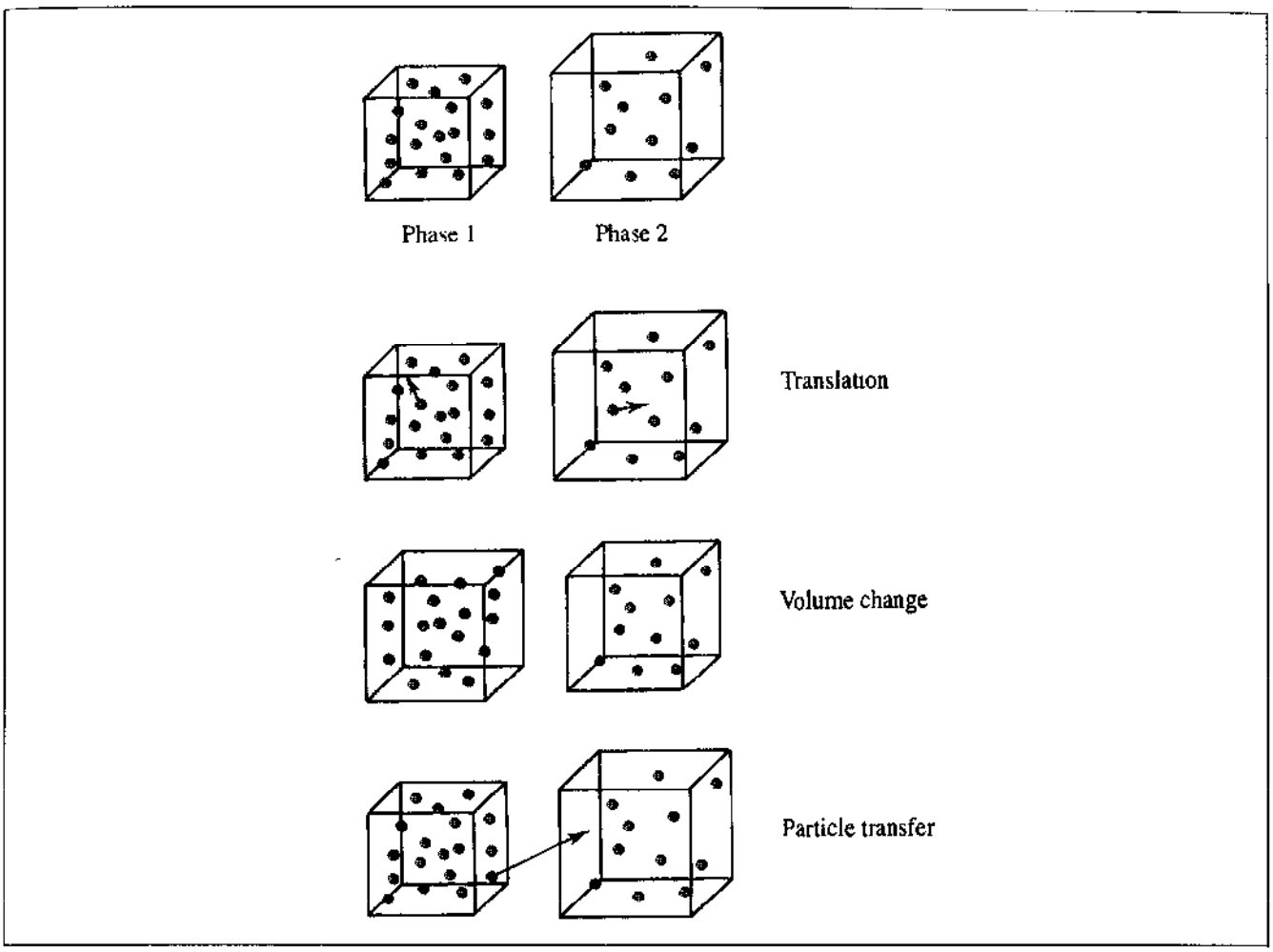


Fig 8 23 The Gibbs ensemble Monte Carlo simulation method uses one box for each of the two phases. Three types of move are permitted translations within either box; volume changes (keeping the total volume constant) and transfer of a particle from one box to the other

particularly exciting development is the use of the configurational bias Monte Carlo method in conjunction with the Gibbs ensemble method to construct the phase diagrams of complex, long-chain molecules. For example, the vapour–liquid phase equilibria of *n*-pentane and *n*-octane have been investigated by Siepmann, Karaborni and Smit using this combined approach on systems containing 200 pentane or 160 octane molecules [Siepmann *et al.* 1993a]. The calculated properties of these two systems agreed very well with the available experimental data, particularly for the shorter alkane. Their studies were subsequently extended to much longer alkanes (up to C_{48}) [Siepmann *et al.* 1993b]. One particularly noteworthy result was that the density at the critical point increased with the length of the carbon chain up to *n*-octane but then *decreased* as the chain increased in length. Until shortly before the simulations were performed it had been assumed that the critical density for longer chains could be extrapolated from the experimental data obtained with short chains under the assumption that the critical density increased with the length of the chain. Later experiments were able to examine longer chains and did indeed demonstrate that the critical point density passed through a maximum at octane and then decreased for shorter chain lengths.

8.13 Monte Carlo or Molecular Dynamics?

In principle, the modeller has the choice of using either the Monte Carlo or molecular dynamics technique for a given simulation. In practice one technique must be chosen over the other. Sometimes the decision 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, molecular dynamics is required if one wishes to calculate time-dependent quantities such as transport coefficients. Conversely, Monte Carlo 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 Monte Carlo method than using the sometimes awkward and ill-defined constant temperature and constant pressure molecular dynamics simulation methods. The Monte Carlo 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 Monte Carlo simulation often gives much more rapid convergence of the calculated thermodynamic properties of a simple molecular liquid (modelled as a rigid molecule), but it may explore the phase space of large molecules very slowly due to the need for small steps unless special techniques such as the configurational bias Monte Carlo method are employed. However, the ability of the Monte Carlo 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. Molecular dynamics 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 Molecular dynamics 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 Monte Carlo method may be more effective for conformational changes, which jump to a completely different area of phase space.

Given that the two techniques in some ways complement each other in their ability to explore phase space, it is not surprising that there has been some effort to combine the two methods. Some of the techniques that we have considered in this chapter and in Chapter 7 incorporate elements of the Monte Carlo and molecular dynamics techniques. Two examples are the stochastic collisions method for performing constant temperature molecular dynamics, and the force bias Monte Carlo method. More radical combinations of the two techniques are also possible.

An obvious way to combine Monte Carlo with molecular dynamics is to use each technique for the most appropriate part of a simulation. For example, when simulating a solvated macromolecule, the equilibration phase is usually performed in a series of stages. In the first stage, the solute is kept fixed while the solvent molecules (and any ions, if present) are allowed to move under the influence of the solute's electrostatic field. This solvent equilibration may often be performed more effectively using a Monte Carlo simulation as the solvent and ions do not have any appreciable conformational flexibility. To simulate the whole system, molecular dynamics is then the most appropriate method. Such a protocol has been used to perform long simulations of DNA molecules [Swaminathan *et al.* 1991].

A variety of hybrid molecular dynamics/Monte Carlo methods have been devised, in which the simulation algorithm alternates between molecular dynamics and Monte Carlo The aim of such methods is to achieve better sampling, and thereby more rapid convergence of thermodynamic properties. *In extremis*, each molecular dynamics (or stochastic dynamics) step is followed by a Monte Carlo step, the velocities being unaffected by acceptance or rejection of the Monte Carlo step. Such a method has been devised by Guarnieri and Still [Guarnieri and Still 1994]. An alternative is to perform a block of molecular dynamics steps to generate a new state, which is then accepted or rejected on the basis of the total energy (potential plus kinetic) using the usual Metropolis criterion. If the new coordinates are rejected then the original coordinates from the start of the block are restored and molecular dynamics is run again, but with an entirely new set of velocities that is chosen from a Gaussian distribution. This approach is very similar to the stochastic collisions method for temperature control discussed in Section 7.7.1 but with the addition of a Monte Carlo acceptance or rejection step [Duane *et al* 1987]. A simulation using this hybrid algorithm samples from the canonical ensemble (constant temperature) and was shown by Clamp and colleagues to be more effective than conventional molecular dynamics or Monte Carlo methods for exploring the phase space of both simple model systems and proteins [Clamp *et al.* 1994].

Appendix 8.1 The Marsaglia Random Number Generator

The Marsaglia random number generator [Marsaglia *et al.* 1990] is known as a *combination generator* because it is constructed from two different generators. It has a period of about 2^{144}. The first generator is a lagged Fibonacci generator that performs the following binary operation on two real numbers x and y:

$$x \bullet y = x - y \text{ if } x \geq y; \quad x \bullet y = x - y - 1 \text{ if } x < y \tag{8.100}$$

The values x and y are chosen from numbers earlier in the sequence, so that the nth value in the sequence is calculated by:

$$x_n = x_{n-r} \bullet x_{n-s} \tag{8.101}$$

r and s are the *lags*, which are chosen to give numbers that are satisfactorily random and have a long period. Marsaglia chose $r = 97$ and $s = 33$. The algorithm does therefore require the last 97 numbers to be stored at all stages.

The second generator is an arithmetic sequence method that generates random numbers using the following mathematical operation:

$$c \circ d = c - d \text{ if } c \geq d, \quad c \circ d = c - d + 16\,777\,213/16\,777\,216 \text{ if } c < d \tag{8.102}$$

The nth value in this sequence is given by:

$$c_n = c_{n-1} \circ (7654\,321/16\,777\,216) \tag{8.103}$$

The nth number, U_n, in the combined sequence is then obtained as

$$U_n = x_n \circ c_n \tag{8.104}$$

The c sequence requires one initial seed value and the x sequence requires 97 initial seeds (which should themselves be reasonably random). These can be supplied by the user but in the published algorithm these 97 values were obtained from another combination generator comprising a lagged Fibonacci generator and a congruential algorithm.

Further Reading

Adams D J 1983. Introduction to Monte Carlo Simulation Techniques In Perran J W (Editor) *Physics of Superionic Conductors and Electrode Materials.* New York, Plenum, pp 177–195.

Allen M P and D J Tildesley 1987 *Computer Simulation of Liquids* Oxford, Oxford University Press.

Colbourn E A (Editor) 1994 *Computer Simulation of Polymers* Harlow, Longman.

Frenkel D Monte Carlo Simulations: A Primer In van Gunsteren W F, P K Weiner and A J Wilkinson (Editors). *Computer Simulation of Biomolecular Systems* Volume 2 Leiden, ESCOM, pp. 37–66

Galaiatsatos V 1995 Computational Methods for Modelling Polymers. An Introduction. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 6. New York, VCH Publishers, pp. 149–208.

Kaols M H and P A Whitlock 1986. *Monte Carlo Methods, Volume 1. Basics.* New York, John Wiley & Sons.

Kermer K 1993 Computer Simulation of Polymers. In Allen M P and D J Tildesley (Editors) *Computer Simulation in Chemical Physics.* Dordrecht, Kluwer, NATO ASI Series **397**.397–459

Rubinstein R Y 1981. *Simulation and Monte Carlo Methods.* New York, John Wiley & Sons.

References

Barker J A and R O Watts 1969 Structure of Water, A Monte Carlo Calculation *Chemical Physics Letters* **3**.144–145.

Baschnagel J, K Binder, W Paul, M Laso, U Suter, I Batoulis, W Jilge and T Bürger 1991. On the Construction of Coarse-Grained Models for Linear Flexible Polymer Chains – Distribution Functions for Groups of Consecutive Monomers. *Journal of Chemical Physics* **95**:6014–6025.

Clamp M E, P G Baker, C J Stirling and A Brass 1994 Hybrid Monte Carlo· An Efficient Algorithm for Condensed Matter Simulation. *Journal of Computational Chemistry* **15**.838–846.

Cracknell R F, D Nicholson and N Quirke 1994 A Grand Canonical Monte Carlo Study of Lennard-Jones Mixtures in Slit Pores; 2 Mixtures of Two-Centre Ethane with Methane *Molecular Simulation* **13**·161–175.

De Pablo J J, M Laso, J I Siepmann and U W Suter 1993 Continuum–Configurational Bias Monte Carlo Simulations of Long-chain Alkanes. *Molecular Physics* **80** 55–63

De Pablo J J, M Laso, and U W Suter 1992. Estimation of the Chemical Potential of Chain Molecules by Simulation. *Journal of Chemical Physics* **96**:6157–6162.

Dijkstra M 1997 Confined Thin Films of Linear and Branched Alkanes. *Journal of Chemical Physics* **107** 3277–3288.

Duane S, A D Kennedy and B J Pendleton 1987. Hybrid Monte Carlo *Physics Letters* **B195**:216–222

Flory P J 1969. *Statistical Mechanics of Chain Molecules.* New York, Interscience.

Frantz D D, D L Freeman and J D Doll 1990 Reducing Quasi-ergodic Behavior in Monte Carlo Simulations by J-walking Applications to Atomic Clusters *Journal of Chemical Physics* **93**.2769–2784

Frenkel D D, C A M Mooij and B Smit 1991 Novel Scheme to Study Structural and Thermal Properties of Continuously Deformable Materials. *Journal of Physics Condensed Matter* **3**:3053–3076.

Guarnieri F and W C Still 1994. A Rapidly Convergent Simulation Method Mixed Monte Carlo/ Stochastic Dynamics. *Journal of Computational Chemistry* **15** 1302–1310

Marsaglia G, A Zaman and W W Tsang 1990. Towards a Universal Random Number Generator *Statistics and Probability Letters* **8**.35–39.

Martin M G and J I Siepmann 1999. Novel Configurational-bias Monte Carlo Method for Branched Molecules Transferable Potentials for Phase Equilibria. 2 United-atom Description of Branched Alkanes *Journal of Physical Chemistry* **103**:4508–4517

Metropolis N, A W Rosenbluth, M N Rosenbluth, A H Teller and E Teller 1953 Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics* **21**:1087–1092

Okamoto Y and U H E Hansmann 1995. Thermodynamics of Helix-coil Transitions Studied by Multicanonical Algorithms. *Journal of Physical Chemistry* **99**:11276–11287

Panagiotopoulos A Z 1987. Direct Determination of Phase Coexistence Properties of Fluids by Monte Carlo Simulation in a New Ensemble. *Molecular Physics* **61**.813–826

Pangali C, M Rao and B J Berne 1978 On a Novel Monte Carlo Scheme for Simulating Water and Aqueous Solutions *Chemical Physics Letters* **55** 413–417.

Rao M and B J Berne 1979 On the Force Bias Monte Carlo Simulation of Simple Liquids. *Journal of Chemical Physics* **71**.129–132

Rosenbluth M N and A W Rosenbluth 1955. Monte Carlo Calculation of the Average Extension of Molecular Chains. *Journal of Chemical Physics* **23**:356–359

Rossky P J, J D Doll and H L Friedman 1978. Brownian Dynamics as Smart Monte Carlo Simulation. *Journal of Chemical Physics* **69**.4628–4633.

Senderowitz H, F Guarnieri and W C Still 1995. A Smart Monte Carlo Technique for Free Energy Simulations of Multicanonical Molecules. Direct Calculations of the Conformational Populations of Organic Molecules. *Journal of the American Chemical Society* **117** 8211–8219

Sharp W E and C Bays 1992 A Review of Portable Random Number Generators. *Computers and Geosciences* **18**·79–87

Siepmann J I 1990. A Method for the Direct Calculation of Chemical Potentials for Dense Chain Systems. *Molecular Physics* **70**.1145–1158.

Siepmann J I and D Frenkel 1992 Configurational Bias Monte Carlo· A New Sampling Scheme for Flexible Chains *Molecular Physics* **75**:59–70.

Siepmann J I, S Karaborni and B Smit 1993a Vapor–Liquid Equilibria of Model Alkanes *Journal of the American Chemical Society* **115** 6454–6455.

Siepmann J I, S Karaborni and B Smit 1993b. Simulating the Crucial Behaviour of Complex Fluids *Nature* **365**:330–332

Siepmann J I and I R McDonald 1993a Domain Formation and System-size Dependence in Simulations of Self-assembled Monolayers *Langmuir* **9**:2351–2355.

Siepmann J I and I R McDonald 1993b Monte Carlo Study of the Properties of Self-assembled Monolayers Formed by Adsorption of $CH_3(CH_2)_{15}SH$ on the (111) Surface of Gold. *Molecular Physics* **79**:457–473.

Smit B 1993 Computer Simulation in the Gibbs Ensemble. In Allen M P and D J Tildesley (Editors) *Computer Simulation in Chemical Physics* Dordrecht, Kluwer. NATO ASI Series 397, pp. 173–210

Smit B and T L M Maesen 1995. Commensurate 'Freezing' of Alkanes in the Channels of a Zeolite. *Nature* **374**.42–44.

Smit B and J I Siepmann 1994. Simulating the Adsorption of Alkanes in Zeolites *Science* **264** 1118–1120.

Stephenson G 1973 *Mathematical Methods for Science Students*. London, Longman

Swaminathan S, G Ravishanker and D L Beveridge 1991. Molecular Dynamics of B-DNA Including Water and Counterions – A 140-ps Trajectory for d(CGCGAATTCGCG) Based on the Gromos Force Field *Journal of the American Chemical Society* **113**·5027–5040

Verdier P H and W H Stockmayer 1962. Monte Carlo Calculations on the Dynamics of Polymers in Dilute Solution. *Journal of Chemical Physics* **36**·227–235

Vesely F J 1982. Angular Monte Carlo Integration Using Quaternion Parameters· A Spherical Reference Potential for CCl_4 *Journal of Computational Physics* **47** 291–296

Vlugt T J H, R Krishna and B Smit 1999 Molecular Simulations of Adsorption Isotherms for Linear and Branched Alkanes and Their Mixtures in Silicalite *Journal of Physical Chemistry* **103**.1102–1118

Vlugt T J H, W Zhu, F Kapteijn, J A Moulijn, B Smit and R Krishna 1998 Adsorption of Linear and Branched Alkanes in the Zeolite Silicalite-1. *Journal of the American Chemical Society* **120**.5599–5600.

Widom B 1963. Topics in the Theory of Fluids. *Journal of Chemical Physics* **39** 2808–2812

Xu H and B J Berne 1999. Multicanonical Jump Walking: A Method for Efficiently Sampling Rough Energy Landscapes. *Journal of Chemical Physics* **110** 10299–10306

CHAPTER NINE

Conformational Analysis

9.1 Introduction

The physical, chemical and biological properties of a molecule often depend critically upon the three-dimensional structures, or *conformations*, that it can adopt. Conformational analysis is the study of the conformations of a molecule and their influence on its properties. The development of modern conformational analysis is often attributed to D H R Barton, who showed in 1950 that the reactivity of substituted cyclohexanes was influenced by the equatorial or axial nature of the substituents [Barton 1950]. An equally important reason for the development of conformational analysis at that time was the introduction of analytical techniques such as infrared spectroscopy, NMR and X-ray crystallography, which actually enabled the conformation to be determined.

The conformations of a molecule are traditionally defined as those arrangements of its atoms in space that can be interconverted purely by rotation about single bonds. This definition is usually relaxed in recognition of the fact that small distortions in bond angles and bond lengths often accompany conformational changes, and that rotations can occur about bonds in conjugated systems that have an order between one and two.

A key component of a conformational analysis is the *conformational search*, the objective of which is to identify the 'preferred' conformations of a molecule, those conformations that determine its behaviour. This usually requires us to locate conformations that are at minimum points on the energy surface. Energy minimisation methods therefore play a crucial role in conformational analysis. An important feature of methods for performing energy minimisation is that they move to the minimum point that is closest to the starting structure. For this reason, it is necessary to have a separate algorithm which generates the initial starting structures for subsequent minimisation. It is these algorithms for generating initial structures that will be a major focus of this chapter. It is important to recognise the difference between a conformational search and a molecular dynamics or Monte Carlo simulation; the conformational search is concerned solely with locating minimum energy structures, whereas the simulation generates an ensemble of states that includes structures not at energy minima. However, as we shall see, both the Monte Carlo and molecular dynamics methods can be used as part of a conformational search strategy.

If possible, it is desirable to identify all minimum energy conformations on the energy surface However, the number of minima may be so large that it is impractical to contemplate finding them all. Under such circumstances it is usual to try to find all the accessible minima. The relative populations of a molecule's conformations can be calculated using statistical mechanics via the Boltzmann distribution, though it is important to remember that the statistical weights involve contributions from all the degrees of freedom, including the vibrations as

well as the energies. Solvation effects may also be important, and various schemes are now available for calculating the solvation free energy of a conformation, which can be added to the intramolecular energy. These solvation schemes (which will be discussed in more detail in Section 11.9) provide computationally efficient ways to include the effects of the solvent on conformational equilibria. For some molecules such as proteins there are so many minima on the energy surface that it is impractical to try to find them all. Under such circumstances, it is often assumed that the native (i.e. naturally occurring) conformation is the one with the very lowest value of the energy function. This conformation is usually referred to as the *global minimum energy conformation*. One should usually be wary of algorithms which find only a single conformation. For example, even though the global minimum energy conformation has the lowest energy, it may not be the most highly populated because of the contribution of the vibrational energy levels to the statistical weight of each structure. Moreover, the global minimum energy conformation may not be the active (i.e. functional) structure. Indeed, in some cases it is possible that the active conformation does not correspond to any minimum on the energy surface of the isolated molecule. It may even be necessary for a molecule to adopt more than one conformation. For example, a substrate might bind in one conformation to an enzyme and then adopt a different conformation prior to reaction.

Conformational search methods can be conveniently divided into the following categories: systematic search algorithms, model-building methods, random approaches, distance geometry and molecular dynamics. Before discussing these methods, we should note that conformational analysis can sometimes be performed quite effectively using Dreiding or CPK mechanical models. The invention of these models should be regarded as an important development in conformational analysis and molecular modelling. Mechanical models do, however, have some shortcomings. For example, they provide no quantitative information about the relative energies of the various conformations. It is often quite difficult to make accurate measurements of a molecule's internal coordinates such as the distance between two atoms that are on opposite sides of the structure. They are subject to the forces of gravity (which makes them unwieldy for large molecules) and the hands of marauding colleagues! It is also difficult to construct models that have significant deviations from standard bond lengths and angles. Nevertheless, manual models can be very useful, particularly as they are portable and because they can be easily manipulated in a way that is often not possible with computer images (although this may change with the development of 'virtual reality' molecular modelling systems).

We next introduce the basic algorithms and then describe some of the many variants upon them. We then discuss two methods called evolutionary algorithms and simulated annealing, which are generic methods for locating the globally optimal solution. Finally, we discuss some of the ways in which one might analyse the data from a conformational analysis in order to identify a 'representative' set of conformations.

9.2 Systematic Methods for Exploring Conformational Space

As the name suggests, a systematic search explores the conformational space by making regular and predictable changes to the conformation. The simplest type of systematic search

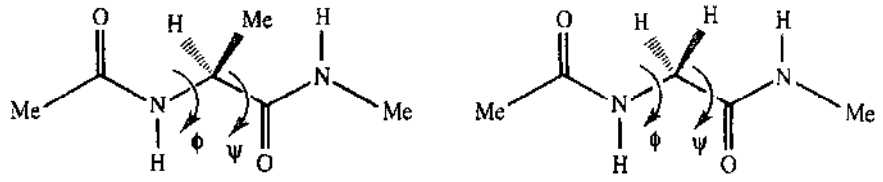


Fig 9 1 The alanine dipeptide and the glycine dipeptide

(often called a *grid search*) is as follows. First, all rotatable bonds in the molecule are identified. The bond lengths and angles remain fixed throughout the calculation. Each of these bonds is then systematically rotated through 360° using a fixed increment. Every conformation so generated is subjected to energy minimisation to derive the associated minimum energy conformation. The search stops when all possible combinations of torsion angles have been generated and minimised. To illustrate the grid search algorithm, let us consider the conformational energy surface for the 'alanine dipeptide' $CH_3CONHCHMeCONHCH_3$ (Figure 9.1), which is used as a model for the conformational behaviour of amino acids in proteins. If we assume that the bond lengths and bond angles are fixed and that the amide bonds adopt *trans* conformations then only the two torsion angles labelled ϕ and ψ in Figure 9.1 can vary. The energy is then a function of just these two variables, and as such it can be represented as a contour diagram as shown in Figure 9.2. This contour plot is known as a *Ramachandran map*, after G N Ramachandran who showed that the amino acids were restricted to a limited range of conformations [Ramachandran *et al.* 1963]. The accessible areas on the contour maps calculated by Ramachandran do indeed correspond to those conformations that are observed in X-ray structures of proteins (Figure 9.3). Two regions are particularly important; these correspond to the α-helix and β-strand structures,

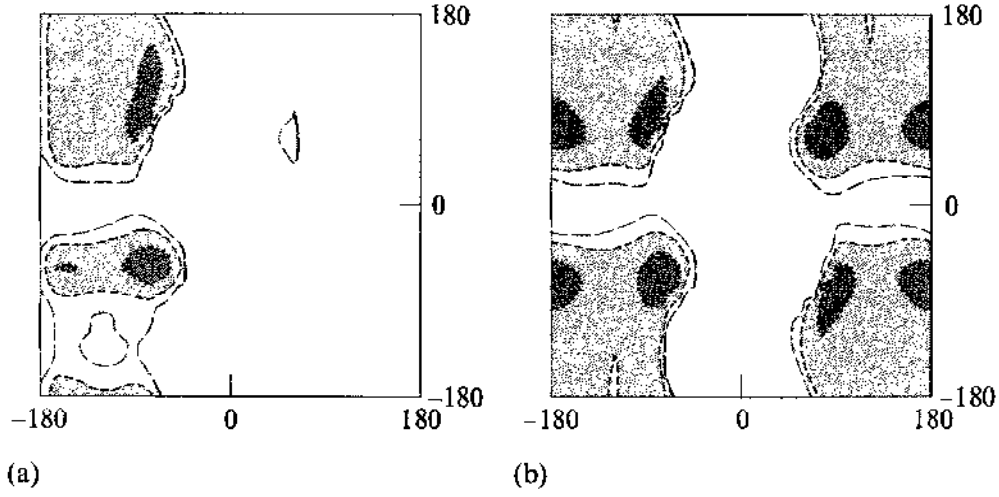


Fig 9.2· Ramachandran map for the alanine dipeptide (a) and glycine dipeptide (b), calculated using the AMBER force field [Weiner et al *1984]. In both cases contours are drawn at 1 0, 2 0 and 3 0 kcal/mol above the lowest-energy conformation found*

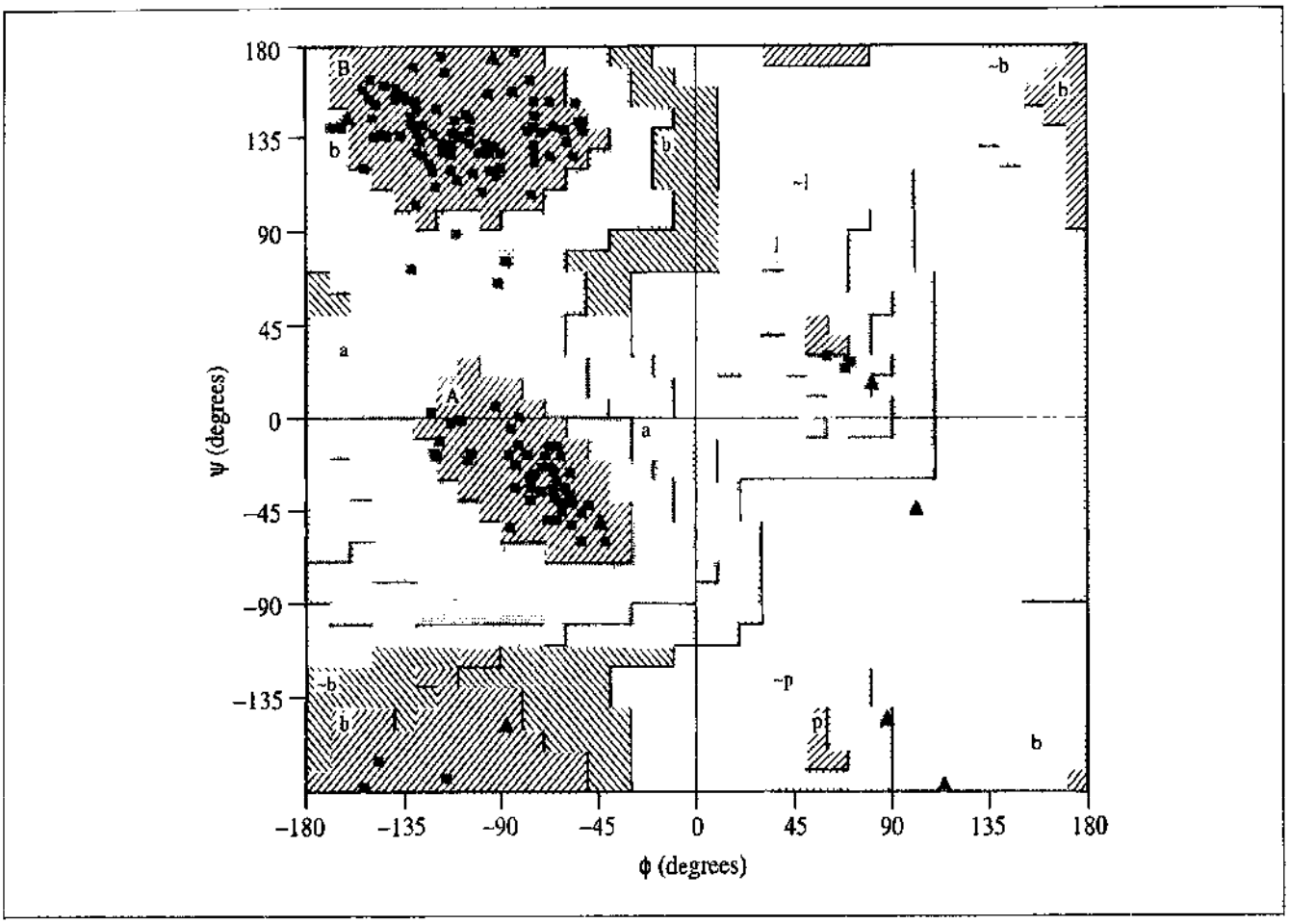


Fig. 9.3 Experimentally observed distribution of (ϕ ψ) angles in the enzyme dihydrofolate reductase The symbols are the actual values, the shaded areas correspond to the (ϕ ψ) distribution averaged over many protein structures

which will be discussed in more detail in Section 10.2. The amino acid glycine, which has no side chain (Figure 9.1), has a wider range of accessible conformations than the other amino acids, as can be seen from the Ramachandran map in Figure 9.2.

To perform a grid search of the conformational space of the alanine dipeptide, a series of conformations would be generated by systematically varying ϕ and ψ between 0° and 360°. This is equivalent to drawing a two-dimensional grid over the Ramachandran contour diagram in Figure 9.2; each grid point corresponds to a conformation generated by the grid search with some combination of ϕ and ψ. It can readily be seen that, even for a relatively large torsional increment, the number of conformations generated by the grid search is much larger than the number of minima on the surface; many of the initial conformations minimise to the same minimum energy structure. Moreover many of the initial conformations are very high in energy.

A major drawback of the grid search is that the number of structures to be generated and minimised increases in an exceptional fashion with the number of rotatable bonds, a phenomenon known as the *combinatorial explosion*. The number of structures generated is given by:

$$\text{Number of conformations} = \prod_{i=1}^{N} \frac{360}{\theta_i} \tag{9.1}$$

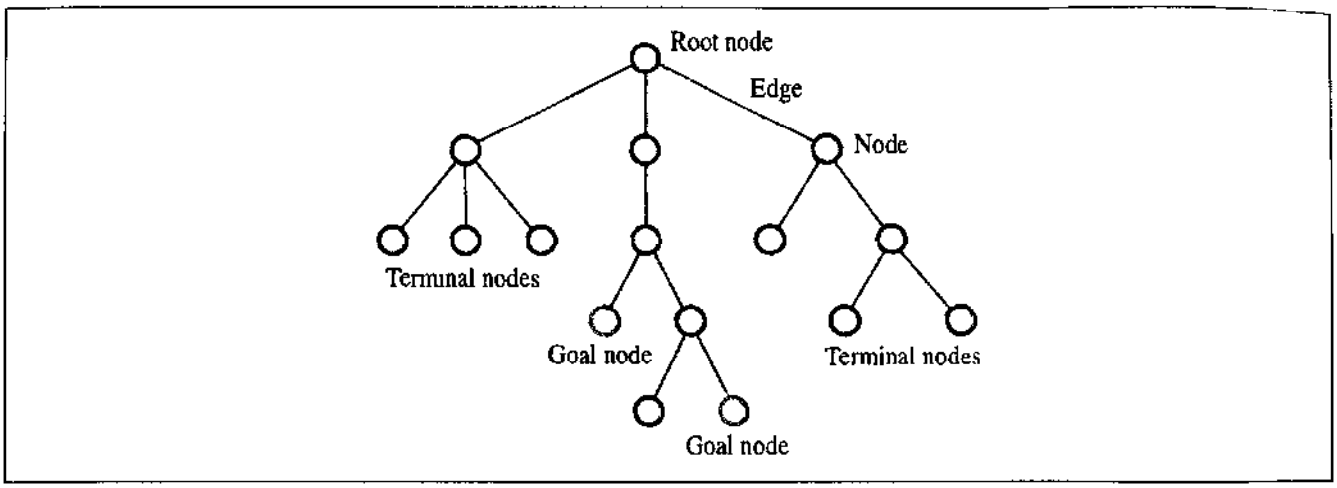


Fig 9 4 Schematic illustration of a search tree

where θ_i is the dihedral increment chosen for bond i. For example, if there are five bonds and an increment of 30° is used for each bond, then 248 832 structures will be generated. If the number of bonds is increased to seven, then the number of structures increases to almost 36 million. To put these figures into context, suppose each structure takes just one second to minimise. The five-bond problem will then require 69 hours to complete and the seven-bond problem will require 415 days. Despite this apparent limitation, systematic search algorithms are routinely employed to consider problems involving 10–15 bonds. This is achieved by eliminating from the time-consuming energy minimisation stage structures that have a very high energy or some other problem. A good way to understand these enhanced systematic search methods is to use a *tree representation* of the problem.

Search trees are widely used to represent the different states that a problem can adopt. An example is shown in Figure 9.4 from which it should be clear where the name derives, especially if the page is turned upside down. A tree contains *nodes* that are connected by *edges*. The presence of an edge indicates that the two nodes it connects are related in some way. Each node represents a state that the system may adopt. The *root node* represents the initial state of the system. *Terminal nodes* have no child nodes. A *goal node* is a special kind of terminal node that corresponds to an acceptable solution to the problem.

Suppose we wish to use a grid search to explore the conformational space of a simple alkane, *n*-hexane. We will assume that rotation of the terminal methyl groups can be ignored and so just three bonds can vary. If we permit each of the variable bonds to adopt just three values, corresponding to the *trans*, *gauche*(+) and *gauche*(−) conformations*, then the search tree for this problem contains 27 terminal nodes ($\equiv 3 \times 3 \times 3$) and is shown in Figure 9.5. The root node represents the starting point, where none of the rotatable bonds have been assigned a torsion angle. When the first rotatable bond is set to its first value (i.e. the *trans* conformation with a torsion angle of 180°), this corresponds to moving from the root node to the node numbered 1 in the tree. The second bond is now set to a *trans* conformation; this corresponds

* The *trans* conformation corresponds to a torsion angle of 180°, the *gauche*(+) conformation to one of +60° and the *gauche*(−) conformation to −60°. These approximately correspond to the torsion angles of the three minimum energy conformations of butane

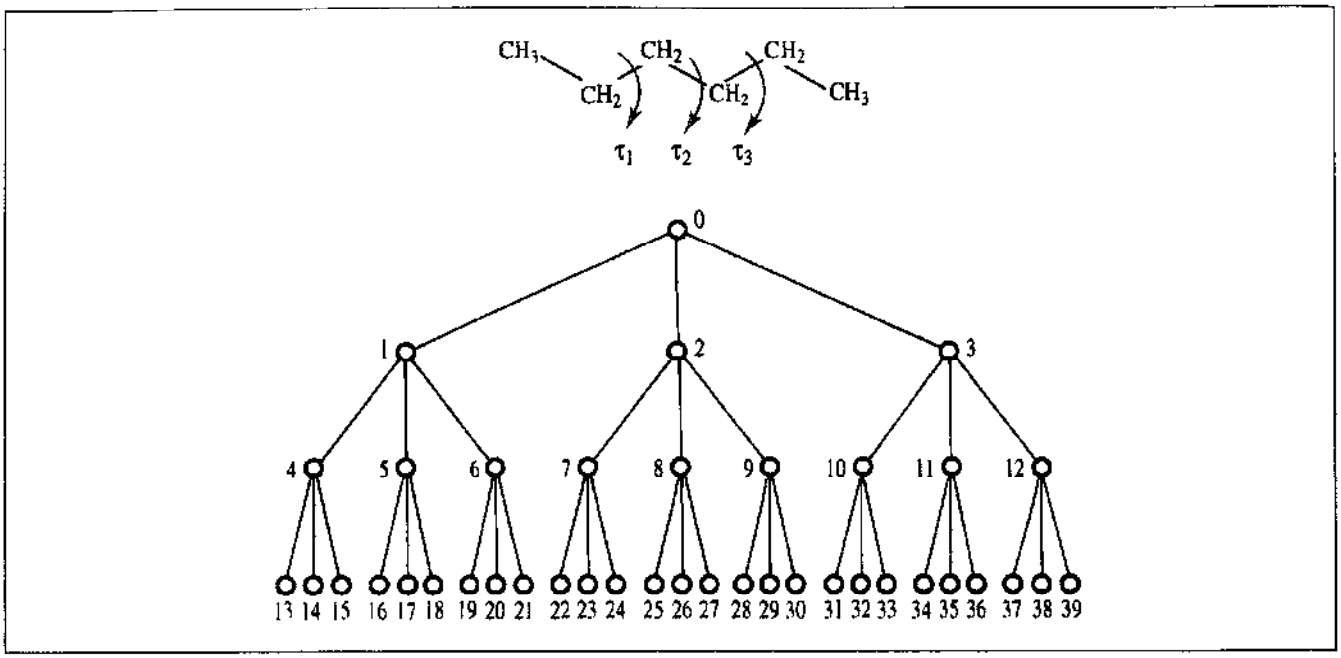


Fig. 9.5 Tree representation of the conformation search problem for hexane. Unlike the tree in Figure 9 4 the path length from the root node to any of the terminal nodes is constant

to a move to node 4. When the third bond is assigned *trans* we reach node 13, which is a terminal node and corresponds to a conformation ready for minimisation. To generate further conformations, it is necessary to change one of the three torsion angles. The most convenient way to do this is to assign a new value to the last bond to be set (i.e. bond 3). Setting bond 3 to a *gauche*(+) conformation is equivalent to moving back up the tree from node 13 to node 4 and then down to the terminal node 14. This gives a second completed conformation By proceeding in this fashion through the search tree (a process called *backtracking*) all conformations of the molecule can be generated. The search algorithm we have described is known as the *depth-first search.*

The efficiency of a depth-first search can be enhanced by discarding structures that violate some energetic or geometric criterion. Structures with high-energy steric interactions are

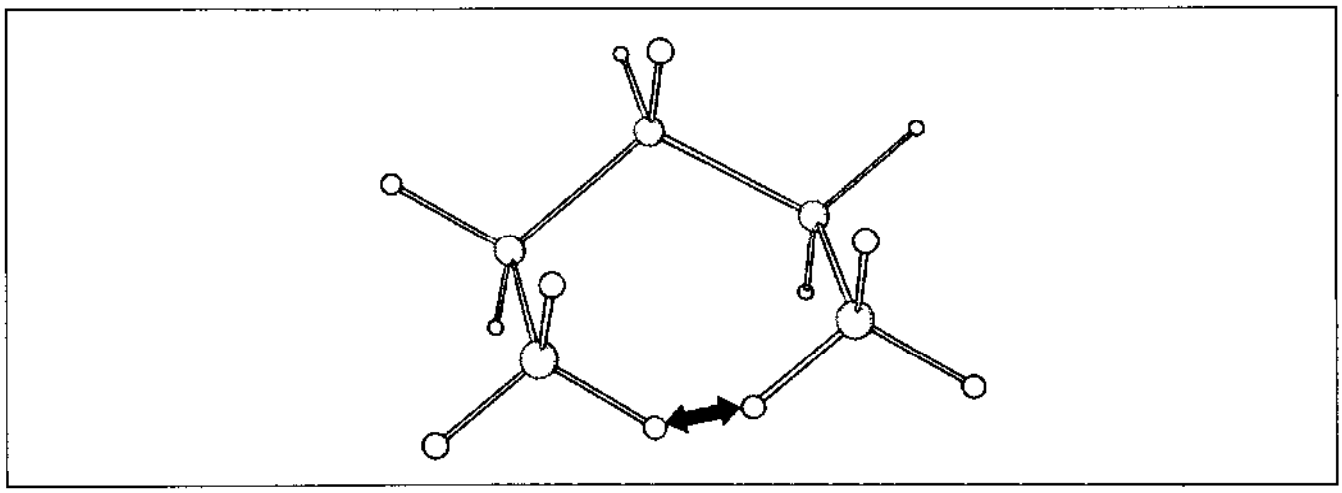


Fig. 9.6· A pentane violation arises when there are successive gauche(+) and gauche(−) torsion angles in an alkane chain.

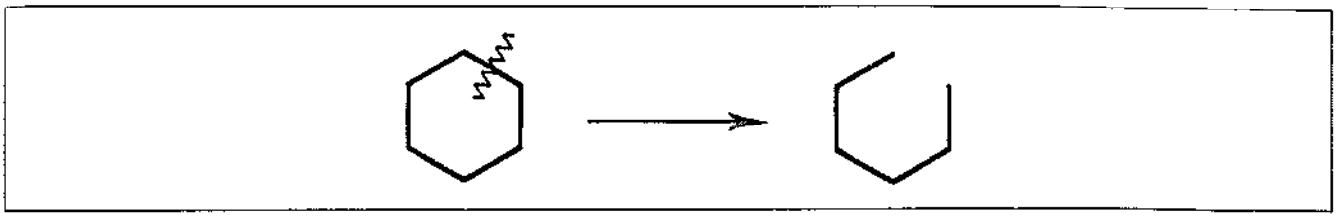


Fig. 9.7· A 'pseudo-acyclic' molecule is generated by breaking the ring.

then rejected before the energy-minimisation stage. We can further enhance the efficiency of the systematic search by checking partially constructed conformations before all the torsion angles have been assigned. Suppose we generate a partial structure containing two non-bonded atoms that are very close in space. In hexane, such a high-energy structure is generated if the first rotatable bond is set to a *gauche*(+) conformation and the second rotatable bond is assigned *gauche*(−) (a pentane violation, Figure 9.6). Whatever value is assigned to the third torsion angle, this high-energy steric problem will remain. All structures that lie below that node in the search tree (number 9 in Figure 9.5) can thus be eliminated, or *pruned*. It is important to stress that this is only possible if those parts of the molecule that are in violation will not be moved relative to each other by a subsequent torsional assignment.

Cyclic molecules are often quite difficult to analyse using a systematic search. The usual strategy is to break the ring, giving a 'pseudo-acyclic' molecule that can then be treated as a normal acyclic molecule. This process is illustrated for cyclohexane in Figure 9.7 When searching the conformational space of cyclic molecules additional checks must be included to ensure that the rings are properly formed. For example, an all-*trans* structure is a perfectly acceptable conformation of *n*-hexane, but is not an acceptable conformation of cyclohexane due to the unreasonable bond length between the ring-closure atoms. It is therefore common practice to check several intramolecular parameters when using a systematic search to explore the conformational space of a cyclic system; these parameters usually include the bond length between the ring-closure bonds, together with the bond angles at these atoms (Figure 9.8). In some programs other internal parameters (e.g. the torsion angles adjacent to the ring closure bond) are also checked. The main reason why rings are problematic

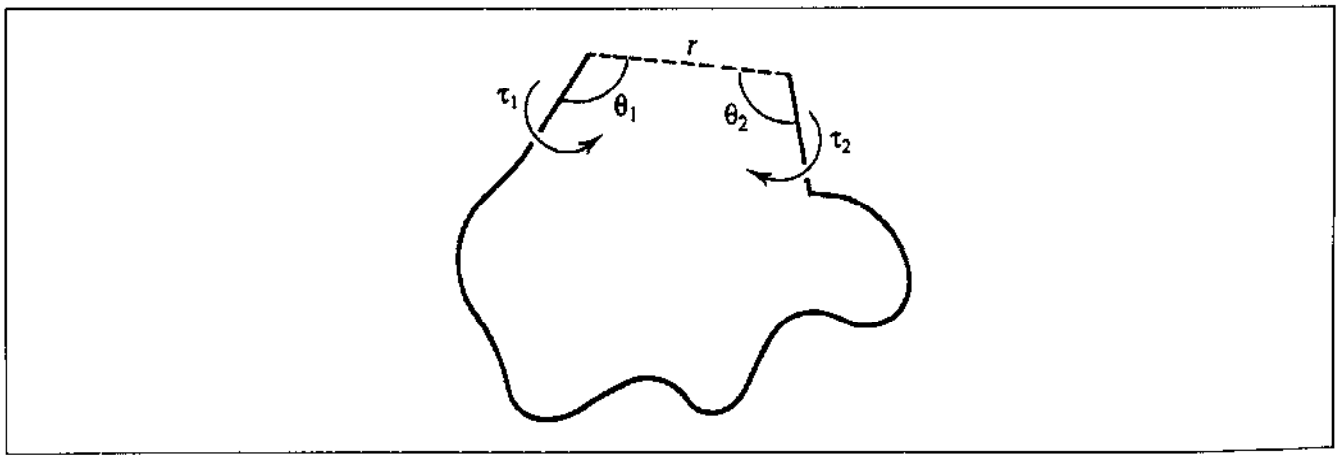


Fig 9 8 The intramolecular parameters that may be checked when exploring the conformational space of a ring system

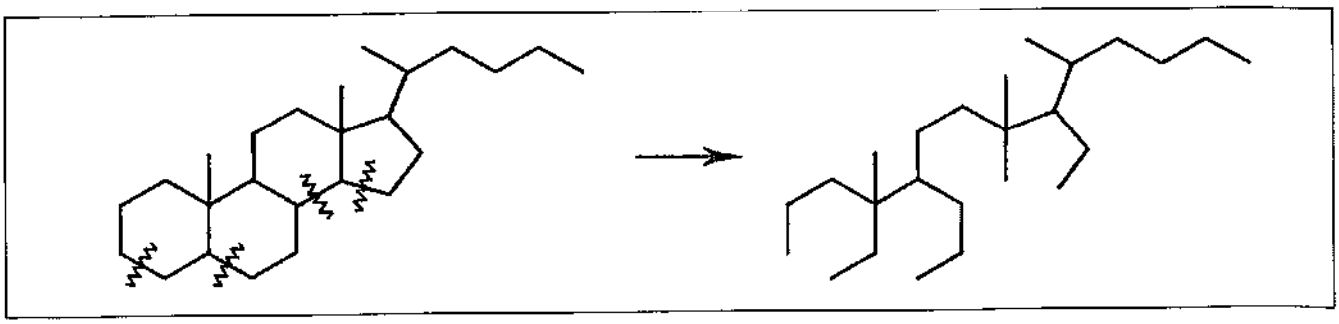


Fig 9 9 Creation of pseudo-acyclic molecule for a system with many rings.

for the systematic search is that these checks can often be applied only very late in the analysis; it is often necessary almost to complete the ring before the structure is rejected or accepted. One simple check that can be used when constructing cyclic molecules is to ensure that at all stages the distance of the growing chain from the start atom is short enough to enable the remaining bonds to close the ring.

The systematic search is most efficient when the rotatable bonds are processed in a unidirectional fashion. This ensures that once an atom has been fixed in space relative to the atoms already considered then it will not be moved again. For an acyclic molecule this means that the search starts at one end of the molecule and moves down the chain. Molecules containing rings are treated by opening the cycles to give the pseudo-acyclic molecule as described above and then processed (Figure 9.9).

Any systematic search ultimately requires a balance to be made between the resolution of the grid and the available computer resources Too fine a grid and the search may take too long; too coarse and important minima may be missed. The non-bonded criteria, which determine whether a structure is to be rejected, must also be assigned. The non-bonded criterion is often referred to as a 'bump check' and is usually set to a modest value (say, 2.0 Å) as the energy-minimisation step will be able to remove minor problems in the structure. For cyclic molecules the ring-closure criteria may also affect the results. It must also be remembered that the various cutoffs are interdependent, so that changing one may require others to be reassigned.

9.3 Model-building Approaches

One way to at least partially alleviate the combinatorial explosion that inevitably accompanies a systematic search is to use larger 'building blocks', or *molecular fragments*, to construct the conformations [Gibson and Scheraga 1987; Leach *et al.* 1988, 1990]. Fragment- or model-building approaches to conformational analysis construct conformations of a molecule by joining together three-dimensional structures of molecular fragments. This approach would be expected to be more efficient than the normal systematic search because there are usually many fewer combinations of fragment conformations than combinations of torsion angle values. This is particularly so for cyclic fragments, which are in any case problematic for the systematic search method. For example, the molecule in Figure 9.10 could be constructed by joining together the fragments indicated. Many molecular modelling systems offer a facility

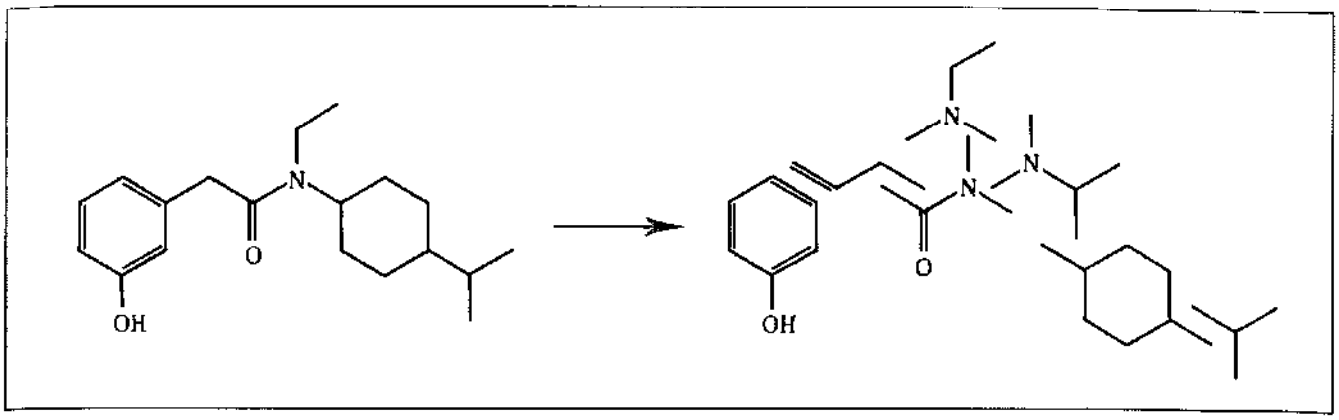


Fig 9 10 A conformation may be obtained by joining appropriate fragments

for constructing structures from molecular fragments, though the user usually has to specify manually which fragments are to be joined and how this is to be achieved. Clearly, if each fragment can adopt a number of conformations, then it is impractical to tackle the problem manually and some means of automating the method is required.

A program to explore conformational space automatically using the fragment-building approach must first decide which fragments are needed to construct the molecule [Leach *et al.* 1990]. This is done using a *substructure search algorithm*, which determines whether each of the fragments that the program 'knows about' is present in the molecule and how the atoms in the fragment match onto the atoms in the molecule. Having identified the fragments that are required, conformations can be generated. The conformations available to each fragment (often called templates) should span the range of conformations the fragment can adopt. For example, cyclohexane rings adopt the chair, twist-boat and boat conformations in molecules and so templates corresponding to these structures should be available. A conformation of the molecule is constructed by assigning a template to each fragment and then attempting to join the templates together. The search problem can be represented as a tree, as for a systematic search, and so all of the usual tree-searching algorithms are applicable. The search can be significantly enhanced by tree pruning.

The fragment-based approach to conformational analysis relies upon two assumptions The first assumption is that each fragment must be conformationally independent of the other fragments in the molecule. The second assumption is that the conformations stored for each fragment must cover the range of structures that are observed in fully constructed molecules. The fragment conformations can be obtained from a variety of sources, two common approaches are by analysing a structural database (see Section 9.11) or from some other conformational search method. A third limitation is that one can obviously only analyse molecules for which there are fragments available.

9.4 Random Search Methods

A random search is, in many ways, the antithesis of a systematic search A systematic search explores the energy surface of the molecule in a predictable fashion, whereas it is not

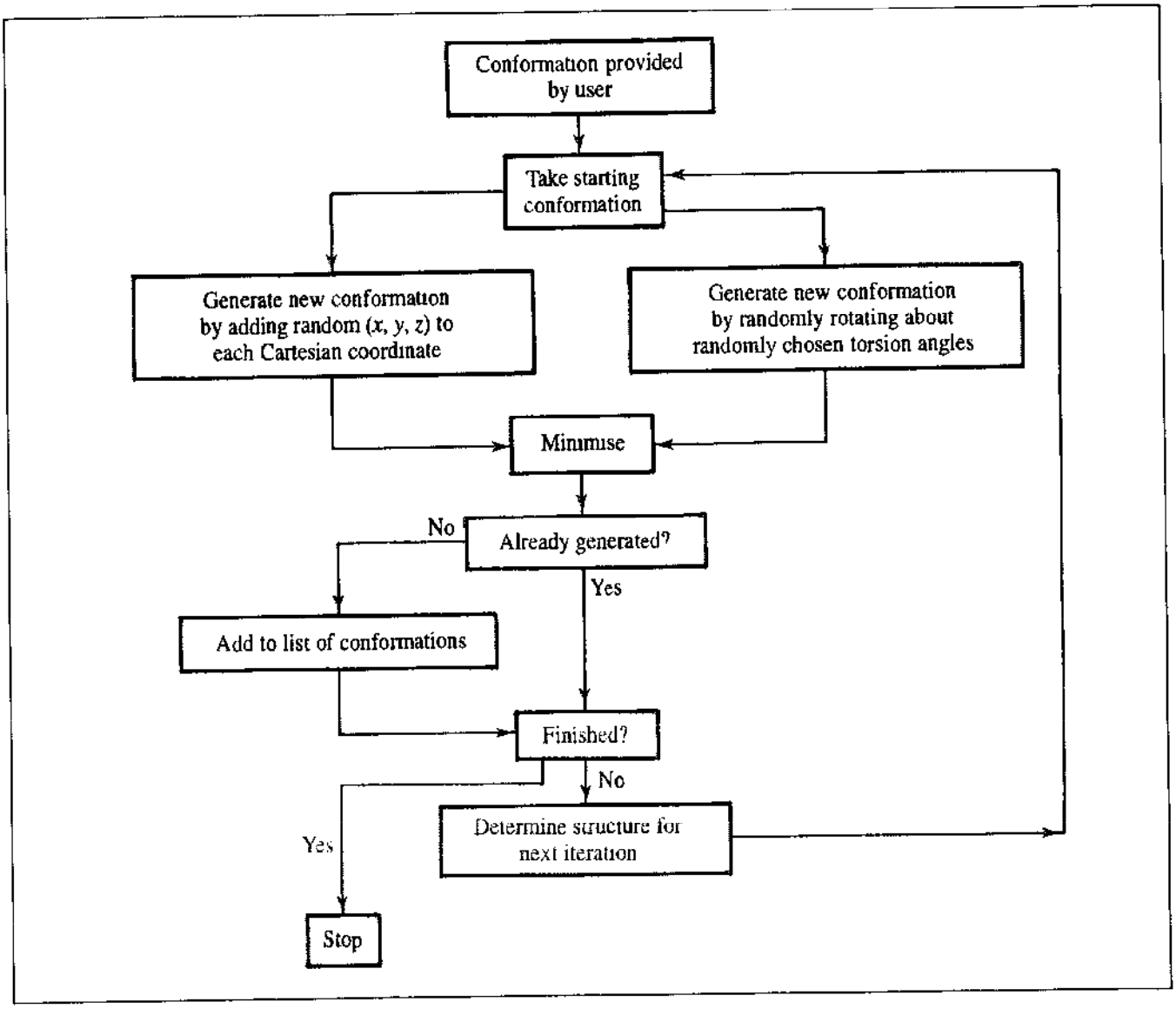


Fig. 9.11 Flow chart steps followed by a random conformational search

possible to predict the order in which conformations will be generated by a random method. A random search can move from one region of the energy surface to a completely unconnected region in a single step. A random search can explore conformational space by changing either the atomic Cartesian coordinates or the torsion angles of rotatable bonds. Both types of algorithm use a similar approach, which is outlined in flow chart form in Figure 9.11. At each iteration, a random change is made to the 'current' conformation. The new structure is then refined using energy minimisation. If the minimised conformation has not been found previously, it is stored. The conformation to be used as the starting point for the next iteration is then chosen and the cycle starts again. The procedure continues until a given number of iterations have been performed or until it is decided that no new conformations can be found.

The Cartesian and dihedral versions of the random search differ in the way in which each new structure is generated. The Cartesian method adds a random amount to the x, y and z coordinates of all the atoms in the molecule [Saunders 1987; Ferguson and Raber 1989]

whereas the dihedral method generates new conformations by making random changes to the rotatable bonds, with the bond lengths and bond angles being kept fixed [Li and Scheraga 1987; Chang *et al.* 1989]. The Cartesian method is extremely simple to implement but does rely heavily upon the minimisation step as the initial structures that are generated by the randomisation procedure can be very distorted and extremely high in energy; in some implementations the coordinates can change by 3 Å or more. The advantage of the dihedral search method is that many fewer degrees of freedom need to be considered. However, special procedures are required when applying the dihedral method to molecules that contain rings; typically these are broken in a manner simular to the systematic search described above to give a pseudo-acyclic molecule (Figure 9.9). After randomisation each ring is checked to ensure that any ring-closure constraints are satisfied. In the random dihedral method it is possible to change all dihedrals or just a randomly chosen subset of them.

There are many ways in which the structure for input to the next iteration of the search can be selected. A simple approach is to take the structure obtained from the previous step. An alternative is to select randomly a structure from those generated previously, weighting the choice towards those structures that have been selected the least (a *uniform usage* protocol). A third method is to use the lowest-energy structure found so far, or to bias the selection towards the lowest-energy structures. The Metropolis Monte Carlo scheme is often used to make the choice. Each newly generated structure (after energy minimisation) is accepted as the starting point for the next iteration if it is lower in energy than the previous structure or if the Boltzmann factor of the energy difference, $\exp[-(\mathscr{V}_{\text{new}}(\mathbf{r}^N) - \mathscr{V}_{\text{old}}(\mathbf{r}^N))/k_B T]$ is larger than a random number between 0 and 1. If not, the previous structure is retained for the next iteration. There is no fundamental reason why any of these methods should be preferred over another, but some are reported to be more efficient at exploring the conformational space or finding the global minimum energy conformation.

In a systematic search there is a defined endpoint to the procedure, which is reached when all possible combinations of bond rotations have been considered. In a random search, there is no natural endpoint; one can never be absolutely sure that all of the minimum energy conformations have been found. The usual strategy is to generate conformations until no new structures can be obtained. This usually requires each structure to be generated many times and so the random methods inevitably explore each region of the conformational space a large number of times.

9.5 Distance Geometry

One way to describe the conformation of a molecule other than by Cartesian or internal coordinates is in terms of the distances between all pairs of atoms. There are $N(N-1)/2$ interatomic distances in a molecule, which are most conveniently represented using an $N \times N$ symmetric matrix. In such a matrix, the elements (i, j) and (j, i) contain the distance between atoms i and j and the diagonal elements are all zero. Distance geometry explores conformational space by randomly generating many distance matrices, which are then converted into conformations in Cartesian space. The crucial feature about distance geometry (and the reason why it works) is that it is not possible to arbitrarily assign values to the

interatomic distances in a molecule and always obtain a low-energy conformation. Rather, the interatomic distances are closely interrelated and indeed many combinations of distances are geometrically impossible. This can be illustrated using a simple three-atom molecule (ABC). Simple trigonometry requires that the sum of the distances AB and AC must be greater than or equal to the distance BC. Thus, a conformation in which the distances are AB = 1.5 Å, AC = 1.4 Å and BC = 3.5 Å is not geometrically possible.

Distance geometry uses a four-stage process to derive a conformation of a molecule [Crippen 1981; Crippen and Havel 1988]. First, a matrix of upper and lower interatomic *distance bounds* is calculated. This matrix contains the maximum and minimum values permitted to each interatomic distance in the molecule. Values are then randomly assigned to each interatomic distance between its upper and lower bounds. In the third step, the distance matrix is converted into a trial set of Cartesian coordinates, which in the fourth step are then refined.

Some of the interatomic distance bounds can be determined from simple chemical principles. For example, X-ray crystallographic studies have shown that bond lengths are restricted to a small range of values that are determined primarily by the atomic number and hybridisation of the two atoms. The distance between two atoms which are both bonded to a third atom (i.e. are in 1,3 relationship) is also severely restricted and can be calculated from the angle at the central atom and the lengths of the two bonds. The distance between two atoms that are separated by three bonds (i.e. are in a 1,4 relationship) can vary with the torsion angle of the central bond, the minimum distance corresponding to a torsion angle of 0° and the maximum distance to a torsion angle of 180°. These three cases are shown in Figure 9.12. It is not so easy to determine limits on the other interatomic distances (i.e. between atoms in a 1, n relationship where $n > 4$) but it is usual to require that such atom pairs do not approach closer than the sum of the van der Waals radii of the two atoms. The upper bound is then usually assigned an arbitrarily large value.

A procedure called *triangle smoothing* is then used to refine the initial set of distance bounds. Triangle smoothing uses two simple trigonometrical restrictions on groups of three atoms, which are illustrated in Figure 9.13. The first restriction is that the distance between two atoms A and C can be no greater than the sum of the maximum values of the distances AB and BC. This can be written:

$$u_{AC} \leq u_{AB} + u_{BC} \tag{9.2}$$

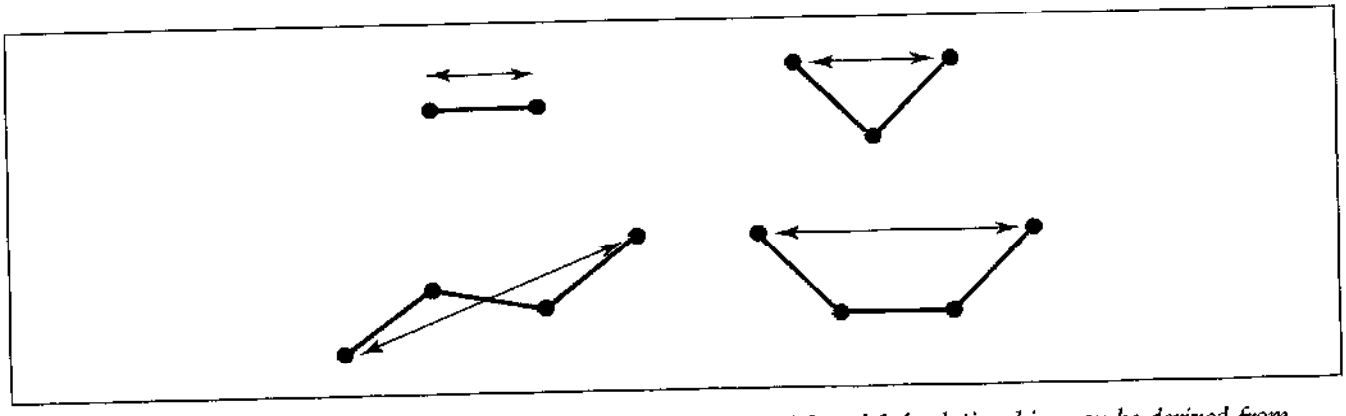


Fig. 9.12: The upper and lower distance bounds for atoms in 1,2, in 1,3 and 1,4 relationships can be derived from simple chemical principles

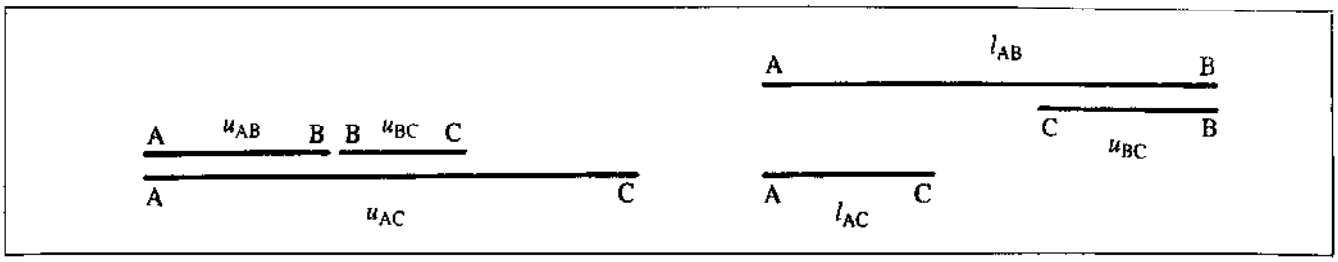


Fig 9 13. The two triangle inequalities used in distance geometry.

where u_{AB} indicates the upper bound on the AB distance. The second restriction is that the minimum value of the AC distance can be no less than the difference between the lower bound on AB and the upper bound on BC:

$$l_{AC} \geq l_{AB} - u_{BC} \tag{9.3}$$

where l_{AB} is used to indicate the lower bound distance. These two inequalities are repeatedly applied to the set of distances bounds until the entire set of distance bounds is self-consistent and all possible interatomic distance triplets satisfy both inequalities. Triangle smoothing need be performed only once for each molecule.

We can now proceed to the generation of conformations. First, random values are assigned to all the interatomic distances between the upper and lower bounds to give a trial distance matrix. This distance matrix is now subjected to a process called *embedding*, in which the 'distance space' representation of the conformation is converted to a set of atomic Cartesian coordinates by performing a series of matrix operations. We calculate the *metric matrix*, **G**, each of whose elements (i, j) is equal to the scalar product of the vectors from the origin to atoms i and j:

$$G_{ij} = \mathbf{i} \cdot \mathbf{j} \tag{9.4}$$

The elements G_{ij} can be calculated from the distance matrix using the cosine rule:

$$G_{ij} = (d_{io}^2 + d_{jo}^2 - d_{ij}^2)/2 \tag{9.5}$$

where d_{io} is the distance from the origin to atom i and d_{ij} is the distance between atoms i and j.

It is usual to take the centre of the molecule as the origin of the coordinate system The distance of each atom from the centre can be calculated directly from the interatomic distances using the following expression:

$$d_{io}^2 = \frac{1}{N}\sum_{j=1}^{N} d_{ij}^2 - \frac{1}{N^2}\sum_{j=2}^{N}\sum_{k=1}^{j-1} d_{jk}^2 \tag{9.6}$$

The metric matrix **G** is a square symmetric matrix. A general property of such matrices is that they can be decomposed as follows:

$$\mathbf{G} = \mathbf{V}\mathbf{L}^2\mathbf{V}^{\mathrm{T}} \tag{9 7}$$

The diagonal elements of $\mathbf{L}^2$ are the eigenvalues of **G** and the columns of **V** are its eigenvectors. The atomic coordinates can be derived from the metric matrix by rewriting

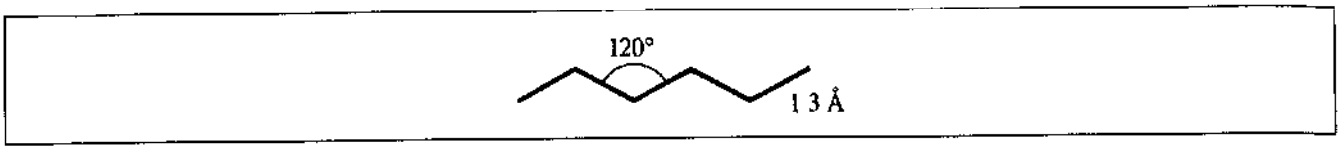


Fig. 9.14: Five-carbon fragment to illustrate distance geometry algorithm

Equation (9.4) as

$$\mathbf{G} = \mathbf{X}\mathbf{X}^{\mathrm{T}} \tag{9.8}$$

where **X** is a matrix containing the atomic coordinates. Equating Equations (9.7) and (9.8) gives:

$$\mathbf{X} = \mathbf{V}\mathbf{L} \tag{9.9}$$

As **L** has only diagonal entries, the matrix **L** is identical to its transpose: $\mathbf{L} = \mathbf{L}^{\mathrm{T}}$. The atomic coordinates are thus obtained by multiplying the square roots of the eigenvalues by the eigenvectors.

The triangle-smoothing and embedding steps of distance geometry are best understood using a specific example. Let us consider a five-atom, all-carbon fragment (Figure 9.14), in which all of the carbon–carbon bonds are assumed to have an optimal length of 1.3 Å and all internal angles are 120°. If we further assume that the carbon van der Waals radius is 1.4 Å, then the initial bounds matrix is as follows.

$$\begin{pmatrix} 0.0 & 1.3 & 2\,2517 & 3\,4395 & 99.0 \\ 1.3 & 0\,0 & 1.3 & 2.2517 & 3.4395 \\ 2.2517 & 1.3 & 0\,0 & 1.3 & 2.2517 \\ 2\,6 & 2.2517 & 1\,3 & 0.0 & 1.3 \\ 2.8 & 2.6 & 2.2517 & 1.3 & 0.0 \end{pmatrix} \tag{9.10}$$

Note that the lower bound for the distance between atoms 1 and 5 equals the sum of their van der Waals radii and that the upper bound has been set to an arbitrarily large value of 99 Å. All other distances have been allocated on the basis of geometric arguments. In a 'real' example the upper and lower bounds for bonded atoms would usually be slightly different (by approximately 0.1 Å) to reflect the fact that bond lengths in real molecules do vary a little. The 1,3 bounds would also be made slightly different. After triangle smoothing only one distance bound is changed, the upper bound of the distance between atoms 1 and 5. This distance is changed to a value that is equal to the sum of the upper bounds of the distances between atoms 1 and 3 and 3 and 5. The smoothed bounds matrix that results is:

$$\begin{pmatrix} 0.0 & 1.3 & 2.2517 & 3.4395 & 4.5033 \\ 1.3 & 0.0 & 1.3 & 2\,2517 & 3.4395 \\ 2.2517 & 1.3 & 0.0 & 1.3 & 2.2517 \\ 2.6 & 2.2517 & 1.3 & 0.0 & 1.3 \\ 2.8 & 2.6 & 2.2517 & 1.3 & 0.0 \end{pmatrix} \tag{9.11}$$

Suppose interatomic distances are now randomly assigned between the lower and upper bounds to give the following distance matrix:

$$\begin{pmatrix} 0.0 & 1.3 & 2.25 & 3.11 & 3.42 \\ & 0.0 & 1.3 & 2.25 & 2.85 \\ & & 0.0 & 1.3 & 2.25 \\ & & & 0.0 & 1\,3 \\ & & & & 0.0 \end{pmatrix} \tag{9.12}$$

The corresponding metric matrix is:

$$\begin{pmatrix} 3.571 & 1.569 & -0.427 & -2.276 & -2.436 \\ 1.569 & 1.256 & 0.105 & -1.122 & -1.808 \\ -0.427 & 0.105 & 0\,644 & 0.261 & -0.583 \\ -2.276 & -1.122 & 0.261 & 1.569 & 1.569 \\ -2.436 & -1.808 & -0.583 & 1.569 & 3.259 \end{pmatrix} \tag{9.13}$$

The eigenvalues of this matrix are 8.18, 1.74, 0.26, 0.10 and 0.0, with the matrix of eigenvectors being:

$$\mathbf{W} = \begin{pmatrix} 0.621 & 0.455 & -0.425 & 0.164 \\ 0.355 & -0.184 & 0.800 & 0.020 \\ 0.0 & -0.573 & -0.368 & -0\,580 \\ -0.408 & -0.287 & -0.153 & 0\,727 \\ -0.567 & 0.590 & 0.145 & -0\,330 \end{pmatrix} \tag{9.14}$$

The 'best' three-dimensional structure is obtained by taking the eigenvectors that correspond to the three largest eigenvalues, providing they are all positive. If these eigenvalues are λ_1, λ_2 and λ_3 and **W** is the matrix containing the associated eigenvectors, then the Cartesian coordinates (x_i, y_i, z_i) of each atom i are calculated as follows:

$$x_i = \sqrt{\lambda_1 W_{i1}} \tag{9 15}$$

$$y_i = \sqrt{\lambda_2 W_{i2}} \tag{9.16}$$

$$z_i = \sqrt{\lambda_3 W_{i3}} \tag{9.17}$$

For our five-carbon example, the coordinates obtained using the three highest eigenvalues are:

Atom	*x* coordinate	*y* coordinate	*z* coordinate
1	1.777	0.601	−0 218
2	1.014	−0.244	0.410
3	−0.001	−0.757	−0.188
4	−1.166	0 379	−0.079
5	−1.623	0.799	0.075

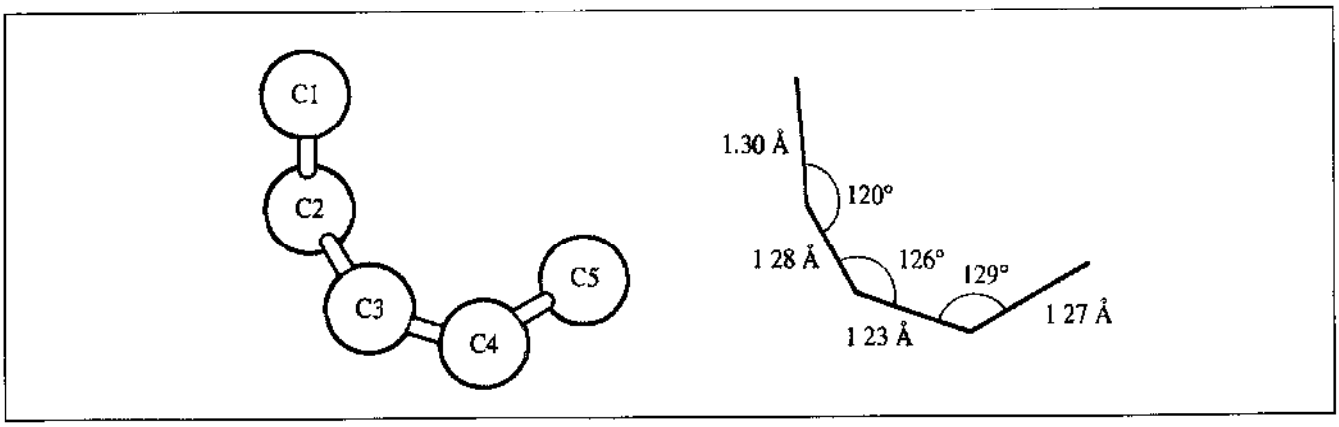


Fig 9 15: Conformation of the five-carbon fragment generated by distance geometry

This conformation is illustrated schematically in Figure 9.15. The interatomic distance matrix for this conformation is:

$$\begin{pmatrix} 0.0 & 1.299 & 2.24 & 3\,10 & 3.42 \\ & 0.0 & 1.29 & 2.24 & 2.85 \\ & & 0.0 & 1.23 & 2.25 \\ & & & 0.0 & 1.25 \\ & & & & 0\,0 \end{pmatrix} \tag{9.18}$$

Note that the distances in this conformation do not equal the original randomly chosen distances, nor do they all lie between the upper and lower bound values in the bounds matrix. For example, the distance between atoms 4 and 5 is 1 25 Å rather than 1.3 Å. This is because it may be necessary to use more than just three dimensions to find a conformation that satisfies the distances in the initial distance matrix. The number of non-zero eigenvectors of the metric matrix equals the dimensionality of the space in which a solution can be found. In general, if there are N interatomic distances then a solution can be found in $N - 1$ dimensions. This is in part a consequence of the fact that three-dimensional objects must not only satisfy triangle inequalities but also tetrangle, pentangle and hexangle relationships. Moreover, triangle smoothing is often only applied to the bounds matrix; the distance matrix that is used as input to the embedding stage may, in fact, contain combinations of distances that violate the triangle inequalities. Improved sampling of conformational space can be achieved if the trial distances are selected so that they do satisfy the triangle inequalities (a process known as *metrisation*), but for reasons of computational cost it is often not used in its full form.

If we add the coordinates corresponding to the fourth eigenvalue, then the original distance matrix is reproduced exactly. These fourth-dimensional coordinates are as follows:

Atom	4th coordinate
1	0.053
2	0.006
3	−0.188
4	0.235
5	−0.107

The distance between atoms 4 and 5 in this four-dimensional space is exactly 1.3 Å.

In the final step of the distance geometry algorithm the coordinates are refined so that the conformation better satisfies the initial distance bounds. A conjugate gradients minimisation algorithm is often employed for this step. The function to be minimised has a positive value for distances that are outside the permitted range but is zero otherwise. The penalty functions most commonly used are:

$$E = \sum_i \sum_{j>i} \begin{cases} (d_{ij}^2 - u_{ij}^2)^2 & d_{ij} > u_{ij} \\ 0 & l_{ij} \leq d_{ij} \leq u_{ij} \\ (l_{ij}^2 - d_{ij}^2)^2 & d_{ij} < l_{ij} \end{cases} \tag{9.19}$$

$$E = \sum_i \sum_{j>i} \begin{cases} [(d_{ij}^2 - u_{ij}^2)/u_{ij}^2]^2 & d_{ij} > u_{ij} \\ 0 & l_{ij} \leq d_{ij} \leq u_{ij} \\ [(l_{ij}^2 - d_{ij}^2)/d_{ij}^2]^2 & d_{ij} < l_{ij} \end{cases} \tag{9.20}$$

where u_{ij} is the upper bound distance between atoms i and j and l_{ij} is the lower bound distance. The first function weights long distances more than short distances whereas the second error function weights all distances equally. Both functions are zero when all the distances are between the upper and lower bounds. A conformation in which all distance bounds are satisfied is not necessarily at an energy minimum, and so the final structure may subsequently be subjected to force field energy minimisation to derive the associated minimum energy structure.

During the optimisation of the structure against the distance constraints it is usual to incorporate *chiral constraints*. These are used to ensure that the final conformation is the desired stereoisomer. Chiral constraints are necessary because the interatomic distances in two enantiomeric conformations are identical and as a consequence the 'wrong' isomer may quite legitimately be generated. Chiral constraints are usually incorporated into the error function as a chiral volume, calculated as a scalar triple product. For example, to maintain the correct stereochemistry about the tetrahedral atom number 4 in Figure 9.16, the following scalar triple product must be positive:

$$(\mathbf{v}_1 - \mathbf{v}_4) \cdot [(\mathbf{v}_2 - \mathbf{v}_4) \times (\mathbf{v}_3 - \mathbf{v}_4)] \tag{9.21}$$

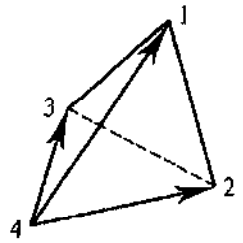


Fig 9.16: The stereochemistry about tetrahedral atoms can be maintained with an appropriate chiral constraint

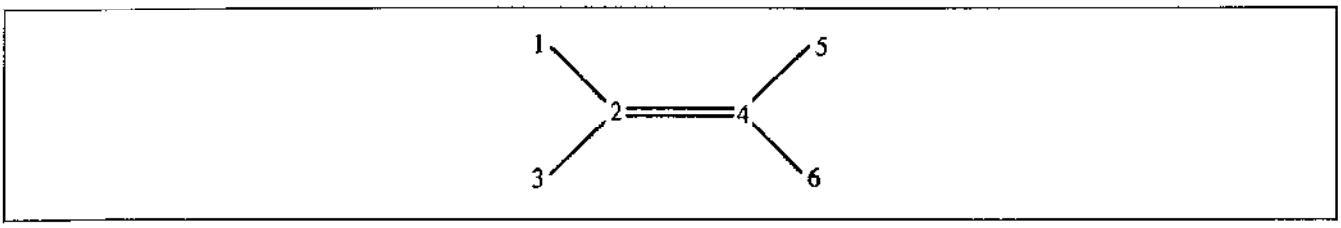


Fig. 9 17 A double bond can be forced to adopt a planar conformation through the use of appropriate chiral constraints.

The other stereoisomer corresponds to a negative chiral volume. Chiral constraints are included in the penalty function by adding terms of the following form:

$$(V_{ch} - V^*_{ch})^2 \tag{9.22}$$

where V^*_{ch} is the desired value of the chiral constraint. Chiral constraints can also be used to force groups of atoms to lie in the same plane by requiring the chiral volume to have a value of zero. More than one such constraint may be required for each planar group. For example, to force all six atoms about the double bond in Figure 9.17 to lie in the same plane, the three sets of chiral volumes defined by the atoms 1, 2, 3, 4; 2, 4, 5, 6; 1, 2, 4, 5 must be zero. A commonly used strategy in many distance geometry programs is to perform the first few steps of refinement using a conformation defined in four dimensions, as this can help to invert any incorrect chiral centres. The minimisation then swiches to three-dimensional conformations for the final stages of the distance geometry refinement. A force field energy minimisation may also be used. When the conformation has been refined, the next structure is generated, starting with the assignment of random distances.

Many enhancements have been made to the basic distance geometry method. Some of the most useful enhancements result from the incorporation of chemical information. For example, if the lower bound for the 1,4 distances is set to a value equivalent to a torsion angle of 60° rather than one of 0° then eclipsed conformations can be avoided. Similarly, amide bonds can be forced to adopt a nearly planar structure by an appropriate choice of distance bounds and chiral constraints.

9.5.1 The Use of Distance Geometry in NMR

One of the most important uses of distance geometry is for deriving conformations that are consistent with experimental distance information, especially distances obtained from NMR experiments. The NMR spectroscopist has at his or her disposal a range of experiments that can provide a wealth of information about the conformation of a molecule. Two of the most commonly used NMR experiments that provide such conformationally dependent information are the 2D-NOESY (nuclear Overhauser enhancement spectroscopy) and the 2D-COSY (correlated spectroscopy) experiments [Derome 1987]. NOESY provides information about the distances between atoms which are close together in space but may be separated by many bonds. The strength of the NOESY signal is inversely proportional to the sixth power of the distance and so by analysing the nuclear Overhauser spectrum it is possible

to calculate approximate values for the distance between relevant pairs of atoms. COSY experiments are often used to provide information about atoms which are covalently separated by three bonds (i.e. torsion angles). Both types of experiment provide information about interatomic distances, and so distance geometry is a natural technique to use for generating conformations that are consistent with the experimental data Distance geometry has been particularly useful for solving the structures of proteins and nucleic acids, where the amount of data is so large that it is impossible to perform the task manually. The distance information provided by NMR experiments does of course supplement the geometrical constraints on the interatomic distances that are derived from the internal geometry (i.e. bond lengths and angles).

Distance geometry is, at heart, a random technique. It is therefore usual to generate more than one conformation in order to try to explore the conformational space that is consistent with the experimentally derived distances. The resulting set of structures is often displayed as a superimposed set; this enables the similarities and differences between the structures to be easily identified. For example, in Figure 9.18 (colour plate section) we show an ensemble of conformations of RANTES, a small protein called a chemokine that is implicated in inflammation [Chung *et al.* 1995]. It is often found that some parts of the molecule adopt very similar conformations in all the structures whereas other regions show considerable variation. This is often interpreted as an indication of conformational flexibility, but it is important to remember that it may also indicate a lack of experimental data for those atoms.

9.6 Exploring Conformational Space Using Simulation Methods

The Monte Carlo and molecular dynamics simulation methods can be used to explore the conformational space of molecules. During such a simulation the system is able to

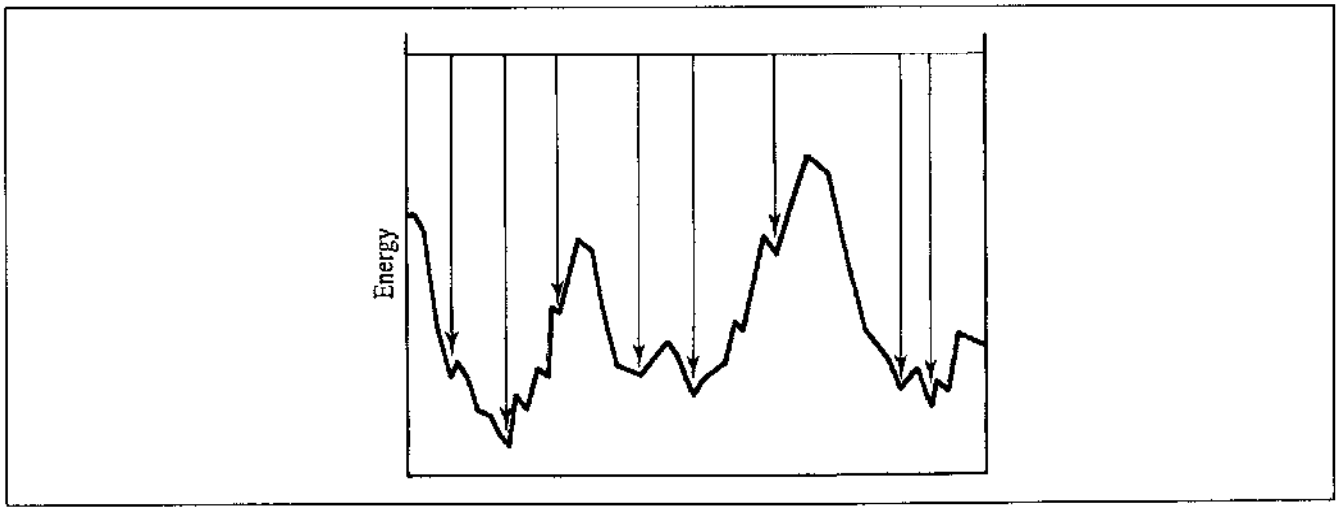


Fig. 9.19 Schematic illustration of an energy surface. A high-temperature molecular dynamics simulation may be able to overcome very high energy barriers and so explore conformational space On minimisation, the appropriate minimum energy conformation is obtained (arrows)

overcome energy barriers and so explore different regions of the conformational space. A distinction must be made between a 'true' Monte Carlo simulation (Chapter 8) and the minimisation-based random search methods discussed in Section 9.4. A true Monte Carlo simulation does not include any energy minimisation, and each randomly generated conformation would be accepted or rejected using the Metropolis criterion. There are difficulties in applying the Monte Carlo technique in simulations of flexible molecules, as we have discussed in Chapter 8.

Molecular dynamics is widely used for exploring conformational space. A common strategy is to perform the simulation at a very high, physically unrealistic temperature. The additional kinetic energy enhances the ability of the system to explore the energy surface and can prevent the molecule getting stuck in a localised region of conformational space This is schematically illustrated in Figure 9.19. Structures are then selected at regular intervals from the trajectory for subsequent energy minimisation.

9.7 Which Conformational Search Method Should I Use? A Comparison of Different Approaches

With such an array of methods for exploring conformational space, it can be difficult to decide which to choose. Each method has its own strengths and weaknesses. Systematic searches are subject to the effects of combinatorial explosion, and they are not naturally suited to molecules with rings However, they do have a definite endpoint; when the search has finished, one can be guaranteed to have found all conformations for a given dihedral increment. Random search methods can require long runs to ensure that the conformational space has been covered, and they can generate the same structure many times. Distance geometry is particularly useful when experimental information can be incorporated, as is restrained molecular dynamics.

A comparison of various methods for searching conformational space has been performed for cycloheptadecane ($C_{17}H_{34}$) [Saunders *et al.* 1990]. The methods compared were the systematic search, random search (both Cartesian and torsional), distance geometry and molecular dynamics. The number of unique minimum energy conformations found with each method within 3 kcal/mol of the global minimum after 30 days of computer processing were determined (the study was performed in 1990 on what would now be considered a very slow computer). The results are shown in Table 9.1.

Method	Total unique conformers found after 30 days processing
Systematic search	211
Random Cartesian search	222
Random dihedral search	249
Distance geometry	176
Molecular dynamics	169

Table 9 1: A comparison of five different conformational searching algorithms. (Data from [Saunders et al *1990])*

Combining the results from all the different methods revealed a grand total of 262 conformations within 3 kcal/mol of the global energy minimum. No one method found all of them but the random dihedral search did give the best performance in this case. The global energy minimum would be expected to constitute only about 8% of the total population of conformational states for this molecule if it is assumed that the entropies of all the conformations are the same. Largely as a result of this study cycloheptadecane is now often considered as the proving ground for new conformational search methods (despite the fact that it is not a very good representative for a typical 'organic' molecule).

9.8 Variations on the Standard Methods

Each year usually sees the publication of a handful of new methods for exploring conformational space. Many can be considered as variants on one of the approaches discussed thus far but which may provide some advantage in terms of the efficiency and effectiveness with which they explore conformational space. Some of these alternative methods are designed for quite specific types of molecule (such as ring systems) and as may not be particularly 'general' approaches to conformational analysis. Here we describe in more detail two of these newer methods, one which extends the systematic search and one which uses an alternative approach to generating the initial structure prior to energy minimisation.

9.8.1 The Systematic Unbounded Multiple Minimum Method

One of the possible limitations of a regular systematic conformational search is that successive conformations are often very similar to each other. In a typical depth-first search strategy successive conformations often differ by just one or two torsion angles. An additional potential limitation is that the torsional increment is usually specified at the start of the search, so that if one wants to perform a search at a higher resolution one typically has to rediscover all the conformations generated at the lower resolution. The systematic unbounded multiple minimum (SUMM) method [Goodman and Still 1991] is designed to address these two concerns. SUMM generates conformations by first selecting a structure from those generated previously (using the uniform usage protocol) and changing one or more of its torsion angles. This gives a new structure, which is then energy minimised, checked to determine whether it has already been generated, and if not it is added to the list of conformational minima. Central to the approach is that the changes in torsion angles are determined in a preordained, systematic manner. This is achieved by setting up a sequence of torsional modifications such that the first components in the sequence correspond to changes of a single torsion angle through 120°. Later components correspond to changes of two torsion angles through 120°, and so on. For a given number of torsion angle changes and a given torsional increment the actual changes are mixed up so that successive components represent modifications to very different parts of the molecule For example, the first two modifications in a normal systematic search of hexane could correspond to changes of 0°, 0°, 120° and 0°, 0°, 240° for torsions τ_1, τ_2 and

τ_3, respectively, whereas in SUMM the second modification might correspond to a rather different sequence (e.g. 0°, 240°, 0°). If one were to run the search to completion then all possible torsional variations would be considered both with and without this mixing protocol. However, if the search was terminated after a fixed number of steps then the mixing protocol increases the chances of achieving greater coverage of conformational space. The algorithm maintains a record of the torsional modifications that have been made to each distinct minimum energy conformation so, when that structure is next selected to act as the starting point for a conformational change, it knows which torsional changes should be performed.

The SUMM approach can be applied to both cyclic and acyclic molecules, though for cyclic systems it is still necessary to check for ring-closure violations. When rings are present then SUMM can use a preoptimisation procedure to reduce the length of the often abnormally long ring-closure bond. If a structure with such a long bond is subjected to normal energy minimisation then the bond length will be rapidly corrected but this often leads to significant distortions to the rest of the molecule, with torsion angles changing significantly from their initial values. This could be construed as undermining the rationale of a systematic search. The preoptimisation procedure makes small sequential changes to those torsion angles that affect each of the ring-closure bonds in an iterative fashion, in order to gradually bring the ring-closure bond(s) closer to their ideal value. SUMM is considered to be particularly efficient for locating all low energy conformations of a molecule. A random search method spends more and more time generating structures that have already been identified earlier in the search, in contrast to systematic methods such as SUMM

9.8.2 Low-mode Search

The low-mode search method [Kolossvary and Guida 1996] is closely related to the methods described in Section 5.9 for locating saddle points on energy surfaces. As we discussed there, one way to locate a transition state is to follow the 'path of shallowest ascent' from a minimum. In favourable cases, this path will largely correspond to one of the low-frequency normal modes of vibration. By continuing to move along the path one might expect to locate the second minimum that is connected via the saddle point with the starting structure. Locating saddle points can be a very difficult and time-consuming process and so some modifications are required to make such an approach practical for conformational searching. In the low-mode search an initial minimum energy conformation is subjected to normal mode analysis. Those low-frequency modes which are below a user-specified frequency threshold (e.g. 250 cm^{-1}) are identified and searched by changing the atomic coordinates in a manner given by the relevant eigenvector. This perturbation of the initial structure is performed in discrete steps until either the energy of the structure increases beyond a specified threshold or until the energy first increases but then starts to fall. This latter case may correspond to a movement over the saddle point and into the locality of a nearby minimum. In such cases (which are relatively rare), the structure is then fully energy minimised to give a new minimum energy conformation.

A particular advantage of the low-mode search is that it can be applied to both cyclic and acyclic molecules without any need for special ring closure treatments. As the low-mode search proceeds a series of conformations is generated which themselves can act as starting points for normal mode analysis and deformation. In a sense, the approach is a systematic one, bounded by the number of low-frequency modes that are selected. An extension of the technique involves searching random mixtures of the low-frequency eigenvectors using a Monte Carlo procedure.

9.9 Finding the Global Energy Minimum: Evolutionary Algorithms and Simulated Annealing

Evolutionary algorithms and simulated annealing are two methods that have found widespread use in molecular modelling. Their use is by no means restricted to the problem of finding the global minimum energy conformation of a molecule, but they have been applied to problems as diverse as protein–ligand docking, molecular design, QSAR and pharmacophore mapping [Clark and Westhead 1996; Jones 1998; Judson 1997]. We will consider some of these alternative applications in Chapter 12. Nevertheless, conformational analysis is a very good problem with which to introduce and describe these two methods.

9.9.1 Genetic and Evolutionary Algorithms

Evolutionary algorithms (EA) are a group of methods based on ideas of biological evolution that are designed to find optimal solutions to problems. There are currently three basic classes of evolutionary algorithm: genetic algorithms (GA), evolutionary programming (EP) and evolution strategies (ES). There are many similarities between these three classes but some key differences. Common to all three is the idea of creating a 'population' of possible solutions to the problem. The members of the population are scored using a 'fitness function' that measures how 'good' they are. The population changes over time and (hopefully) evolves towards better solutions. This process of generating new solutions is often referred to as 'breeding', with the new solutions being the 'children' that are generated from the 'parents' of the previous 'generation'. We will give an outline of these methods using the problem of finding the global minimum energy conformation as an example.

Probably the best-known of the three classes is the genetic algorithm [Goldberg 1989]. The following is a description of the basic method (or *canonical* genetic algorithm). The first step is to create a population of μ possible solutions. In conformational analysis, this initial population would correspond to a set of randomly generated conformations of the molecule. Each member of the population is coded by a 'chromosome'. This is usually stored as a linear string of bits (i.e. 0s and 1s). The chromosome codes for the values of the torsion angles of the rotatable bonds in the molecule, as illustrated in Figure 9.20. The initial population is most easily obtained by randomly setting bits to 0 or 1 in the chromosomes. After decoding each chromosome and assigning the torsion angles to the

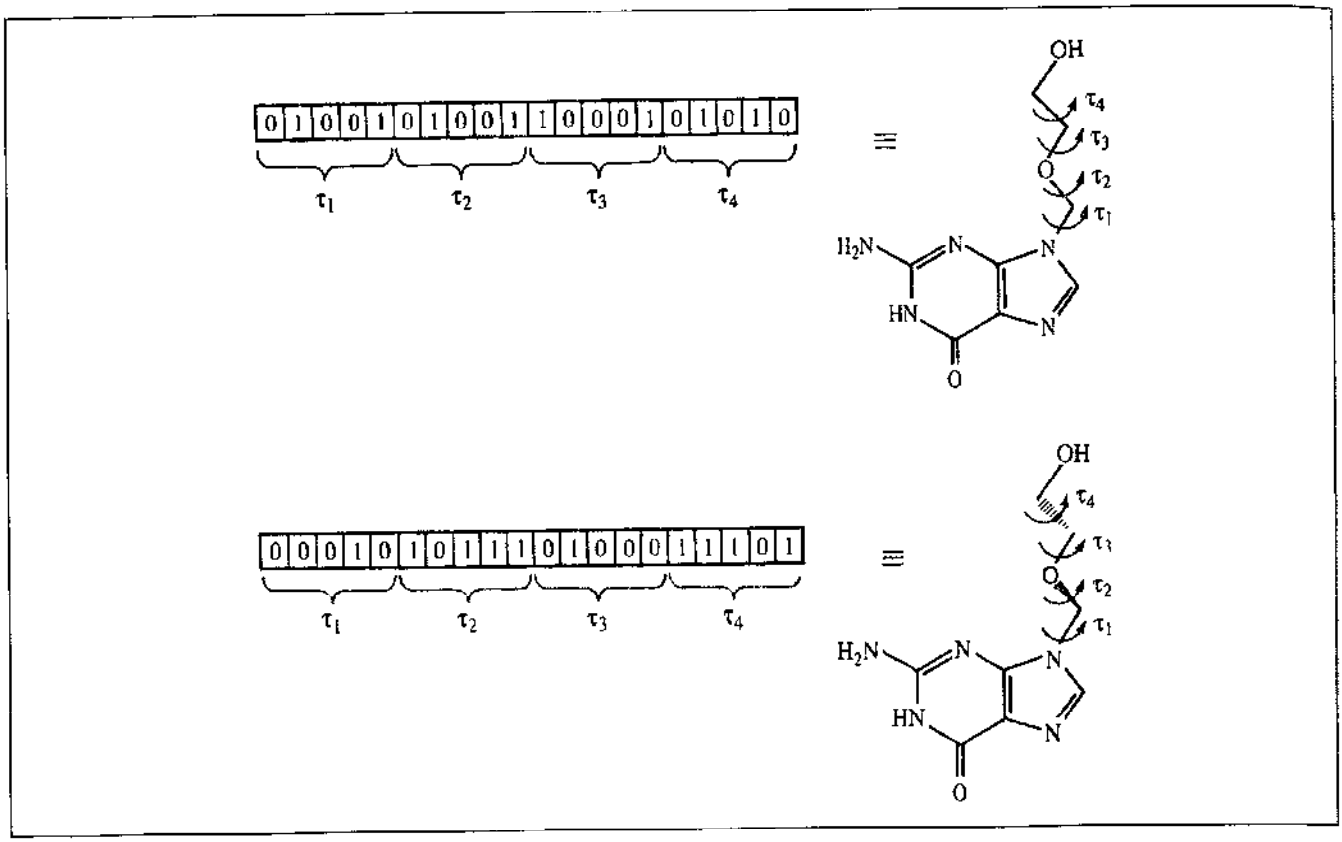


Fig 9.20 The chromosome in a genetic algorithm codes for the torsion angles of the rotatable bonds

appropriate values in the molecule, the fitness of each member of the population can be calculated. In conformational analysis, an appropriate fitness function would be the internal energy, as might be calculated using molecular mechanics. A new population is then generated. In the canonical genetic algorithm, $\mu/2$ pairs of parents are selected from the current population. These pairs are chosen at random, but with a bias towards the most fit individuals. A technique called *roulette wheel selection* is often used to achieve this bias by using slot sizes in the roulette wheel that are proportional to the values of the fitness function. A simple example is shown in Figure 9.21. The use of roulette wheel selection means that particularly fit members of the population may be able to produce many offspring. The new population is then subjected to genetic operators, the two most commonly used of which are *crossover* (or *recombination*) and *mutation*. In crossover, a cross

Object	Fitness	Roulette fraction
A	3	1/4
B	6	1/2
C	2	1/6
D	1	1/12

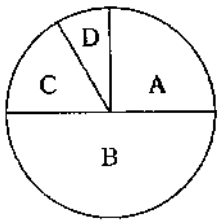


Fig 9 21: The basis of roulette wheel selection showing how the more fit members of the population are selected in proportion to their fitness values

position i is randomly selected ($1 \leq i \leq l-1$, where l is the length of the chromosome). Two new strings are then created by swapping the bits between positions $i+1$ and l. For example, suppose we have the following two chromosomes:

00100011110001
11000011001100

and the crossover point is chosen to be 6. Then the two new strings are:

00100011001100
11000011110001

The crossover operator is applied to the selected pairs of parents with a probability P_c, a typical value being 0.8 (i.e. there is an 80% chance that any of the $\mu/2$ pairs will actually undergo this type of recombination). Following the crossover phase mutation is applied to all individuals in the population. Here, each bit may be inverted (0 to 1 and vice versa) with a probability P_m. The mutation operator is usually assigned a low probability (e.g. 0.01).

This completes one complete cycle of the genetic algorithm. The new population then becomes the current population ready for a new cycle. The algorithm repeatedly applies this sequence for a predetermined number of iterations and/or until it converges

Many variants on the canonical genetic algorithm have been suggested. For example, it is common practice to carry forward the highest-ranking individuals unchanged: this is often referred to as an 'elitist' strategy and it ensures that the best individuals are not lost. In a *steady-state* genetic algorithm each iteration involves just one operator (mutation or crossover) with the resulting one or two individuals replacing the worst members of the population. A variety of different crossover schemes have been suggested, such as the two-point crossover. Real-valued chromosomes are an alternative to the binary representation of the canonical genetic algorithm. A real-valued chromosome consists of a string of real numbers, which correspond to the parameters of the problem (e.g. the torsion angles in the molecule). The main difference is that mutation is typically effected by adding a random increment (often chosen from a Gaussian distribution) to a randomly chosen parameter.

One of the main issues when implementing a genetic algorithm is the need to prevent premature convergence. The 'selection pressure' can be an important factor in determining whether such premature convergence occurs (or, conversely, that the program takes too long to converge to a solution). The selection pressure is defined as the relative probability that the fittest individual in a population will be chosen as a parent relative to an individual with average fitness. The selection pressure can be controlled by rescaling the fitness values when applying roulette wheel selection. Another way to deal with premature convergence is to use the *island model* in which one maintains a number of separate sub-populations within the whole and to introduce another operator that corresponds to the movement of an individual from one island to another. *Niching* is a related technique for achieving the same goal; here one tries to force the individuals in the population away from the most heavily populated regions of the space. This can be achieved by calculating a 'distance' between all pairs of members in the population, which is then used to reduce the chances of selecting pairs of individuals that are very similar

The main difference between the genetic algorithm and evolutionary programming is that the latter does not use a crossover operator. Rather, new individuals are generated from their parents using mutation alone. In addition, individuals in evolutionary programming are typically represented using a sequence of real numbers, rather than a binary representation (though it should be noted that chromosomes containing integers and real numbers have been successfully incorporated into genetic algorithms). At the start of each iteration of the evolutionary programming method one child is bred from each of the members of the current population using a mutation operator. During mutation each of the real variables in the chromosome is modified by adding a randomly generated real number, usually taken from a Gaussian distribution. These μ children are scored using the fitness function. The 2μ parents and children then compete for survival into the next generation This is achieved by performing a series of 'tournaments'. In the tournament stage, each individual is compared with a number M of opponents selected at random from the 2μ population of parents and children. The individuals are then ranked according to the number of 'wins' they achieve and the appropriate number selected from the top of the set to give the next population A win is recorded for each opponent with a worse fitness score. As the number of opponents, M, increases so the selection pressure increases, and it is necessary to choose an appropriate value for M to find the appropriate compromise between premature convergence and taking too long to find a solution.

Evolutionary strategies are very similar to evolutionary programming but differ in two key respects. First, crossover operators are permitted and, second, the probabilistic tournament is replaced with a straightforward ranking. At each iteration, λ children are generated using crossover and mutation from the current population. Typically, λ would be about seven times larger than μ. The children are scored and then the set of $(\mu + \lambda)$ parents and offspring are ranked according to fitness, with the top μ individuals being selected to form the next generation. A slight alternative to this approach is to select the μ individuals for the next population from just the λ new offspring. This is referred to as (μ, λ) selection.

Genetic and evolutionary algorithms are primarily intended for performing global optimisation. However, they do involve a significant random element and so they are not guaranteed to produce the same solution (e.g. the global minimum energy conformation) from each run, except for rather simple problems. What they are particularly useful for is producing solutions very close to the global optimum in a reasonable amount of time. An additional advantage of genetic and evolutionary algorithms is that as they maintain a population of possible solutions one may obtain several 'reasonable' solutions from a single run. Nevertheless, it is common practice to perform several runs in order to obtain a variety of different solutions and to investigate the nature of the energy surface.

Judson and co-workers were among the first to investigate the use of genetic algorithms in conformational analysis [Judson *et al.* 1993; McGarrah and Judson 1993]. Their implementation was tested on a variety of types of molecule, including a cyclic hexapeptide and a selection of more 'drug-like' molecules extracted from the Cambridge Structural Database. A key conclusion from these studies was the need to avoid premature convergence and to maintain a diverse population. When compared to a straightforward systematic search procedure the genetic algorithm was found to be particularly effective for the more flexible

molecules, especially those molecules with more than eight rotatable bonds [Meza *et al.* 1996].

9.9.2 Simulated Annealing

Annealing is the process in which the temperature of a molten substance is slowly reduced until the material crystallises to give a large single crystal. It is a technique that is widely used in many areas of manufacture, such as the production of silicon crystals for computer chips. A key feature of annealing is the use of very careful temperature control at the liquid–solid phase transition. The perfect crystal that is eventually obtained corresponds to the global minimum of the free energy Simulated annealing is a computational method that mimics this process in order to find the 'optimal' or 'best' solutions to problems which have a large number of possible solutions [Kirkpatrick *et al.* 1983].

In simulated annealing, a cost function takes the role of the free energy in physical annealing and a control parameter corresponds to the temperature. To use simulated annealing in conformational analysis the cost function would be the internal energy. At a given temperature the system is allowed to reach 'thermal equilibrium' using a molecular dynamics or Monte Carlo simulation. At high temperatures, the system is able to occupy high-energy regions of conformational space and to pass over high energy barriers. As the temperature falls, the lower energy states becomes more probable in accordance with the Boltzmann distribution. At absolute zero, the system should occupy the lowest-energy state (i.e. the global minimum energy conformation). To guarantee that the globally optimal solution is actually reached would require an infinite number of temperature steps, at each of which the system would have to come to thermal equilibrium. Careful temperature control is required when the energy of the system is comparable with the height of the barriers that separate one region of conformational space from another. This is often difficult to achieve in practice and so simulated annealing cannot *guarantee* to find the global minimum, much as a genetic algorithm cannot guarantee to identify the globally optimal solution. However, if the same answer is obtained from several different runs then there is a high probability that it corresponds to the true global minimum. Several simulated annealing runs may enable a series of low-energy conformations of a molecule to be obtained.

9.10 Solving Protein Structures Using Restrained Molecular Dynamics and Simulated Annealing

A particularly important application of molecular dynamics, often in conjunction with the simulated annealing method, is in the refinement of X-ray and NMR data to determine the three-dimensional structures of large biological molecules such as proteins. The aim of such refinement is to determine the conformation (or conformations) that best explain the experimental data. A modified form of molecular dynamics called *restrained molecular dynamics* is usually used in which additional terms, called *penalty functions*, are added to the potential energy function. These extra terms have the effect of penalising conformations

that do not agree with the experimental data. Molecular dynamics is used to explore the conformational space in order to find a conformation (or conformations) that not only has a low intrinsic energy but is also consistent with the experimental data. Simulated annealing can often be a convenient way to ensure that the conformational space is explored effectively.

9.10.1 X-ray Crystallographic Refinement

X-ray crystallography is a powerful technique for elucidating the structures of molecules. An X-ray diffraction pattern arises because of constructive and destructive interference between X-rays scattered from different parts of the crystal. An X-ray beam scattered by an electron at a point r travels a different distance to the detector than a beam scattered by an electron at the origin (Figure 9.22) As a consequence, the two scattered X-ray beams will have different phases and will interfere. As the detector is moved through different scattering angles, θ (Figure 9.22), the intensity of the scattered radiation will fluctuate between zero (destructive interference) and twice the amplitude of the original beam (constructive interference). In a real sample the amplitude of the scattered radiation from a point is proportional to the electron density at that point. The total signal reaching the detector is obtained by integrating the electron density over the whole crystal and is expressed as the structure factor, F. The structure factor is a complex number that can be written $F = |F|e^{i\phi}$, where $|F|$ is the amplitude and $e^{i\phi}$ is the phase. If the electron distribution is known (i.e. if we know the three-dimensional structure) it is possible to determine the structure factor for all scattering angles, and so we can calculate the X-ray diffraction pattern. The X-ray crystallographer is faced with the reverse problem to determine the electron distribution (and thereby the three-dimensional structure) from the diffraction pattern. The difficulty is that it is only possible to measure the intensities of the spots (which are equal to the amplitudes $|F|^2$), but not the phases; this is the famous *phase problem*, which is one of the major obstacles in solving an X-ray structure.

To obtain the electron density distribution it is necessary to guess, calculate or indirectly estimate the phases. Various methods have been developed to tackle the phase problem. For proteins the most common strategy is multiple isomorphous replacement in which the protein crystals are soaked in solutions containing salts of heavy metals such as mercury,

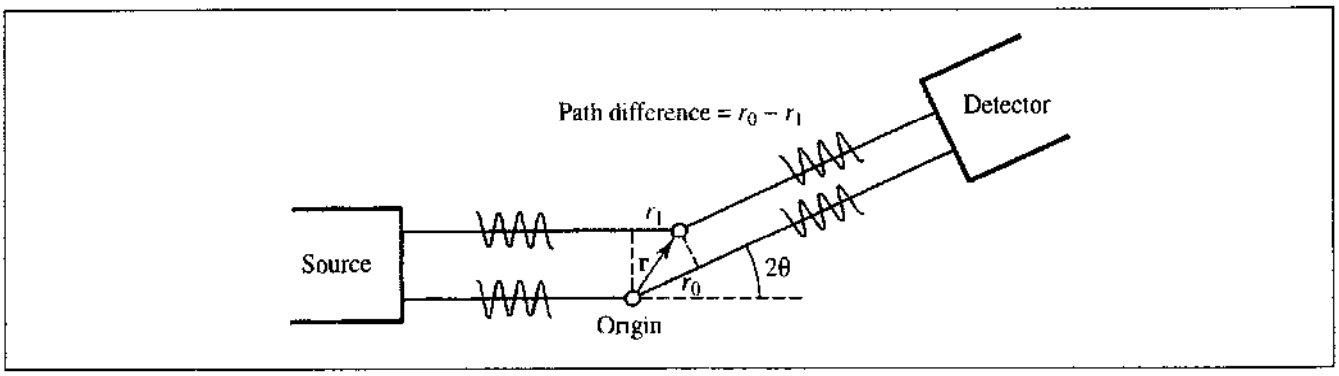


Fig 9 22 Schematic illustration of an X-ray scattering experiment The X-ray beam travels a different distance when scattered by an electron at the origin compared to an electron situated at **r**

platinum or silver. These heavy atoms may bind to specific parts of the protein (e g. mercury ions may react with exposed SH groups). By comparing the diffraction patterns of the native crystals and the crystals of the heavy-atom derivatives it can be possible to estimate some of the phases (under the assumption that no structural change occurs). Once some of the phases are known, others can be determined, and eventually an initial electron density map can be obtained. The electron density map is often represented as a three-dimensional surface by contouring at a constant value (Figure 9.23, colour plate section). An initial model of the molecule is then fitted to the electron density. When the diffraction experiment is performed at high resolution then the locations of individual atoms are often easy to identify. However, at lower resolution it can be difficult to find the optimal fit of the atoms in the model to the electron density as individual features are not so well defined. This is often the case in proteins.

The objective of the refinement is to obtain a structure that gives the best possible agreement with the experimental data. This is done by gradually changing the structure to give better and better agreement between the calculated and observed structure factor amplitudes. This degree of agreement is quantified by the value of the crystallographic R factor, which is defined as the difference between the observed ($|F_{obs}|$) and calculated ($|F_{calc}|$) structure factor amplitudes:

$$R = \frac{\sum \left| |F_{obs}| - |F_{calc}| \right|}{\sum |F_{obs}|} \tag{9.23}$$

Traditionally, least-squares methods have been used to refine protein crystal structures. In this method, a set of simultaneous equations is set up whose solutions correspond to a minimum of the R factor with respect to each of the atomic coordinates. Least-squares refinement requires an $N \times N$ matrix to be inverted, where N is the number of parameters. It is usually necessary to examine an evolving model visually every few cycles of the refinement to check that the structure looks reasonable. During visual examination it may be necessary to alter a model to give a better fit to the electron density and prevent the refinement falling into an incorrect local minimum. X-ray refinement is time consuming, requires substantial human involvement and is a skill which usually takes several years to acquire.

Jack and Levitt introduced molecular modelling techniques into the refinement in the form of an energy minimisation step (using a force field function) that was performed alternately with the least-squares refinement [Jack and Levitt 1978]. This approach was shown to give convergence to better structures. More recently, restrained molecular dynamics methods were introduced by Brunger, Kuriyan and Karplus [Brunger *et al.* 1987]. These methods have had a dramatic impact on the refinement of X-ray and NMR structure of proteins.

In the restrained molecular dynamics approach the total 'potential energy' is written as the sum of the usual potential energy and the penalty term, as usual:

$$E_{tot} = \mathscr{V}(\mathbf{r}^N) + E_{sf} \tag{9.24}$$

The additional penalty function that is added to the empirical potential energy function in restrained dynamics X-ray refinement has the form.

$$E_{sf} = S \sum [|F_{obs}| - |F_{calc}|]^2 \tag{9.25}$$

where E_{sf} describes the differences between the observed structure factor amplitudes and those calculated from the atomic model. S is a scale factor which is chosen so that the gradient of E_{sf} is comparable to the gradient of the potential energy part of the function. The conformational space is explored using molecular dynamics with simulated annealing, very high temperatures are used in the initial stages to permit the system to range widely over the energy surface. The temperature is then gradually reduced as the structure settles into a conformation which not only has a low energy but also a low R factor.

9.10.2 Molecular Dynamics Refinement of NMR Data

We have already discussed in Section 9.5.1 the type of information that NMR experiments can provide about the conformation of a molecule and the use of distance geometry for determining structures that are consistent with the experimental data. In the simplest molecular dynamics approach, we could incorporate harmonic restraint terms of the form $k(d - d_0)^2$ where d is the distance between the atoms in the current conformation and d_0 is the desired distance dynamics approach derived from the NMR spectrum. k is a force constant, the value of which determines how tightly the restraint should be applied. The information provided by the COSY experiment can also be expressed as a torsion angle via the Karplus equation; torsional restraints may be incorporated into the molecular dynamics energy function as an alternative to the use of distances. There are many other ways in which the restraints can be incorporated; for example, some practitioners prefer to penalise a structure only if the distance exceeds the target:

$$v(d) = k(d - d_0)^2 \qquad d > d_0 \tag{9.26}$$

$$v(d) = 0 \qquad d \leq d_0 \tag{9.27}$$

The atoms are prevented from coming too close by the van der Waals terms in the force field. More sophisticated functional forms have also been used which try to take into account the imprecise nature of the experimental values. A simple Hooke's law relationship implies that an exact value is known for the distance, whereas there can be significant uncertainty about its value. A more appropriate functional form has the following form:

$$v(d) = k_l(d - d_l)^2 \qquad d < d_l \tag{9.28}$$

$$v(d) = 0 \qquad d_l \leq d \leq d_u \tag{9.29}$$

$$v(d) = k_u(d - d_u)^2 \qquad d_u < d \tag{9.30}$$

This potential is shown schematically in Figure 9.24. d_l and d_u are the lower and upper distances that are considered to be consistent with the experimental data. $(d_l + d_u)/2$ is thus the assigned target distance obtained from a measurement of the NOESY intensity and the error associated with that measurement is $\pm(d_u - d_l)/2$. A distance between d_l and d_u incurs no penalty. Outside this region the restraint is applied using two harmonic potentials. These restraining potentials may have different force constants and so be of different steepness. In some functional forms, the harmonic potential is eventually replaced by a linear function, as illustrated in Figure 9.24.

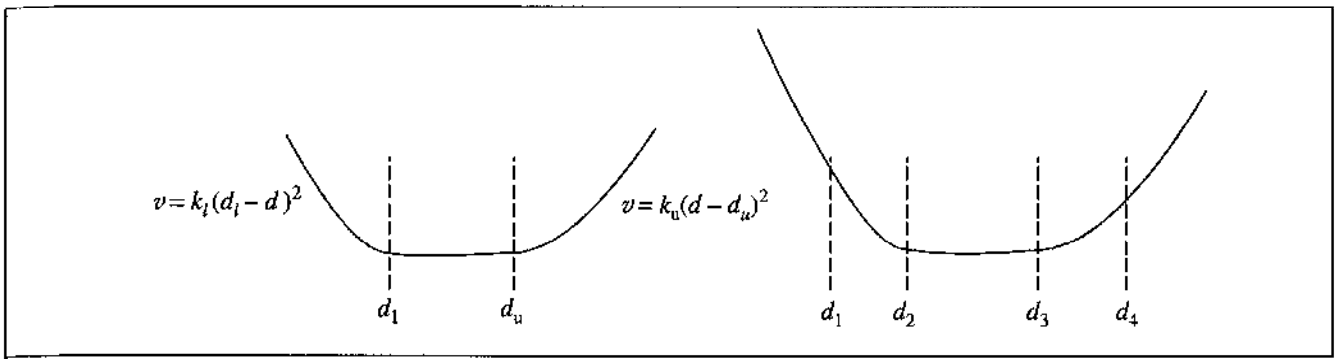


Fig 9.24: A restraining potential that does not penalise structures in which the distance lies between the lower and upper distances d_l and d_u and uses harmonic functions outside this range (left). The harmonic potentials may also be replaced by linear restraints further from this region (right)

9.10.3 Time-averaged NMR Refinement

If the molecule interconverts between two or more conformations on a timescale that is rapid compared to the chemical shift timescale then the NMR spectrum only shows the average of the signals from the individual conformations. This behaviour is illustrated schematically in Figure 9.25, where an atom or group (such as a leucine side chain) interconverts between two energy minima within the protein. The NMR spectrum comprises a single peak that is a weighted average of the resonances from the two individual conformations. If the two conformations make distinct interactions then two sets of distance restraints can be derived, and a standard refinement procedure would attempt to satisfy both sets of restraints simultaneously This would lead to a conformation in which the group is positioned at the top of the barrier between the two minima. This incorrect result is a consequence of assuming that one single structure is consistent with all of the experimental data, rather than recognising that the experimental data may arise from more than one conformation.

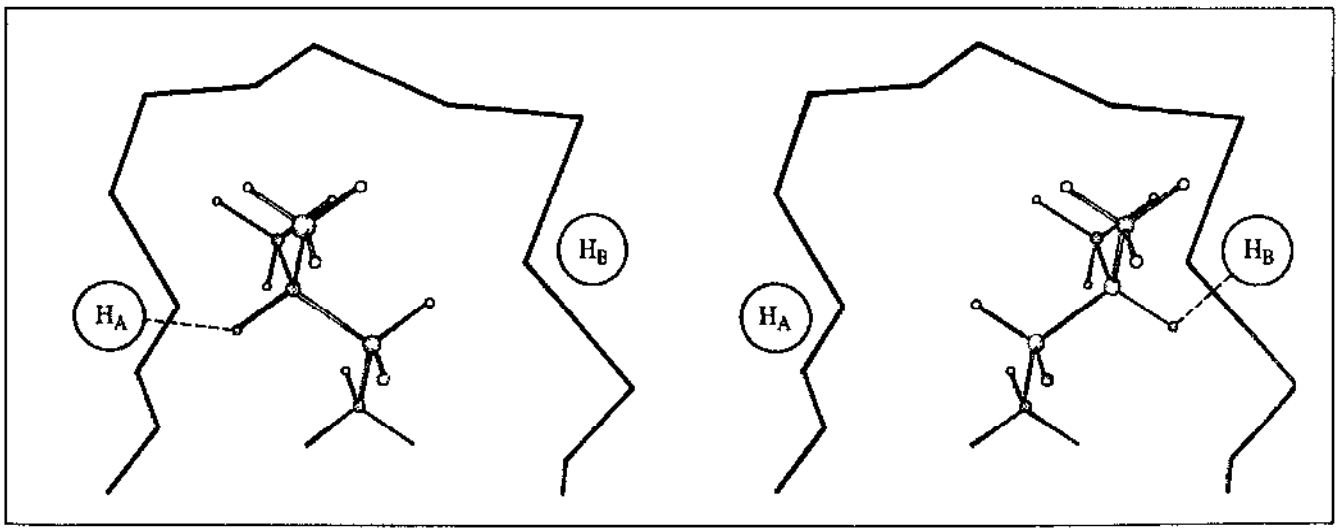


Fig 9 25: If the leucine side chain interconverts rapidly between two conformations then the NMR spectrum will be an average of them With a traditional refinement this leads to a structure that simultaneously tries to satisfy all restraints and is at the top of the energy barrier between the two minima

Time-averaged restraints may be able to overcome this problem [Torda *et al.* 1990]. Rather than using the instantaneous value of a distance in the restraint function, the time-averaged restraint method uses a value that is averaged over time. The simple harmonic error function then becomes:

$$v(d) = k(\langle d(t)\rangle - d_0)^2 \tag{9.31}$$

where $\langle d(t)\rangle$ is the time-averaged value of the distance, as obtained from the molecular dynamics simulation. At a time t', $\langle d(t')\rangle$ is given by:

$$\langle d(t')\rangle = \frac{1}{t'}\int_0^{t'} d(t)\,dt \tag{9.32}$$

As the intensity of the NOESY signal is proportional to the inverse sixth power of the distance, the 'distance' to use in this case is actually given by:

$$d_{\text{NOESY}} = \langle d(t')^{-6}\rangle^{-1/6} \tag{9.33}$$

The time-averaged value of the distance that should be used in the error function is thus

$$\langle d(t')\rangle = \left[\frac{1}{t'}\int_0^{t'} d(t)^{-6}\,dt\right]^{-1/6} \tag{9.34}$$

One way to implement time-averaged restraints is to evaluate $\langle d(t')\rangle$ using Equation (9.34) as the simulation proceeds and incorporate the value in the error function (Equation (9.31)). If the simulation is run for long enough, all the accessible conformational states should be visited and be included in the calculation of the average distances. However, it is rarely possible to achieve the length of simulation needed to ensure that the conformation space has been adequately covered. We require a method that can supply an accurate picture of the dynamics of the molecule with minimal computational effort. The normalisation factor $1/t'$ in Equation (9.34) becomes progressively larger as the simulation proceeds, thus making $\langle d(t')\rangle$ increasingly less sensitive to the current value of the distance. What we require is a means to bias the instantaneous value of $\langle d(t')\rangle$ towards the values from the most recent part of the simulation. In this way, if the 'current' value of $\langle d(t')\rangle$ is incompatible with the restraint, then the penalty function should be proportionately increased. This can be achieved using an exponential 'memory function' which has the effect of weighting the recent history more heavily. Various memory functions are possible; one functional form is:

$$\langle d(t')\rangle = \left(\frac{\int_0^{t'} e^{(t-t')/\tau}\, d(t)^{-6}\,dt}{\int_0^{t'} e^{(t-t')}\,dt}\right)^{-1/6} \tag{9.35}$$

where τ is the time constant for the exponential damping factor. Small values of τ give a higher weighting to recent values of the distance. If τ is infinite, all the past history of the simulation is given equal weight.

The time-averaged restraint method is quite complicated to implement, and some skill is required when choosing the most appropriate functional form and the damping constant. The data produced by the simulation must also be interpreted with care The technique is only truly applicable where the conformations are relatively close together, so that

interconversion between the different conformations can be achieved relatively easily. Nevertheless, the technique does provide a more accurate representation of the dynamics of the real system, and it does enable the conformation to fluctuate more. One drawback of any restraint method is that the additional penalty terms represent an unnatural perturbation of the forces within the molecule. When using 'static' restraints the size of the force constants for the restraint terms can be quite large, which can often cause the conformations to have rather high energies. Smaller force constants can often be used with time-averaged restraints, which means that the conformations generally have lower energies.

9.11 Structural Databases

Experimental information about the structures of molecules can often be extremely useful for forming theories of conformational analysis and helping to predict the structures of molecules for which no experimental information is available. The most important technique currently available for determining the three-dimensional structure of molecules is X-ray crystallography. The international crystallographic community has established centres where crystallographic data is collected and then distributed in electronic form. Two particularly important databases for the molecular modeller are the Cambridge Structural Database (CSD) [Allen *et al.* 1979], which contains crystal structures of organic and organometallic molecules; and the Protein Databank (PDB) [Bernstein *et al.* 1977; Berman *et al.* 2000], which contains structures of proteins and some DNA fragments. Other databases are also available, such as the Inorganic Structural Database of inorganic compounds and complexes [Bergerhoff *et al.* 1983].

A database is of little use without software tools to search, extract and manipulate the data. A simple use of a database is for extracting information about a particular molecule or group of molecules. For example, one may wish to retrieve the crystal structure of ranitidine (Figure 9.26). The molecule(s) may be specified in a variety of ways, such as by name, molecular formula or literature citation. The data may also be identified by creating a two-dimensional representation of the molecule (as in Figure 9.26) and using a substructure search program (see Section 12.2) to search the database. In fact, the CSD contains two entries for ranitidine: one is the crystal structure of the hydrochloride salt and the other is the structure of the oxalate salt. Crystallographic databases have also been used to develop an understanding of the factors that influence the conformations of molecules, and of the ways in which molecules interact with each other. For example, the CSD has been comprehensively analysed to characterise how the lengths of chemical

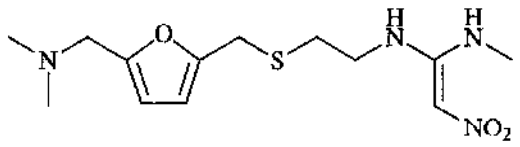


Fig 9 26 Ranitidine.

bonds depend upon the atomic numbers, hybridisations and environment of the atoms involved [Allen *et al.* 1987]. A major use of the CSD is searching for molecules which contain a particular fragment, in order to investigate the conformation(s) that the fragment adopts. Intermolecular interactions can also be investigated. For example, analyses of intermolecular hydrogen bonding have revealed distinct distance and angular preferences [Murray-Rust and Glusker 1984; Glusker 1995]. This type of analysis can now be applied to a wide range of functional groups and molecular fragments, as illustrated in Figure 9.27 (colour plate section), which shows the distribution of OH groups around thiazole rings [Bruno *et al.* 1997]. This shows that the nitrogen in thiazole is a much stronger hydrogen bond acceptor than the sulphur atom.

The protein database has provided much useful information about the structures that proteins adopt and the PDB has been extensively analysed to try to understand the principles that determine why a given amino acid sequence folds into one specific conformation. We shall discuss the use of molecular modelling methods to predict protein structure in more detail in Chapter 10. Here we shall just mention one interesting way in which the information contained in the protein database has been put to a practical purpose. One of the steps in determining the structure of a protein by X-ray crystallography involves fitting the polypeptide chain to the electron density. This can be a complex and time-consuming task, even with today's sophisticated molecular graphics. Computer programs have been developed which extract the conformations of short polypeptide fragments (up to four amino acids long) from known X-ray structures [Jones and Thirup 1986]. These fragments are then used to generate a chain that fits the electron density. This method is feasible because a given segment of polypeptide chain often adopts a limited selection of conformations in protein structures, and a significant proportion of an unknown protein structure can often be constructed using such 'spare parts' taken from other proteins.

It should be remembered that a crystallographic database can only provide information about the crystalline state of matter, and that the possible influence of crystal packing forces should always be taken into account. This is less of a concern for proteins than for 'small' molecules as protein crystals contain a large amount of water and indeed NMR studies have established that proteins have approximately the same structure in solution as in the crystal. A second, more subtle, bias is that crystallographic databases contain only molecules that can be crystallised and indeed only those molecules whose X-ray structures were considered important enough to be published. The structures in a crystallographic database may therefore not necessarily be a wholly representative set.

9.12 Molecular Fitting

Fitting is the procedure whereby two or more conformations of the same or different molecules are oriented in space so that particular atoms or functional groups are optimally superimposed upon each other. Fitting methods are widely used in molecular modelling. For example, fitting is an integral part of many conformational search algorithms, particularly those that require each conformation to be compared with those generated previously in order to check for duplicates.

A molecular fitting algorithm requires a numerical measure of the 'difference' between two structures when they are positioned in space. The objective of the fitting procedure is to find the relative orientations of the molecules in which this function is minimised. The most common measure of the fit between two structures is the root mean square distance between pairs of atoms, or RMSD:

$$\text{RMSD} = \sqrt{\frac{\sum_{i=1}^{N_{\text{atoms}}} d_i^2}{N_{\text{atoms}}}} \tag{9.36}$$

where N_{atoms} is the number of atoms over which the RMSD is measured and d_i is the distance between the coordinates of atom i in the two structures, when they are overlaid.

When fitting two structures, the aim is to find the relative orientations of the two molecules in which the RMSD is a minimum. Many methods have been devised to perform this seemingly innocuous calculation. Some algorithms, such as that described by Ferro and Hermans [Ferro and Hermans 1977] use an iterative procedure in which the one molecule is moved relative to the other, gradually reducing the RMSD. Other methods locate the best fit directly, such as Kabsch's algorithm [Kabsch 1978].

If the molecules are flexible then a better fit might be achieved if one or both of the molecules can change their conformation (for example, by rotating about single bonds). This is often referred to as *flexible fitting* or *template forcing*. In its simplest form flexible fitting is achieved by minimising the RMSD using a special minimisation algorithm that permits rotation about single bonds as well as translation and rotation in space. An alternative approach is to use restrained molecular dynamics, which may enable a more thorough exploration of the conformational space in order to find the best fit. Here, restraints are placed on the distance between pairs of matched atoms, which are incorporated into the energy function as additional penalty terms.

9.13 Clustering Algorithms and Pattern Recognition Techniques

Molecular modelling programs can generate a large amount of data, which must often be processed and analysed. Many of the conformational search algorithms that we have considered can generate conformations that are very similar, if not identical. Under such circumstances it is desirable to be able to select from the data set a smaller, 'representative' set of conformations for subsequent analysis. This can be done using cluster analysis, which groups together 'similar' objects, from which the representatives can be extracted (Figure 9.28).

There is no 'correct' method of performing cluster analysis and a large number of algorithms have been devised from which one must choose the most appropriate approach. There can also be a wide variation in the efficiency of the various cluster algorithms, which may be an important consideration if the data set is large.

A cluster analysis requires a measure of the 'similarity' (or dissimilarity) between pairs of objects. When comparing conformations, the RMSD would be an obvious measure to use.

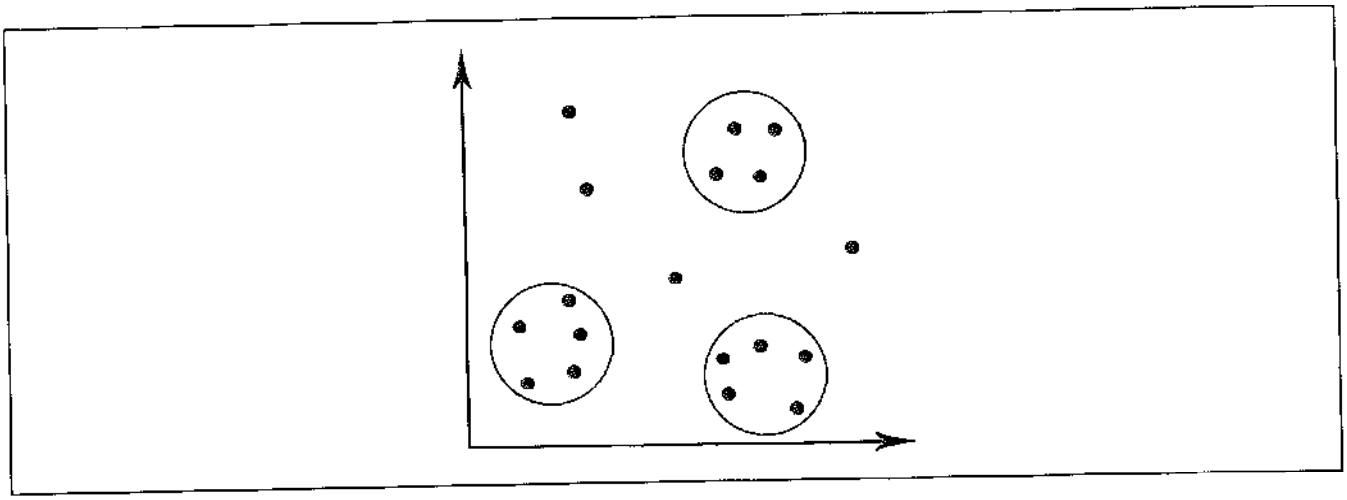


Fig. 9.28: The aim of cluster analysis is to group together 'similar' objects.

Alternatively, the 'distance' between two conformations can be measured in terms of their torsion angles. Here, there may be more than one way in which the 'distance' can be calculated. The Euclidean distance between two conformations would be calculated using.

$$d_{ij} = \sqrt{\sum_{m=1}^{N_{tor}} (\omega_{m,i} - \omega_{m,j})} \tag{9.37}$$

where $\omega_{m,i}$ is the value of torsion angle m in conformation i. N_{tor} is the total number of torsion angles. An alternative is the Hamming distance (also known as the Manhattan or city-block distance, Figure 9.29):

$$d_{ij} = \sum_{m=1}^{N_{tor}} |\omega_{m,i} - \omega_{m,j}| \tag{9.38}$$

When using torsion angles to calculate 'distances' between conformations it is important to remember that a torsion angle is a cyclic measure and that the difference should be measured along the shortest path, in either a clockwise or an anticlockwise direction. The clusters produced using the RMSD and the torsion angle measures may be very different. This is due to a 'leverage' effect when using torsion angles, which arises because small

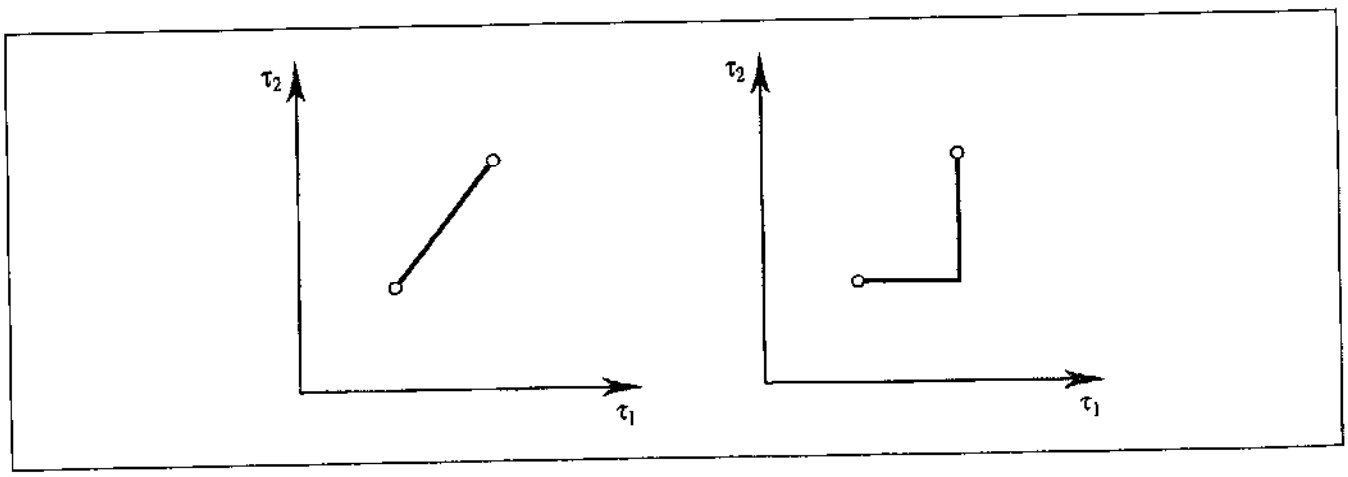


Fig 9.29 Euclidean and Hamming distance measures of torsional similarity

changes in the torsion angles in the middle of a molecule can give rise to large movements near the ends. The RMSD produces clusters in which the molecules have a similar shape.

One family of relatively straightforward clustering algorithms is the *linkage methods*. These algorithms first require the distance between each pair of conformations to be calculated. At the start of the cluster analysis the data set contains as many clusters as there are conformations; each cluster contains just a single conformation. At each step the total number of clusters is reduced by one by merging the 'closest' or 'most similar' pair of clusters into a single cluster. Thus in the first step the closest two conformations are merged into a single cluster. In the next step, the closest two clusters are merged, and so on. Clustering continues until the distance between the closest pair of clusters exceeds a predetermined value, until the number of clusters falls below a specified maximum number, or until all the conformations have been merged into a single cluster. Such algorithms are referred to as *agglomerative* methods, in contrast to *divisive* clustering algorithms, which start with a single cluster containing all of the data that is then partitioned into clusters. A representative conformation may then be chosen from each cluster, for example the conformation that is closest to the average structure of the cluster. The linkage methods differ in the way in which they calculate the distance between two clusters.

In the *single-linkage* or *nearest-neighbour* method the 'distance' between a pair of clusters is equal to the shortest distance between any two members, one from each cluster. The *complete-linkage* or *furthest-neighbour* method is the logical opposite of the single-linkage method in that it considers the furthest pair of objects in a pair of clusters. The *group average* method computes the average of the similarities between all pairs of objects in the two clusters.

We can contrast these methods using the data shown in Figure 9.30, which were obtained by searching the Cambridge Structural Database for the ribose phosphate fragment also shown

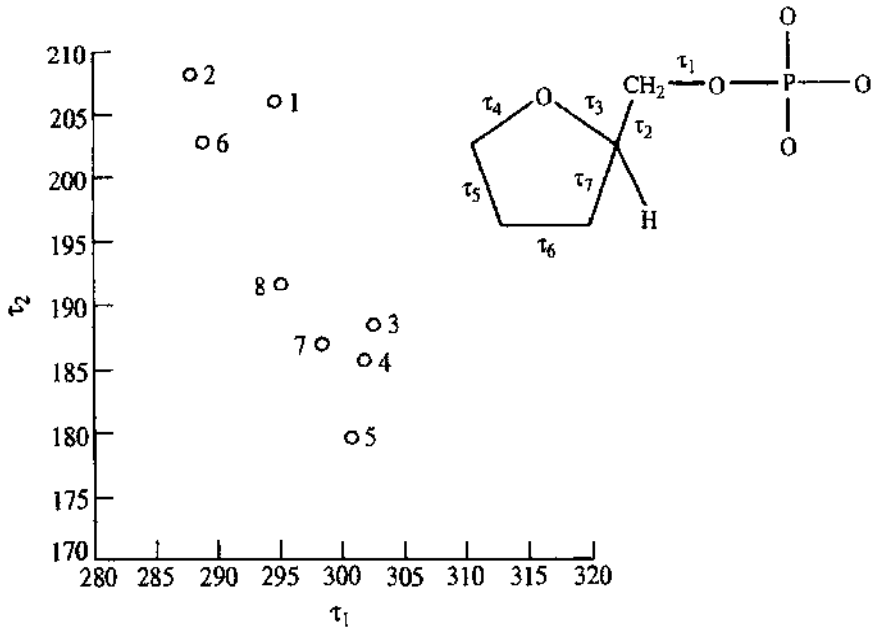


Fig. 9.30: Ribose phosphate fragment used to extract data from Cambridge Structural Database and eight sets of torsion angle values for τ_1 and τ_2

	1	2	3	4	5	6	7	8
1	0.0	7 2	19.3	21.4	26.9	6.6	19.4	14 2
2	7.2	0.0	24.6	26.3	31.1	5.3	23.6	17.9
3	19.3	24.6	0.0	2.7	8.8	19.9	4.5	8.1
4	21.4	26.3	2.7	0.0	6 1	21.4	3.7	8.9
5	26 9	31.1	8.8	6.1	0.0	26 0	7.6	13 2
6	6.6	5.3	19.9	21.4	26.0	0.0	18.5	12.8
7	19.4	23.6	4.5	3.7	7.6	18.5	0.0	5 7
8	14.2	17.9	8.1	8 9	13.2	12.8	5.7	0.0

Table 9.2. Distance matrix for eight ribose phosphate fragments.

in Figure 9.30. A total of 44 molecules were found to contain this fragment. The values of the two torsion angles τ_1 and τ_2 indicated in Figure 9.28 were determined for each occurrence of the fragment (some molecules contained more than one representative of the fragment). For simplicity, the results for just eight fragments are plotted in Figure 9.28. The distance matrix for these eight sets of two torsion angles, calculated using a Euclidean measure, is given in Table 9.2.

All of the clustering methods first join the two structures that are 'closest' (conformations 3 and 4), to which is then added conformation 7 In the third step, conformations 2 and 6 are connected. In the fourth step, the single-linkage and complete-linkage methods differ; the single-linkage algorithm joins conformation 8 to the cluster 3–4–7, whereas the complete-linkage method joins conformation 1 to the cluster 2–6. The order in which the points are clustered together for the three linkage methods is given in Table 9.3. As can be seen, the three linkage methods form the clusters in a similar but not identical order.

These three linkage methods are all *hierarchical agglomerative* clustering methods, because there is a specific order in which the clusters are formed and amalgamated. The same basic approach underlies all such methods, involving a series of iterations at each of which the two closest clusters are identified and combined into a larger cluster. The process continues until just a single cluster remains. These methods have the advantage of being simple to program, and they also produce a clustering that is independent of the order in

Step number	Single linkage	Complete linkage	Group average
1	3–4 (2.7)	3–4 (2.7)	3–4 (2.7)
2	3–4–7 (3.7)	3–4–7 (4 5)	3–4–7 (4.1)
3	2–6 (5.3)	2–6 (5.3)	2–6 (5.3)
4	3–4–7–8 (5.7)	2–6–1 (7.2)	2–6–1 (6.9)
5	3–4–7–8–5 (6.1)	3–4–7–5 (8.8)	3–4–7–5 (7.5)
6	2–6–1 (6.6)	3–4–7–5–8 (13.2)	3–4–7–5–8 (9 0)
7	2–6–1–3–4–7–8–5 (12.8)	2–6–1–3–4–7–5–8 (31.1)	2–6–1–3–4–7–5–8 (21 3)

Table 9.3: A comparison of the single-linkage, complete-linkage and average-linkage cluster methods using the data in Table 9 2 The figures in parentheses indicate the 'distance' between the clusters as they are formed In this particular case Ward's clustering follows the same order of cluster formation as the group average method.

which the objects are stored. However, they do suffer from some drawbacks. For example, the commonly used single-linkage method tends to produce long, elongated clusters. In addition, simple implementations require an $M \times M$ similarity matrix to be calculated, which can severely limit their applicability when clustering large data sets.

A fourth hierarchical method that is quite popular is Ward's method [Ward 1963]. This method merges those two clusters whose fusion minimises the 'information loss' due to the fusion. Information loss is defined in terms of a function which for each cluster i corresponds to the total sum of squared deviations from the mean of the cluster:

$$E_i = \sum_{j=1}^{N_i} (|\mathbf{r}_j - \bar{\mathbf{r}}_i|)^2 \tag{9.39}$$

The summation runs over the N_i objects in cluster i, each located at $\mathbf{r}_j$ and where the mean of the cluster is $\bar{\mathbf{r}}_i$. The total information loss is calculated by adding together the values for each cluster. At each iteration that pair of clusters which gives rise to the smallest increase in the total error function are merged. Two more hierarchical clustering algorithms are the centroid method, which determines the distance between two clusters as the distance between their centroids, and the median method, which represents each cluster by the coordinates of the median value Fortunately, all six hierarchical agglomerative methods can be represented by a single equation, first proposed by Lance and Williams [Lance and Williams 1967], with the different algorithms having different coefficients.

A hierarchical clustering can be represented visually by constructing a *dendrogram*, which indicates the relationship between the items in the data set. A sample dendrogram is shown in Figure 9.31 for the single-linkage clustering described above. Along the x axis are represented the M individual objects. The y axis indicates the intercluster distance.

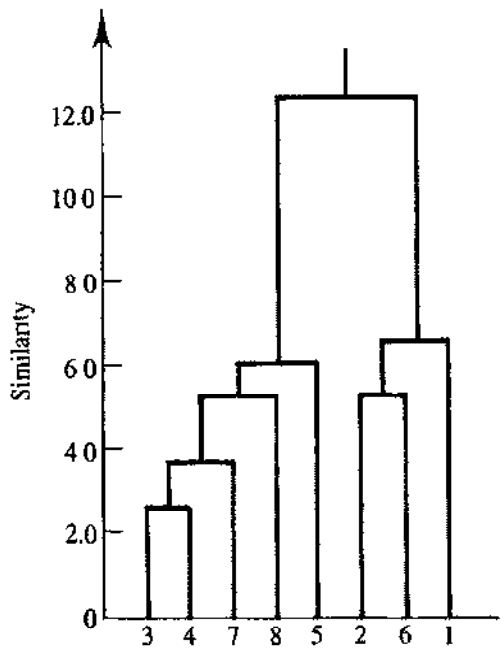


Fig 9.31 Dendrogram of the single-linkage data in Table 9.3.

The dendrogram enables us to identify how many clusters there are at any stage, and what the members of those clusters are. A dendrogram can thus be a very useful way to show the underlying structure of the data and for suggesting the appropriate number of clusters to choose. A line drawn across the dendrogram enables one to read off how many clusters there are at that particular distance measure. For example, there are four clusters at a value of 6.0. For the data in Figure 9.30 it would probably be decided that the data fall into two clusters, one containing conformations 1, 2 and 6 and the other containing 3, 4, 5, 7 and 8. As this example illustrates, deciding how many clusters there are can be somewhat subjective; a small threshold can lead to a large number of 'tight' clusters (and frequently many clusters with just one member) whereas a larger threshold can produce clusters that are spread out.

An example of a non-hierarchical clustering method is the Jarvis–Patrick algorithm [Jarvis and Patrick 1973]. The Jarvis–Patrick method uses a 'nearest-neighbours' approach. The nearest neighbours of each conformation are the conformations that are the shortest distance away. Two conformations are considered to be in the same cluster in the Jarvis–Patrick method if they satisfy the following criteria:

1. They are in each other's list of m nearest neighbours
2. They have p (where $p < m$) nearest neighbours in common.

Conformations can thus be placed in clusters and clusters fused together (because any two individual elements satisfy the two criteria) without any hierarchical relationships. The Jarvis–Patrick method can also be extended to take account not only of the number of nearest neighbours but also the position of each conformation within the neighbour list. In addition, it is possible to require that a molecule's nearest neighbours must be within some defined distance. This ensures that the nearest neighbours of each conformation are not too dissimilar.

To illustrate the use of the Jarvis–Patrick method, let us consider the data in Figure 9.30 once more. The three nearest neighbours of each fragment are given in Table 9.4. Suppose we require that two out of the three nearest neighbours should be common. If we examine the pair 1, 2 we find that each neighbour list contains the other fragment and the remaining two nearest neighbours are the same (i.e. 6, 8). These objects would therefore be placed in the same cluster. However, fragments 2 and 6 would not be considered in the same cluster

Fragment	Nearest neighbours
1	2, 6, 8
2	1, 6, 8
3	4, 7, 8
4	3, 5, 7
5	3, 4, 7
6	1, 2, 8
7	3, 4, 8
8	3, 4, 7

Table 9.4. The three nearest neighbours of each fragment in Figure 9 30

according to these criteria; although they are in each other's list they do not have two out of three nearest neighbours in common. One advantage of the Jarvis–Patrick algorithm is that it can be used to cluster very large data sets which may be too large for any of the hierarchical methods to handle, due to their typically larger computational requirements. The K-means method, which is another non-hierarchical approach, is also applicable to larger sets. The K-means algorithm first chooses a set of c 'seed' objects, usually at random. The remaining objects are assigned to the nearest seed to give an initial set of c clusters. The centroids of each of these clusters are then determined and the objects are reassigned to the nearest of these new cluster centroids. New centroids are then determined, and the process continues until no objects change clusters. The K-means method is obviously dependent upon the initial set of (random) cluster centroids and different results will usually result from different initial seeds.

A common use of cluster analysis is in selecting a set of representative molecules from a large chemical database; the advent of robotic methods for high-throughput screening has made this of particular interest and some studies have been published comparing various approaches [Downs *et al.* 1994]. It is always crucial to bear in mind that, in addition to the differences between clustering algorithms, the performance of a cluster analysis also depends critically upon the methods used to calculate the distances between the objects in the data set.

9.14 Reducing the Dimensionality of a Data Set

The *dimensionality* of a data set is the number of variables that are used to describe each object. For example, a conformation of a cyclohexane ring might be described in terms of the six torsion angles in the ring. However, it is often found that there are significant correlations between these variables. Under such circumstances, a cluster analysis is often facilitated by reducing the dimensionality of a data set to eliminate these correlations. *Principal components analysis* (PCA) is a commonly used method for reducing the dimensionality of a data set.

9.14.1 Principal Components Analysis

Consider the data shown in Figure 9.32. It is easy to see that there is a high degree of correlation between the x and the y values. If we were to define a new variable, $z = x + y$, then we could express most of the variation in the data as the values of this new variable z. The new variable is called a *principal component*. In general, a principal component is a linear combination of the variables:

$$p_i = \sum_{j=1}^{v} c_{i,j} x_j \tag{9.40}$$

where p_i is the ith principal component and $c_{i,j}$ is the coefficient of the variable x_j. There are v such variables The first principal component of a data set corresponds to that linear combination of the variables which gives the 'best fit' straight line through the data when

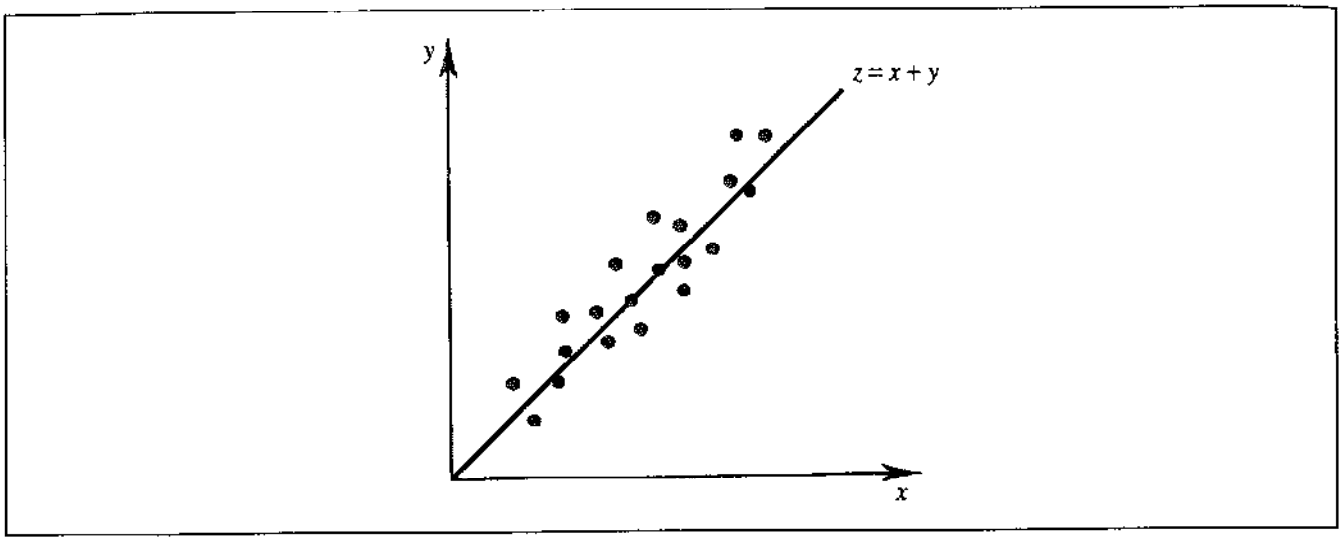


Fig. 9.32 Most of the variance in this set of highly correlated data values can be explained in terms of a new variable, $z = x + y$

they are plotted in the v-dimensional space. More specifically, the first principal component maximises the *variance* in the data so that the data have their greatest 'spread' of values along the first principal component. This is clear in the two-dimensional example shown in Figure 9.32. The second and subsequent principal components account for the maximum variance in the data not already accounted for by previous principal components. Each principal component corresponds to an axis in a v-dimensional space, and each principal component is orthogonal to all the other principal components. There can clearly be as many principal components as there are dimensions in the original data, and indeed in order to explain all of the variation in the data it is usually necessary to include all the principal components. However, in many cases only a few principal components may be required to explain a significant proportion of the variation in the data. If only one or two principal components can explain most of the data then a graphical representation is possible.

The principal components are calculated using standard matrix techniques [Chatfield and Collins 1980]. The first step is to calculate the variance–covariance matrix. If there are s observations, each of which contains v values, then the data set can be represented as a matrix $\mathbf{D}$ with v rows and s columns. The variance–covariance matrix $\mathbf{Z}$ is:

$$\mathbf{Z} = \mathbf{D}^{\mathrm{T}}\mathbf{D} \tag{9.41}$$

The eigenvectors of $\mathbf{Z}$ are the coefficients of the principal components. As $\mathbf{Z}$ is a square symmetric matrix its eigenvectors will be orthogonal (provided there are no degenerate eigenvalues). The eigenvalues and their associated eigenvectors can be obtained by solving the secular equation $|\mathbf{Z} - \lambda\mathbf{I}| = 0$ or by matrix diagonalisation. The first principal component corresponds to the largest eigenvalue, the second principal component to the second largest eigenvalue, and so on. The ith principal component accounts for a proportion $\lambda_i / \sum_{j=1}^{v} \lambda_j$ of the total variance in the data. The first m principal components therefore account for $\sum_{j=1}^{m} \lambda_j / \sum_{j=1}^{v} \lambda_j$ of the total variation in the data.

As an example of the application of principal components analysis, we shall consider the conformations adopted by the five-membered ribose ring in our set of conformations

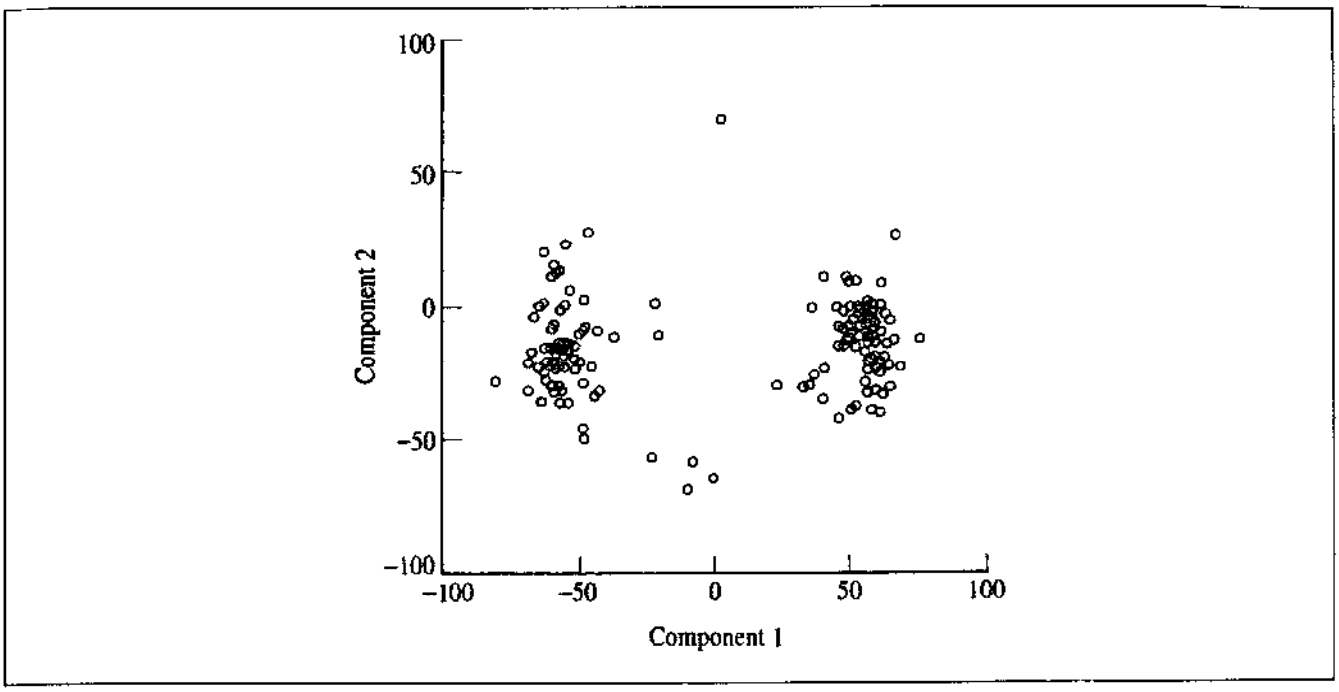


Fig 9 33. Scatterplot of the first two principal components for the ring torsion angles τ_3–τ_7

extracted from the Cambridge Structural Database. The conformation of a five-membered ring can be described in terms of five torsion angles ($\tau_3 \ldots \tau_7$ in Figure 9.30). As we cannot visualise points in a five-dimensional space it would clearly be useful to reduce the dimensionality of the data set. When a principal components analysis is performed on the data, the following results are obtained:

Principal component	Proportion of variance explained (%)	$c(\tau_3)$	$c(\tau_4)$	$c(\tau_5)$	$c(\tau_6)$	$c(\tau_7)$
1	85.9	−0.14	−0.26	0.55	−0.61	0.48
2	14.0	−0.63	0.59	−0.31	−0.06	0 41
3	0.0002	−0 19	0 50	0.65	−0.004	−0.53
4	0.0001	−0.47	−0.38	0.12	0.71	0.19
5	0.0001	0.58	0.43	0 28	0.35	0.53

It can thus be seen that most of the variation in the data (85.9%) is explained by the first principal component, with all but a fraction being explained by the first two components. These two principal components can be plotted as a scatter graph, as shown in Figure 9.33, suggesting that there does indeed seem to be some clustering of the conformations of the five-membered ring in this particular data set.

9.15 Covering Conformational Space: Poling

As we have discussed, a common strategy in conformational analysis is to perform a two-stage process involving, first, the generation of a large number of minimum energy

conformations followed by the selection of a subset using a technique such as cluster analysis. This subset may then be considered 'representative' of the conformational space in subsequent calculations. There can be both practical and scientific objections to this approach. One of the practical problems is that it may take some considerable computational effort to first explore the conformational space and then to cluster the resulting conformations. One of the scientific objections is that, by restricting the initial search to minimum energy conformations, one may not adequately cover the conformational space. Consider the case of a broad, shallow minimum. It might be better to describe this region of the conformational space using an ensemble of structures rather than just a single minimum energy structure. A technique termed 'poling' has been described which is intended to promote the generation of diversity in the conformational coverage [Smellie *et al* 1995a, b]. The poling approach introduces a penalty function into the geometry optimisation step that is a common component of most conformational searching procedures. This function is designed to penalise a conformation that is too close to any of the conformations already generated.

The effect of poling on the conformational space is shown using a one-dimensional energy surface in Figure 9.34, which contains two energy minima. Suppose we first generate conformation 1. The poling function is now introduced to modify the energy surface in the region of this conformation. This can have the effect of introducing new minima (labelled 2 in Figure 9.34) into the energy surface. In the third iteration poling functions are introduced around conformations 1 and 2, enabling conformation 3 to be produced. Note that in this example the unperturbed energy surface contains only two minima,

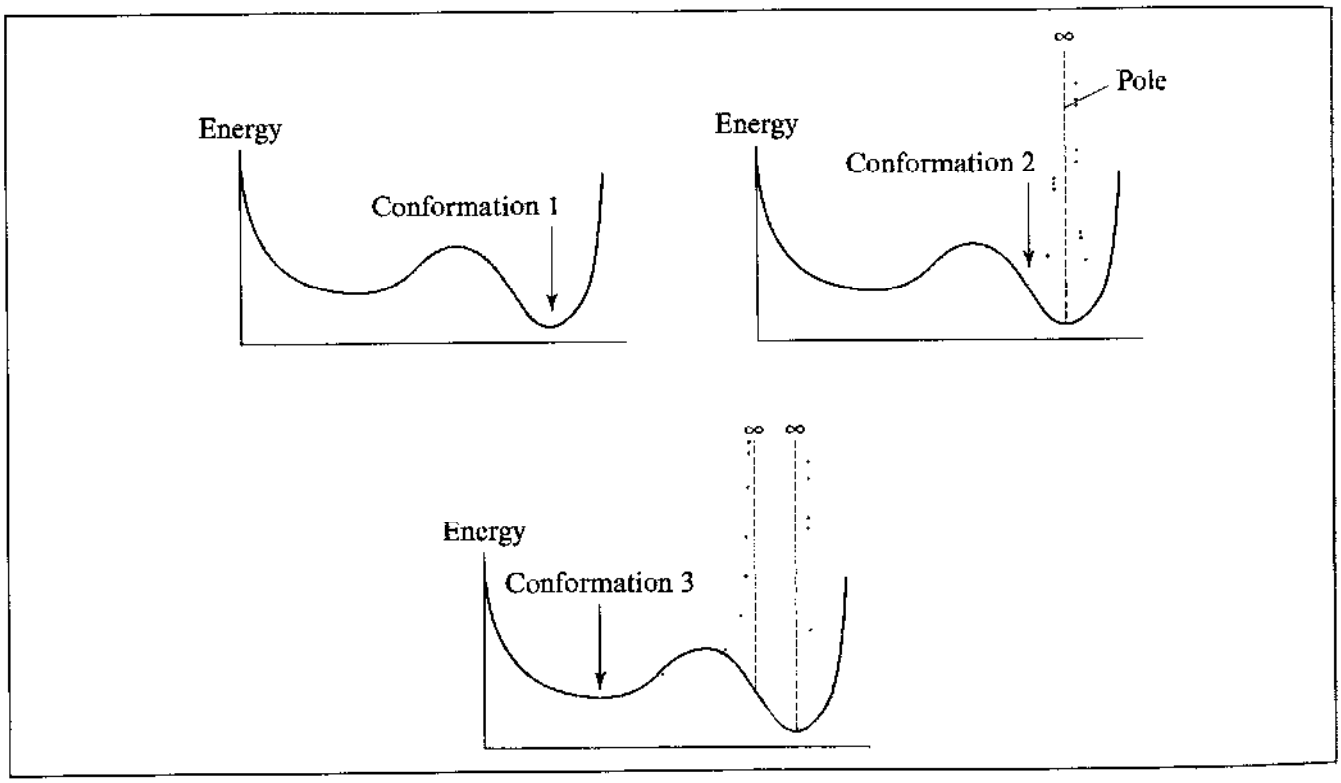


Fig. 9.34. Modification of the energy surface using poling (Figure adapted from Smellie A S, S L Teig and P Towbin 1995b Poling Promoting Conformational Variation Journal of Computational Chemistry **16**:171–187.)

which would be the only structures identified by a traditional conformational search algorithm. The use of the poling function not only ensures that previously explored regions of conformational space are avoided but can also lead to a wider range of conformations being generated, that might be considered to better represent the accessible conformational space.

The poling function typically adopts the following general functional form:

$$F_{\text{pole}} = W_{\text{pole}} \sum_i \frac{1}{(D_i)^N} \tag{9.42}$$

$$D_i = \left(\frac{\sum_{j=1}^{N_d} (d_{j,\text{curr}} - d_{j,i})^2}{N_d} \right)^{1/2} \tag{9.43}$$

Here, N_d poling distances are being compared between the current conformation and a previously generated conformation, i. Thus $d_{j,\text{curr}}$ represents one of these N_d poling distances in the current conformation and $d_{j\,i}$ represents the equivalent distance in the ith conformation. D_i is thus the root-mean-square difference between the current conformation and conformation i over these poling distances The steepness of these poling functions can be changed by modifying the power N One way that the poling distance could be implemented would be to sum over all interatomic distances in the molecule. However, this would be rather inefficient as the number of such distances increases with the square of the number of atoms in the molecule. Rather, in the published work a set of 'chemically significant features', such as hydrogen bond donors and hydrogen bond acceptors, was identified and the poling distance was defined from each such feature to the centroid of all the features.

9.16 A 'Classic' Optimisation Problem: Predicting Crystal Structures

Many molecules are obtained and used in a crystalline form, the nature of which can have a significant impact on their properties and behaviour. Moreover, it is sometimes possible for a given material to exist in more than one crystalline form, depending upon the conditions under which it was prepared. This is the phenomenon of *polymorphism*. This can be important because the various polymorphs may themselves have different properties. It is therefore of interest to be able to predict the three-dimensional atomic structure(s) that a given molecule may adopt, for those cases where it is difficult to obtain experimental data and also where one might wish to prioritise molecules not yet synthesised.

Many different approaches have been suggested as possible approaches to this problem, from the 1960s onwards [Verwer and Leusen 1998]. What is obvious from all of these efforts is that this is an extremely difficult problem Both thermodynamics and kinetics can be important in determining which crystalline form is obtained under a certain set of experimental conditions. Kinetic effects are particularly difficult to take into account and so are usually ignored. A proper treatment of the thermodynamic factors would require one to deal with the relative free energies of the different possible polymorphs.

These relative free energies are dependent upon internal energies, crystal densities and entropies:

$$\Delta G = \Delta U + P\Delta V - T\Delta S \tag{9.44}$$

Of these three contributions, that due to differences in density can safely be ignored (at least at normal pressures). The entropy differences are also invariably neglected, due to difficulties in calculating these contributions. This leaves the internal energy (at 0 K) as the metric by which the relative stabilities of polymorphs are predicted. In practice, the aim of most crystal structure prediction methods is to suggest a (hopefully small) number of solutions, according to their relative energies. These may subsequently be distinguished experimentally. For example, whilst it might prove impossible to obtain high-quality single-crystal diffraction data it may be feasible to acquire powder diffraction data, which, when combined with the computational results, can lead to a plausible solution.

When viewed purely as a search problem, one can readily appreciate its complexity. Not only should one consider the conformational flexibility of the molecule but one also has to suggest how these conformations might be able to pack into a low-energy structure. In addition, real crystals not only contain molecules of the compound of interest (sometimes in more than one conformation) but also solvent molecules and counterions, with the stoichiometry often being unknown prior to the experimental determination. One redeeming feature of the problem is that there are some well-established constraints on the way in which the molecules pack together, namely that the final structure must fall into one of the 230 space groups. In addition, in order to reduce the scale of the problem an algorithm may be limited to the more commonly occurring space groups and/or restricted to consider just one molecule in the asymmetric unit (see Section 3.8.1 for a brief description of some of these crystallographic terms).

Here we will consider just two of the more recent methods, which have much in common but also some significant differences. Gavezzotti's PROMET method [Gavezzotti 1991, 1994] starts by constructing clusters (called 'crystal nuclei') containing two molecules. The molecules are provided to the program in a predefined conformation that remains constant throughout the calculation. The relative locations of the molecules in these clusters are generated by applying common symmetry operators. Each symmetry operator can give rise to a number of possible cluster geometries, each of which is assessed by calculating its intermolecular energy. The most favourable clusters are selected for the subsequent steps, which may involve the application of additional symmetry operators or an attempt to construct a full crystal structure by translating the cluster to give a lattice in three dimensions. The intermolecular energy is again used to guide this process, which proceeds by building a sequence of clusters in one dimension, then two and finally three. Only if an improvement in the energy is achieved does the algorithm proceed to the next stage. In essence, the process is somewhat akin to a systematic search, but one which uses a variety of criteria to prune the search tree. In addition to the energetic criteria the growing lattice must also meet the condition that it has to belong to one of the known space groups. Further, analysis of known crystal structures reveals additional criteria that can be applied to restrict the ranges of some of the unit cell dimensions

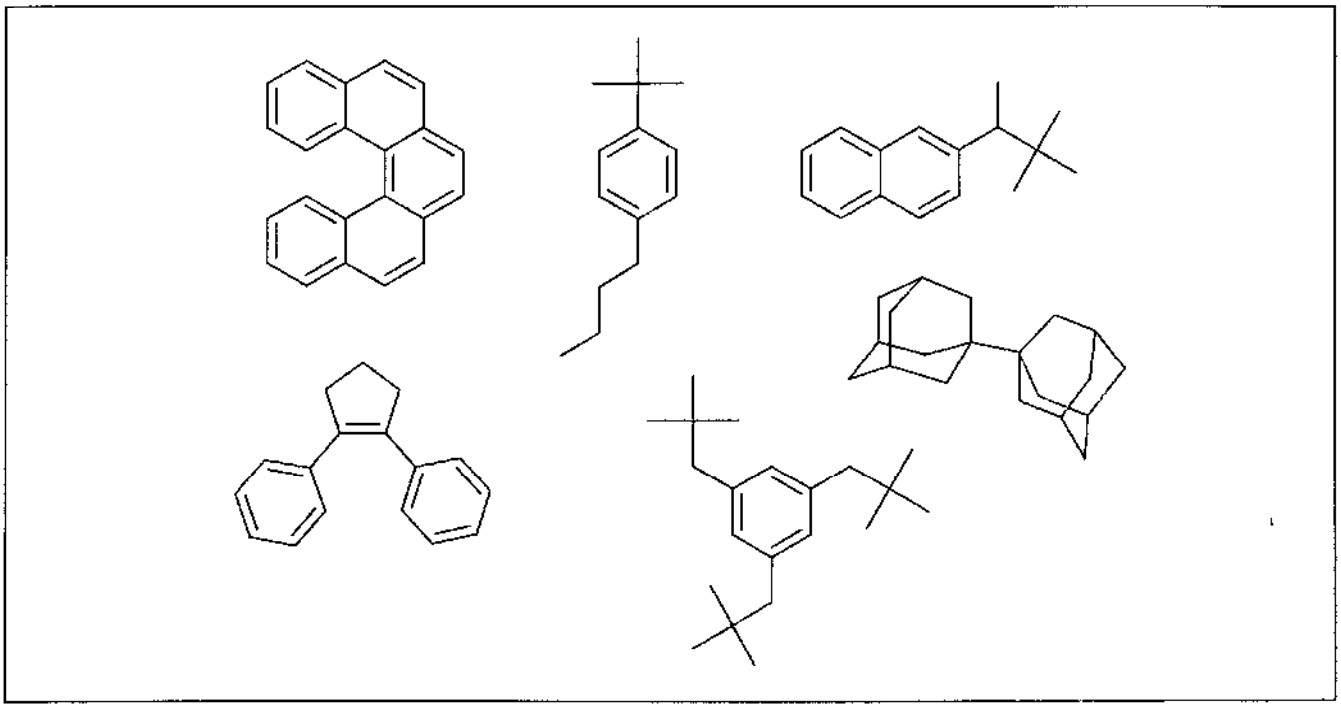


Fig. 9.35: Some typical molecules tested using the PROMET approach to predicting crystal structures

The PROMET method has been applied to a variety of systems, though at the time of writing it is limited to rigid molecules containing C, H, N, O, S and Cl atoms in the eleven space groups which account for more than 90% of organic crystal structures. Some of these molecules are shown in Figure 9.35. To try to simulate polymorph prediction several cases were identified from the literature where the structure of one polymorph was completely characterised and where mention was made of another polymorph for which only the cell parameters and space group could be determined [Gavezzotti and Filippini 1996]. In these cases the computational method was used to try to predict the complete structure of the unknown polymorph. In the small number of test cases examined the approach did prove quite successful, in that it was possible to identify low-energy packing arrangements that agreed with the experimental data. Of perhaps most interest was the fact that the energies of the various polymorphs were not in particularly good agreement with their relative stability but that the geometries were often rather well predicted. This points to a synergy between experiment and theory, in that when some experimental data were available (e.g. unit cell dimensions and space group) then one might be quite confident of being able to predict its structure.

The second type of approach we shall consider can be thought of as more '*ab initio*' in nature, in that there is no assumption that the lattice must be constructed from energetically favourable arrangements of molecules in the crystal nuclei. Rather, all packing arrangements in all possible space groups are generated. The following procedure is typically employed. Initially, molecules are placed into an oversized unit cell such that the symmetry relationships between the molecules are consistent with a particular space group. The molecules are then permitted to move, using either a random or systematic algorithm. This first phase can lead to a large number of trial structures (often several thousands), which are then clustered to identify duplicates The lowest-energy structure from each cluster is then minimised, followed by a final clustering. It can sometimes be appropriate to employ this final clustering before the

structures are properly and completely minimised, as it is usually considered important to use as complete a force field model as possible for this final step (for example, using Ewald summations and very tight convergence criteria).

Probably the main difference between the different variants lies in the way in which the molecules move in the first step. One approach is to use Monte Carlo simulated annealing

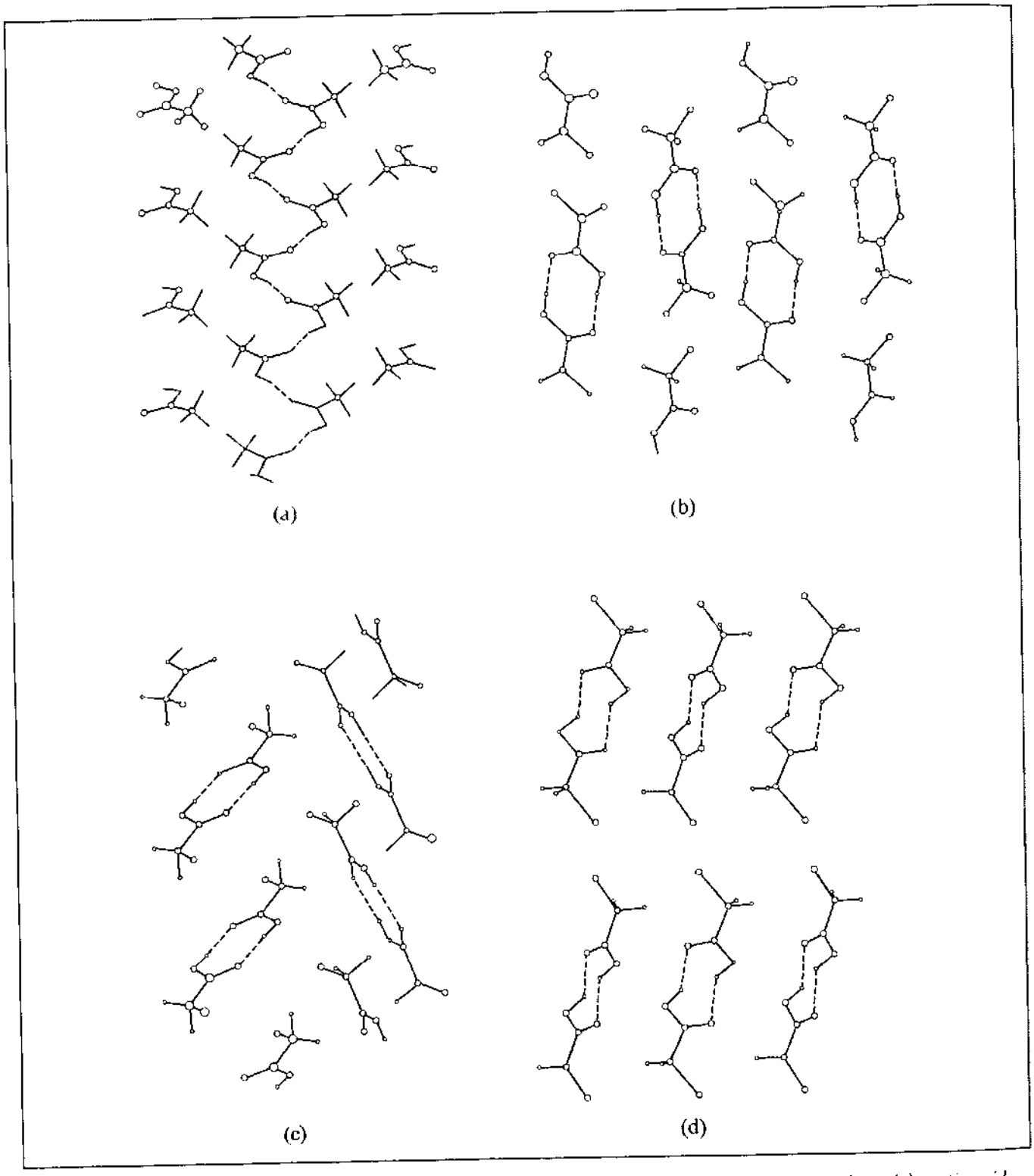


Fig. 9 36. Hydrogen bonding patterns in the crystal structures of acetic acid and its halo derivatives. (a) acetic acid, (b) fluoro acetic acid, (c) chloro acetic acid, (d) bromo acetic acid

[Gdanitz 1992; Karfunkel and Gdanitz 1992; Karfunkel *et al.* 1993; Leusen 1996] In this particular implementation it is the angular degrees of freedom that are varied during the Monte Carlo search. These angular variables comprise the cell angles, the Euler angles which describe the rigid-body rotations of the molecules in the cell, and the Euler angles which describe the actual location of the molecules in the cell. Having chosen a new set of angular parameters (or a subset of them) the translational parameters (which are the cell lengths and the distances between the molecules in the cell) are adjusted to relieve any close contacts that resulted. The new configuration is then accepted or rejected according to the Metropolis Monte Carlo criterion. A standard simulated annealing procedure is used, involving a slow decrease in temperature from several thousand kelvin to 300 K. This typically provides about 2000 accepted structures per space group for the next stage (clustering), from 4000–5000 Monte Carlo steps. An alternative to the Monte Carlo simulated annealing is to use a systematic search method. For example, one approach [van Eijck *et al.* 1995] involves a brute-force grid search in which the relevant parameters are varied systematically, with each trial structure being subjected to a few cycles of minimisation to locate approximately the nearest energy minimum.

One rather simple molecule which has been the subject of detailed comparisons of a number of methods is acetic acid. The reason for the interest in this molecule is that acetic acid (together with formic acid) adopts a chain-like structure, with each molecule forming one hydrogen bond with each of the two neighbouring molecules in the chain (Figure 9.36). Almost all other monocarboxylic acids form dimer-based structures, including fluoro, chloro and bromo acetic acid. Moreover, polymorphism has been detected for both chloro and bromo acetic acid. It is clearly of interest to be able to understand the reasons for this behaviour and also to determine whether it can be predicted by current computational methods. Acetic acid itself has been considered using both the grid search and the Monte Carlo simulated annealing approach, though it should also be noted that other aspects of the procedure differ as well (clustering algorithm, force field, minimisation method, etc.) [Mooij *et al.* 1998; Payne *et al.* 1998]. In both cases a number of low-energy structures were identified. These included the known crystal structure but also alternatives that were very similar in energy (i.e. dimers were found for acetic acid and chain structures for the halogenated derivatives). It thus appears that these methods are currently able to explore the search space quite effectively but the force fields currently used for the final assessment do not provide sufficient discrimination between the low-energy alternatives.

Further Reading

Aldenderfer M S and R K Blahfield 1984. *Cluster Analysis* Newbury Park, CA. Sage; New York, Garland Publishing.

Blaney J M and J S Dixon 1994. Distance Geometry in Molecular Modeling, In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 5. New York, VCH Publishers, pp. 299–335

Chatfield C and A J Collins 1980. *Introduction to Multivariate Analysis* London, Chapman & Hall

Desiraju G R 1997. Crystal Gazing: Structure Prediction and Polymorphism *Science* 278.404–405

Everitt B.S. 1993 *Cluster Analysis.* Chichester, John Wiley & Sons

Gavezzotti A (Editor) 1997. *Theoretial Aspects and Computer Modeling of the Molecular Solid State* Chichester, John Wiley & Sons.

Gavezzotti A 1998 The Crystal Packing of Organic Molecules: Challenge and Fascination below 1000Da. *Crystallography Reviews* 7:5–121

Leach A R 1991 A Survey of Methods for Searchung the Conformational Space of Small and Medium-Sized Molecules In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 2. New York, VCH Publishers, pp. 1–55.

Perutz M 1992. *Protein Structure New Approaches to Disease And Therapy.* New York, W H Freeman.

Scheraga H A 1993. Searching Conformational Space. In van Gunsteren W F, P K Weiner and A J Wilkinson (Editors) *Computer Simulation of Biomolecular Systems* Volume 2 Leiden, ESCOM.

Schulz G E and R H Schirmer 1979. *Principles of Protein Structure* New York, Springer-Verlag.

Torda A E and W F van Gunsteren 1992 Molecular Modeling Using NMR Data In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 2. New York, VCH Publishers, pp 143–172

Verwer P and F J J Leusen 1998. Computer Simulation to Predict Possible Crystal Polymorphs. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 12. New York, VCH Publishers, pp. 327–365.

References

Allen F H, S A Bellard, M D Brice, B A Cartwright, A Doubleday, H Higgs, T Hummelink, B G Hummelink-Peters, O Kennard, W D S Motherwell, J R Rodgers and D G Watson 1979. The Cambridge Crystallographic Data Centre· Computer-based Search, Retrieval, Analysis and Display of Information *Acta Crystallographica* **B35** 2331–2339

Allen F H, O Kennard, D G Watson, L Brammer, A G Orpen and R Taylor 1987 Tables of Bond Lengths Determined by X-ray and Neutron Diffraction 1 Bond Lengths in Organic Compounds. *Journal of the Chemical Society Perkin Transactions* II S1–S19.

Barton D H R 1950. The Conformation of the Steroid Nucleus *Experientia* **6** 316–320.

Bergerhoff G, R Hundt, R Sievers and I S Brown 1983. The Inorganic Crystal Structure Database *Journal of Chemical Information and Computer Sciences* **23**:66–69.

Berman H M, J Westbrook, Z Feng, G Gilliland, T N Bhat, H Weissig, I N Shindyalor and P E Bourne 2000. The Protein Data Bank. *Nucleic Acids Research* **28** 235–242.

Bernstein F C, T F Koetzle, G J B Williams, E Meyer, M D Bryce, J R Rogers, O Kennard, T Shikanouchi and M Tasumi 1977. The Protein Data Bank: A Computer-based Archival File for Macromolecular Structures *Journal of Molecular Biology* **112**:535–542

Brunger A T, J Kuriyan and M Karplus 1987. Crystallographic R-factor Refinement by Molecular Dynamics. *Science* **235**·458–460.

Bruno I J, J C Cole, J P M Lommerse, R S Rowland, R Taylor and M L Verdonk 1997. Isostar: A Library of Information about Nonbonded Interactions. *Journal of Computer-Aided Molecular Design* 11·525–537.

Chang G, W C Guida and W C Still 1989. An Internal Coordinate Monte Carlo Method for Searching Conformational Space *Journal of the American Chemical Scociety* **111**:4379–4386

Chatfield C and A J Collins 1980. *Introduction to Multivariate Analysis.* London, Chapman & Hall

Chung C-W, R M Cooke, A E I Proudfoot and T N C Wells 1995 The Three-dimensional Structure of RANTES *Biochemistry* **34** 9307–9314

Clark D E and D R Westhead 1996 Evolutionary Algorithms in Computer-aided Molecular Design *Journal of Computer-Aided Molecular Design* **10**:337–358

Crippen G M 1981 *Distance Geometry and Conformational Calculations* Chemometrics Research Studies Series 1 New York, John Wiley & Sons

Crippen G M and T F Havel 1988 *Distance Geometry and Molecular Conformation* Chemometrics Research Studies Series 15 New York, John Wiley & Sons

Derome A E 1987. *Modern NMR Techniques for Chemistry Research.* Oxford, Pergamon.

Downs G M, P Willett and W Fisanick 1994 Similarity Searching and Clustering of Chemical Structure Databases using Molecular Property Data *Journal of Chemical Information and Computer Sciences* **34**:1094–1102

Ferguson D M and D J Raber 1989. A New Approach to Probing Conformational Space with Molecular Mechanics Random Incremental Pulse Search *Journal of the American Chemical Society* **111**:4371–4378.

Ferro D R and J Hermans 1977. A Different Best Rigid-body Molecular Fit Routine. *Acta Crystallographica* **A33**:345–347

Gavezzotti A 1991 Generation of Possible Crystal Structures from the Molecular Structure for Low-polarity Organic Compounds *Journal of the American Chemical Society* **113**.4622–4629

Gavezzotti A 1994 Are Crystal Structures Predictable? *Accounts of Chemical Research* **27**.309–314

Gavezzotti A and G Filippini 1996. Computer Prediction of Organic Crystal Structures Using Partial X-ray Diffraction Data *Journal of the American Chemical Society* **118**:7153–7157

Gdanitz, R J 1992. Prediction of Molecular Crystal Structures by Monte Carlo Simulated Annealing Without Reference to Diffraction Data *Chemical Physics Letters* **190**:391–396

Gibson K D and H A Scheraga 1987 Revised Algorithms for the Build-up Procedure for Predicting Protein Conformations by Energy Minimization *Journal of Computational Chemistry* **8**:826–834.

Glusker J P 1995. Intermolecular Interactions Around Functional Groups in Crystals. Data for Modeling the Binding of Drugs to Biological Macromolecules *Acta Crystallographica* **D51**:418–427

Goldberg D E 1989. *Genetic Algorithms in Search, Optimization and Machine Learning.* Reading, MA., Addison-Wesley

Goodman J M and W C Still 1991 An Unbounded Systematic Search of Conformational Space. *Journal of Computational Chemistry* **12**.1110–1117

Jack A and M Levitt 1978. Refinement of Large Structures by Simultaneous Minimization of Energy and R-factor *Acta Crystallographica* **A34**:931–929

Jarvis R A and E A Patrick 1973 Clustering Using a Similarity Measure Based on Shared Near Neighbours. *IEEE Transactions in Computers* **C-22**:1025–1034.

Jones G 1998. Genetic and Evolutionary Algorithms. In Schleyer, P v R, N L Allinger, T Clark, J Gasteiger, P A Kollman, H F Schaefer III and P R Schreiner (Editors) *The Encyclopedia of Computational Chemistry.* Chichester, John Wiley & Sons.

Jones T A and S Thirup 1986 Using Known Substructures in Protein Model Building and Crystallography. *EMBO Journal* **5**.819–822

Judson R 1997, Genetic Algorithms and Their Use in Chemistry. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 10. New York, VCH Publishers, pp. 1–73

Judson R S, W P Jaeger, A M Treasurywala and M L Peterson 1993 Conformational Searching Methods for Small Molecules 2 Genetic Algorithm Approach. *Journal of Computational Chemistry* **14**:1407–1414.

Kabsch W 1978. A Discussion of the Solution for the Best Rotation to Relate Two Sets of Vectors *Acta Crystallographica* **A34**.827–828.

Karfunkel H R and R J Gdanitz 1992. *Ab initio* Prediction of Possible Crystal Structures for General Organic Molecules. *Journal of Computational Chemistry* **13**:1171–1183

Karfunkel H R, B Rohde, F J J Leusen, R J Gdanitz, and G Rihs 1993. Continuous Similarity Measure Between Nonoverlapping X-ray Powder Diagrams of Different Crystal Modifications *Journal of Computational Chemistry* **14**.1125–1135

Kirkpatrick S, C D Gelatt and M P Vecchi 1983. Optimization by Simulated Annealing *Science* **220** 671–680.

Kolossváry I and W C Guida 1996 Low Mode Search An Efficient, Automated Computational Method for Conformational Analysis· Application to Cyclic and Acyclic Alkanes and Cyclic Peptides *Journal of the American Chemical Society* **118**:5011–5019.

Lance G N and W T Williams 1967. A General Theory of Classificatory Sorting Strategies 1. Hierarchical Systems *Computer Journal* **9**·373–380.

Leach A R, D P Dolata and K Prout 1990. Automated Conformational Analysis and Structure Generation: Algorithms for Molecular Perception. *Journal of Chemical Information and Computer Science* **30**:316–324.

Leach A R, K Prout and D P Dolata. 1988. An Investigation into the Construction of Molecular Models using the Template Joining Method. *Journal of Computer-Aided Molecular Design* **2**·107–123.

Leusen F J L 1996 *Ab initio* Prediction of Polymorphs. *Journal of Crystal Growth* **166**:900–903.

Li Z Q and H A Scheraga 1987. Monte-Carlo-minimization Approach to the Multiple-minima Problem in Protein Folding. *Proceedings of the National Academy Of Sciences USA* **84**:6611–6615.

McGarrah D B and R S Judson 1993. Analysis of the Genetic Algorithm Method of Molecular-conformation Determination. *Journal of Computational Chemistry* 14:1385–1395

Meza J C, R S Judson, T R Faulkner and A M Treasurywala 1996. A Comparison of a Direct Search Method and a Genetic Algorithm for Conformational Searching. *Journal of Computational Chemistry* **17**:1142–1151.

Mooij W T M, B P van Eijck, S L Price, P Verwer and J Kroon 1998. Crystal Structure Predictions for Acetic Acid *Journal of Computational Chemistry* **19**:459–474.

Murray-Rust P M and J P Glusker 1984 Directional Hydrogen Bonding to sp^2 and sp^3-hybridized Oxygen Atoms and Its Relevance to Ligand–Macromolecule Interactions. *Journal of the American Chemical Society* **106**·1018–1025

Payne R S, R J Roberts, R C Crowe and R Docherty 1998. Generation of Crystal Structures of Acetic Acid and Its Halogenated Analogs. *Journal of Computational Chemistry* **19**:1–20

Ramachadran G N, C Ramakrishnan and V Sasiekharan 1963. Stereochemistry of Polypeptide Chain Configurations *Journal of Molecular Biology* **7**:95–99

Saunders M 1987 Stochastic Exploration of Molecular Mechanics Energy Surface: Hunting for the Global Minimum *Journal of the American Chemical Society* **109**:3150–3152

Saunders M, K N Houk, Y-D Wu, W C Still, M Lipton, G Chang and W C Guida 1990 Conformations of Cycloheptadecane A Comparison of Methods for Conformational Searching. *Journal of the American Chemical Society* **112** 1419–1427

Smellie A S, S D Kahn and S L Teig 1995a Analysis of Conformational Coverage 1. Validation and Estimation of Coverage. *Journal of Chemical Information and Computer Science* **35**:285–294

Smellie A S, S L Teig and P Towbin 1995b Poling: Promoting Conformational Variation. *Journal of Computational Chemistry* **16**:171–187.

Torda A E, R M Scheek and W F van Gunsteren 1990 Time-averaged Nuclear Overhauser Effect Distance Restraints Applied to Tendamistat. *Journal of Molecular Biology* **214**:223–235.

Van Eijck B P, W T M Mooij and J Kroon 1995. Attempted Prediction of the Crystal Structures of Six Monosaccharides. *Acta Crystallographica* **B51**:99–103.

Verwer P and F J L Leusen 1998. Computer Simulation to Predict Possible Crystal Polymorphs In Lipkowitz K B and Boyd D B (Editors) *Reviews in Computational Chemistry* New York, Wiley-VCH, pp. 327–365

Ward J H 1963. Hierarchical Grouping to Optimise an Objective Function *American Statistical Association Journal*: 236–244.

Weiner S J, P A Kollman, D A Case, U C Singh, C Ghio, G Alagona, S Profeta and P Weiner 1984 A New Force Field for Molecular Mechanical Simulation of Nucleic Acids and Proteins. *Journal of the American Chemical Society* **106**:765–784.

CHAPTER TEN

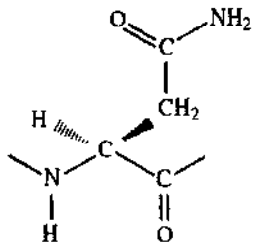
Protein Structure Prediction, Sequence Analysis and Protein Folding

10.1 Introduction

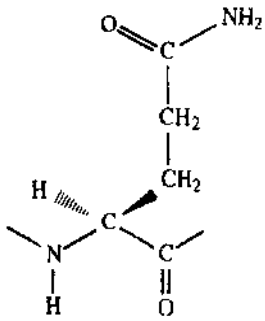
Peptides and proteins are polymers constructed from sequences of amino acids. They perform many functions essential to life. There are twenty common naturally occurring amino acids, shown in Figure 10.1. The amino acids are linked together via amide bonds to give a polypeptide chain. All the naturally occurring amino acids have the same relative stereochemistry at the alpha-carbon (referred to as 'L'). The side chains have different sizes, shapes, hydrogen-bonding capabilities and charge distributions, which enable proteins to display the vast array of biological functions required by living systems.

Protein biosynthesis is a very complex process. The amino acid sequence of a protein is determined by the DNA sequence of the corresponding gene. Each amino acid is coded by three adjacent DNA bases. However, DNA is not used directly in protein biosynthesis. Rather, an RNA copy is made from the DNA template; this *messenger RNA* (mRNA) in turn acts as the template for the protein synthesis. This process is known as *transcription*. In the subsequent *translation* step, the mRNA template is read by transfer RNA (tRNA), which also brings the actual amino acids to the site of synthesis. This two-stage, unidirectional flow of genetic information was proposed by Francis Crick and is known as the 'Central Dogma'. It is often represented by the diagram shown in Figure 10.2(a). Some modifications were required to the theory following the discovery of retroviruses, which can transfer genetic information from RNA to DNA (the dotted lines in Figure 10.2(b)) but the Central Dogma as originally proposed still holds true for most organisms.

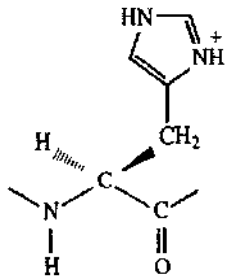
The biological function of a protein or peptide is often intimately dependent upon the conformation(s) that the molecule can adopt. In contrast to most synthetic polymers where the individual molecules can adopt very different conformations, a protein usually exists in a single native state. These native states are found under conditions typically found in living cells (aqueous solvents near neutral pH at 20–40°C). Proteins can be unfolded (or *denatured*) using high-temperature, acidic or basic pH or certain non-aqueous solvents. However, this unfolding is often reversible and so proteins can be folded back to their native structure in the laboratory.



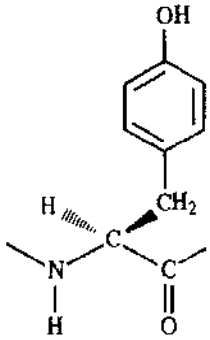
Asparagine
Asn
N



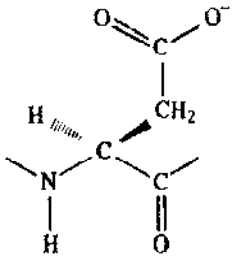
Glutamine
Gln
Q



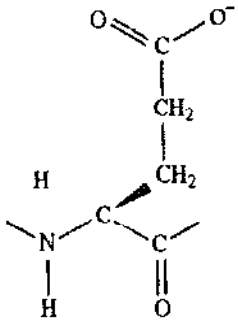
Histidine
His
H



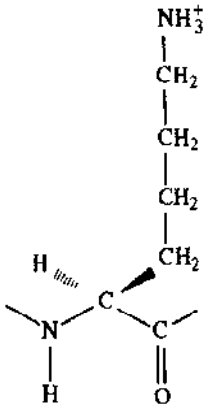
Tyrosine
Tyr
Y



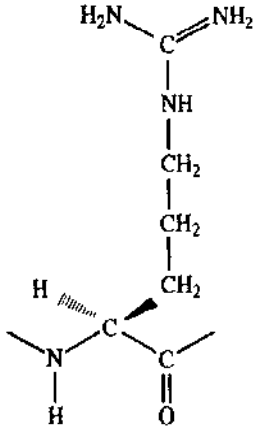
Aspartic acid
Asp
D



Glutamic acid
Glu
E

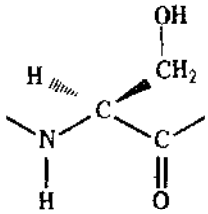


Lysine
Lys
K

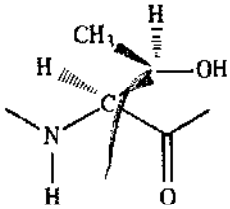


Arginine
Arg
R

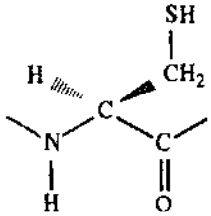
Fig 10.1· The twenty naturally occurring amino acids and their codes.



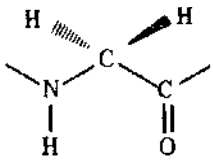
Serine
Ser
S



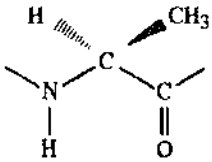
Threonine
Thr
T



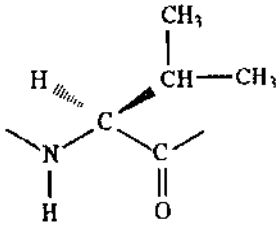
Cysteine
Cys
C



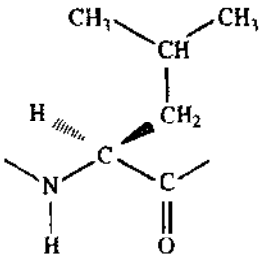
Glycine
Gly
G



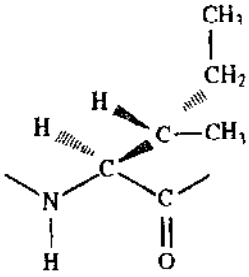
Alanine
Ala
A



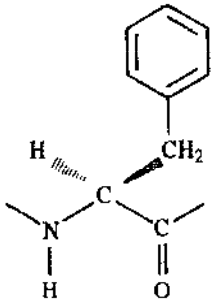
Valine
Val



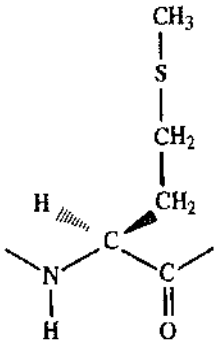
Leucine
Leu
L



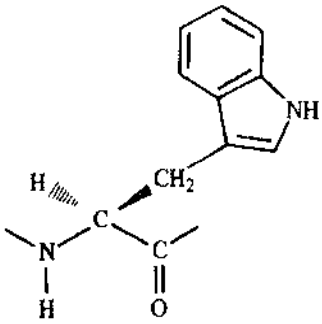
Isoleucine
Ile
I



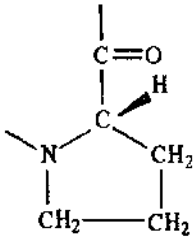
Phenylalanine
Phe
F



Methionine
Met
M



Tryptophan
Trp
W



Proline
Pro
P

Fig 10 1· Continued

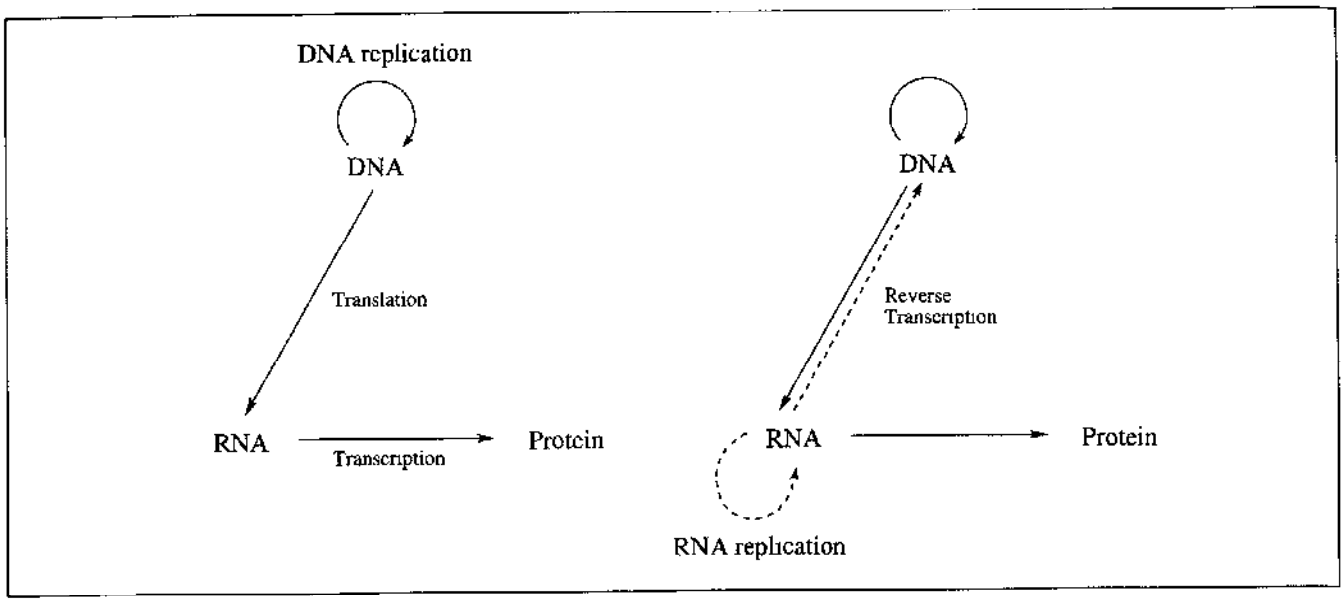


Fig 10.2· The original Central Dogma of molecular biology (left) and its modification in light of the discovery of retroviruses (right)

X-ray crystallography and NMR are the methods most widely used to provide detailed information about protein structures. Unfortunately, the rate at which new protein sequences are being determined far exceeds the rate at which protein structures are determined experimentally. This is a particularly pertinent problem due to the efforts of the Human Genome Project, which is expected to have sequenced the entire human genome by 2003, if not earlier. It will then be necessary to determine the amino acid sequences of the proteins that are encoded by the DNA. This is not quite so straightforward as might be imagined due to the complex nature of the transcription/translation process and also because experimental sequencing methods do not deliver single 'genes'. Moreover, not all DNA actually codes for protein and in many genes the biological information is contained in distinct units called *exons* (the intervening non-coding regions being *introns*). *Functional genomics* is concerned with characterising the proteins expressed by the genome and assigning a biological function. It is possible to some extent to assign function just from an analysis of the sequence alone. However, the intimate relationship between the three-dimensional structure and function of a protein makes functional assignment based upon the structure more appealing (sometimes known as *structural genomics**). The general difficulties in obtaining protein structures using experimental techniques means that there is considerable interest in theoretical methods for predicting the three-dimensional structure of proteins from the amino acid sequence: this is often referred to as the *protein folding problem*. The results of a sequence analysis or structure prediction

* Functional genomics and structural genomics are widely used (and abused) terms. They are sometimes considered to apply only to large-scale, high-throughput technologies. Moreover, although computational methods have an important role to play, input from experimental techniques can also be critical An example of this was the use of X-ray crystallography to predict the function of a protein from a hyperthermophile bacterium, *Methanococcus jannaschii* [Zarembinski *et al.* 1998]. The crystal structure had ATP bound, suggesting that the function of the protein was either an ATPase or an ATP-binding molecular switch, subsequently confirmed by experiments The protein did have structural similarity to other proteins but in this particular case the information was not functionally useful.

are often referred to as *annotations*. This is a generic term used to describe additional information that is attached to the sequence or structure, such as key sequence or structural features, the location of catalytic residues or a proposed function.

Bioinformatics is a relatively new discipline that is concerned with the collection, organisation and analysis of biological data. It is beyond our scope to provide a comprehensive overview of this discipline; a few textbooks and reviews that serve this purpose are now available (see the suggestions for further reading). However, we will discuss some of the main methods that are particularly useful when trying to predict the three-dimensional structure and function of a protein. To help with this, Appendix 10.1 contains a limited selection of some of the common abbreviations and acronyms used in bioinformatics and Appendix 10.2 lists some of the most widely used databases and other resources.

In the rest of this chapter we will first introduce some of the key principles of protein structure and then discuss a number of approaches to tackling the protein folding problem. Another area that we will consider is protein folding: how a protein manages to fold into its own unique three-dimensional structure. A number of experimental and theoretical techniques can be used to investigate protein folding, which has led to a greater understanding of this phenomenon.

10.2 Some Basic Principles of Protein Structure

The first X-ray structures revealed that proteins did not adopt regular or symmetrical structures but were much more complex. However, certain structural motifs were observed to occur frequently. The most common motifs are the α-helix and the β-strand, shown in Figure 10.3. These constitute the *secondary structure* of a protein (the primary structure being the amino acid sequence and the tertiary structure the detailed three-dimensional conformation). Linus Pauling predicted that the α-helix would be a stable element of polypeptide structure well before the first protein structure was solved [Pauling *et al.* 1951]. His prediction was based upon mechanical models constructed after a careful analysis of the geometry of the peptide unit in crystal structures of small molecules and can be considered a classic example of the predictive power of molecular modelling. Another type of helix, the 3_{10} helix, is also found infrequently. The β-strands often form extended structures called β-sheets in which the strands are hydrogen-bonded to each other In a β-sheet the strands can run in either parallel or anti-parallel directions, as shown in Figure 10.4. Secondary structural elements are connected by regions often referred to as 'loops', which adopt less regular structures. Nevertheless, common conformations can be identified in certain types of loop structure, such as conformations that are commonly adopted by the 'β-turn' regions between β-strands [Wilmot and Thornton 1988]

If we ignore the small variations in bond angles and bond lengths then the conformation of an amino acid residue in a protein or peptide can be classified according to the torsion angles of its rotatable bonds. There are three backbone torsion angles, labelled ϕ, ψ and ω (Figure 10.5). The conformations of the side chains are characterised by the torsion angles χ_1, χ_2, etc. The amide bond has a relatively high energy barrier for rotation away from

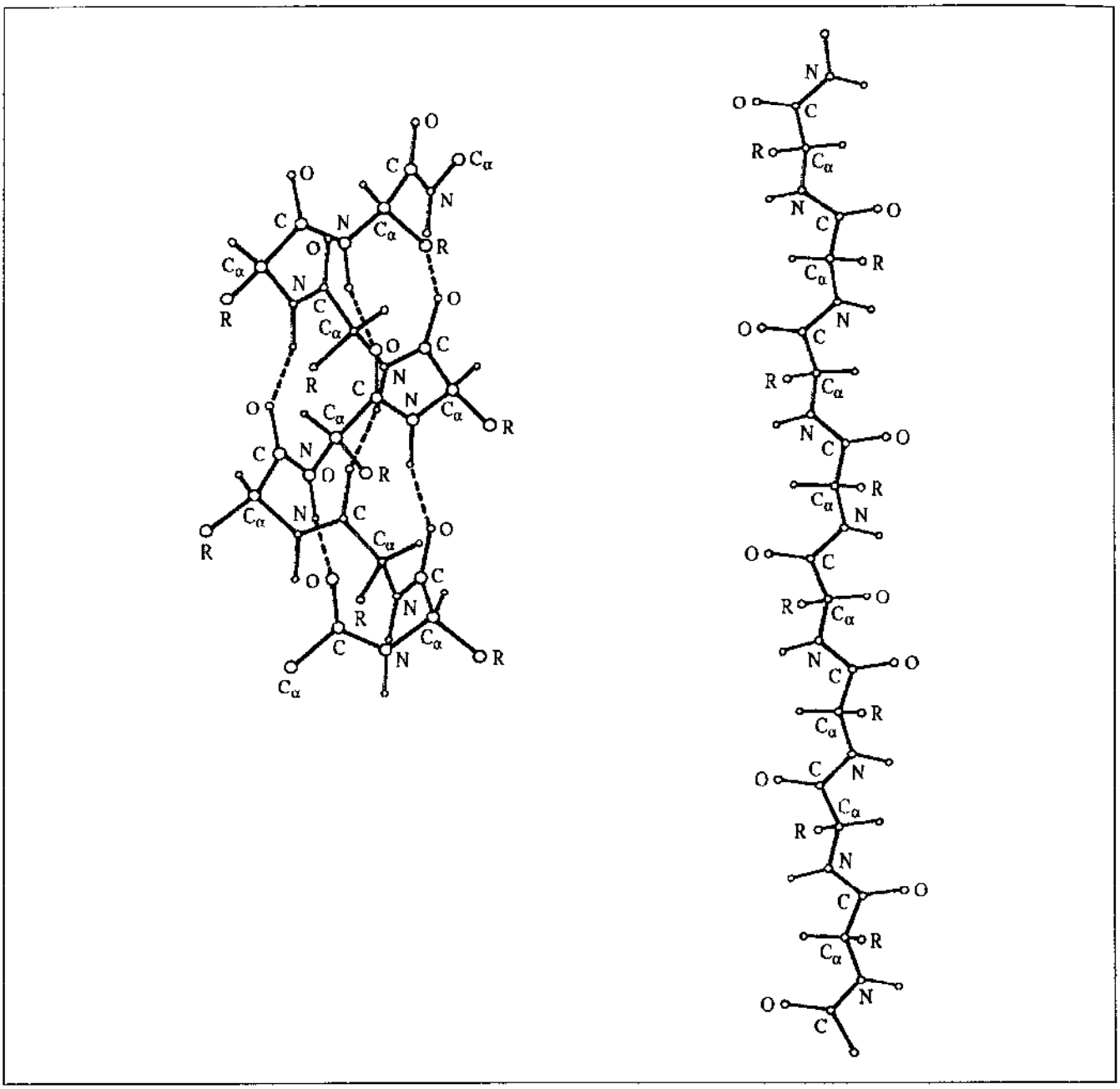


Fig. 10.3: The α-helix and β-strand structures

planarity and so ω rarely deviates significantly from 0° or 180°. Moreover, there is a significant preference for the *trans* ($\omega = 180°$) conformation (except for proline, which shows a relatively high proportion of *cis*-peptide linkages). We have already noted the contribution of Ramachandran to our understanding of protein structure in our discussion of conformational analysis (Section 9.2 and Figure 9.3). Examination of the X-ray structures of proteins shows that most amino acids occupy one of the low-energy regions in the Ramachandran contour map. Indeed, it is now common practice, when assessing an X-ray or NMR structure determination, to construct its Ramachandran map and to examine closely any residues which adopt a conformation outside the preferred regions The side chains also tend to adopt preferred conformations, though there are many examples of unusual or higher-energy structures [Ponder and Richards 1987]. Subsequent investigations have revealed

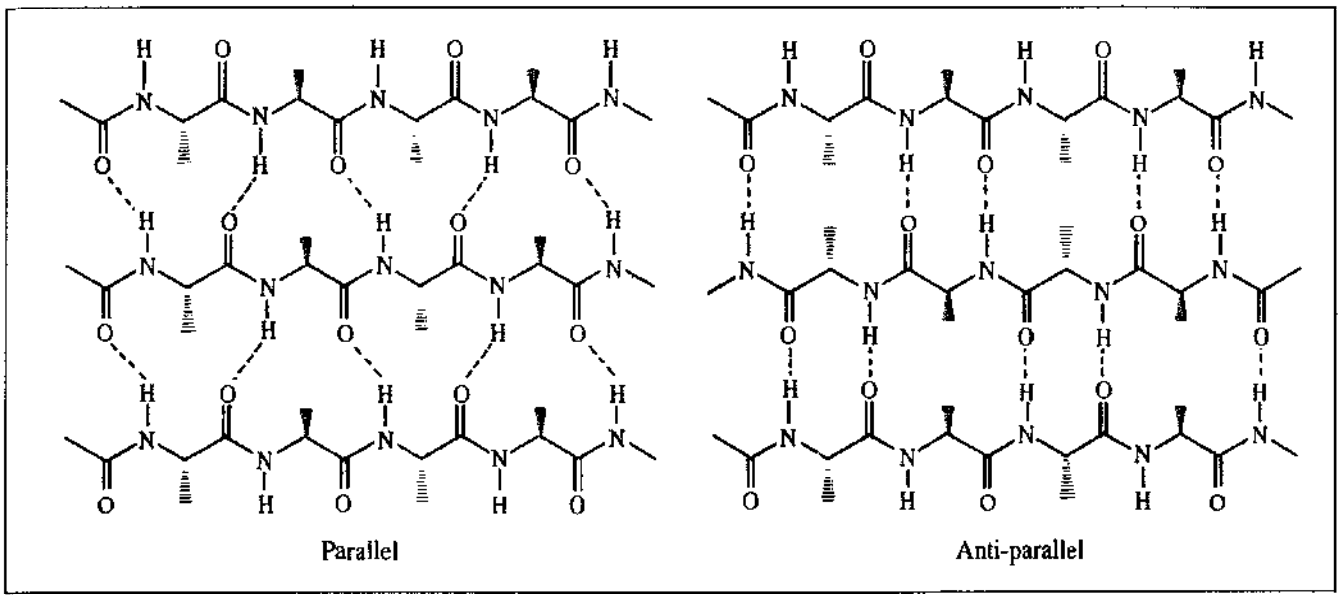


Fig. 10.4: The formation of parallel and anti-parallel β-sheets

that the side-chain conformations are often correlated with the backbone structure; for certain conformations of the backbone only particular side-chain structures are possible [Summers *et al.* 1987; Dunbrack and Karplus 1993].

As more protein structures became available it was observed that some contained more than one distinct region, with each region often having a separate function. Each of these regions is usually known as a *domain*, a domain being defined as a polypeptide chain that can fold independently into a stable three-dimensional structure.

10.2.1 The Hydrophobic Effect

Water-soluble globular proteins usually have an interior composed almost entirely of non-polar, hydrophobic amino acids such as phenylalanine, tryptophan, valine and leucine with polar and charged amino acids such as lysine and arginine located on the surface of the molecule. This packing of hydrophobic residues is a consequence of the *hydrophobic effect*, which is the most important factor that contributes to protein stability. The molecular basis for the hydrophobic effect continues to be the subject of some debate but is generally considered to be entropic in origin. Moreover, it is the entropy change of the solvent that is

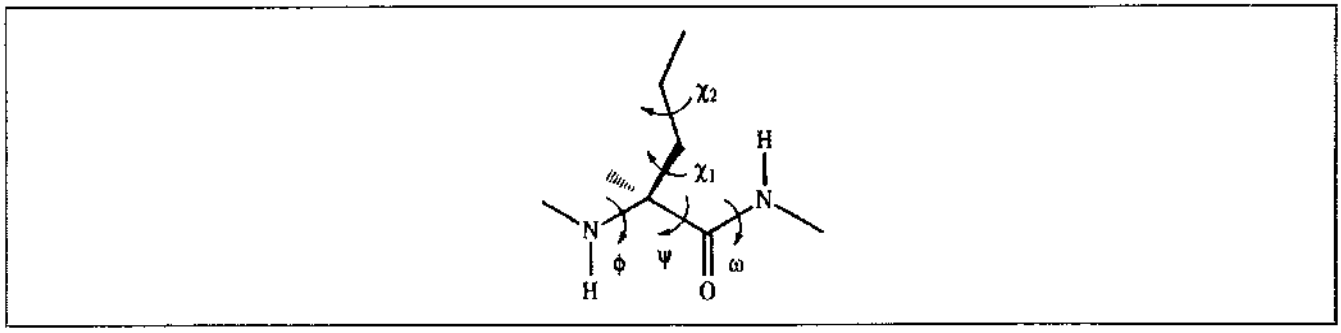


Fig. 10.5: The torsion angles that define the conformation of an amino acid

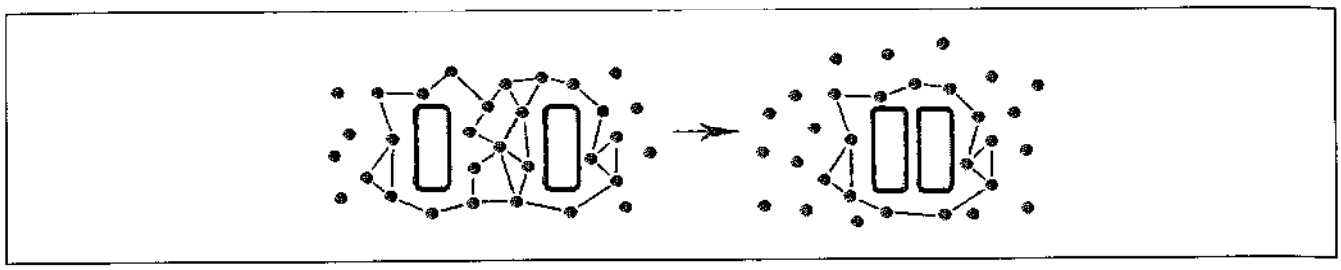


Fig 10 6: The hydrophobic effect Water molecules around a non-polar solute form a cage-like structure, which reduces the entropy When two non-polar groups associate, water molecules are liberated, increasing the entropy

important. The contribution to the overall free energy of folding due to the packing of the non-polar amino acids is positive (i.e. unfavourable) on both enthalpic and entropic grounds. The enthalpy change to remove the non-polar amino acids from water is positive due to dipole/induced-dipole electrostatic interactions between the polar water molecules and the hydrocarbon side chains. When packed, there are only (weaker) dispersion interactions between the side chains. The entropy change associated with packing the amino acids is negative because the unfolded state is less ordered than the packed state (for example, more conformational degrees of freedom are accessible). Water molecules are believed to form a cage-like structure around a non-polar solute, which has been likened to a local 'iceberg'. The water molecules in this region are locally ordered with most of the hydrogen bonding network of pure water intact. The area of the non-polar interface is much larger for the unfolded protein (Figure 10 6) and so the entropy change of the water when the protein folds is large The enthalpy change of the water is negative as the disruption to the hydrogen bonding network is less for the folded protein. Of these four contributions, the two enthalpy terms are believed to be small, with the entropy change associated with the ordering of the solvent water molecules being the dominant term. This is just one possible model for the hydrophobic effect; unfortunately, experimental data (e.g. from X-ray crystallography or NMR) are scarce. It is also worth noting that preliminary molecular dynamics simulations were unable to find any evidence for the enhancement of water structure at a hydrophobic protein interface [Kovacs *et al.* 1997].

Not all proteins are water soluble; a very important class is the membrane-bound proteins, which include receptors and ion channels. The arrangement of the amino acids in these proteins is very different in the membrane-spanning regions. The membrane provides a very hydrophobic environment and so hydrophobic residues are often located on the outside, towards the membrane. It is very difficult to obtain X-ray crystal structures of membrane-bound proteins due to the problems of obtaining satisfactory crystals. The crystal structure of the photosynthetic reaction centre, which earned Michel, Deisenhofer and Huber the Nobel Prize in 1988, was obtained after much painstaking work in which the protein was crystallised from a detergent solution. Electron microscopy has been used to determine the structures of membrane-bound proteins; in favourable cases, the resolution of this technique approaches that of X-ray crystallography but is usually much lower. Henderson and Unwin have pioneered the application of this method to membrane proteins with their determination of the structures of bacteriorhodopsin and rhodopsin [Henderson *et al.* 1990; Havelka *et al.* 1995]. Both of these proteins contain seven *trans*-membrane helices, which are connected by loops in the extracellular and intracellular regions.

No universal solution has yet been found to the protein folding problem, but a variety of promising approaches have been developed. In the next sections we shall consider some of these methods for predicting the structures of proteins and peptides. Our discussion will first consider methods that attempt to predict the structures of proteins from first principles. We will then discuss methods that use a stepwise approach, in which elements of secondary structure are first identified and then these elements are packed together. Finally, we will consider the prediction of protein structures by homology modelling (sometimes referred to as comparative modelling), where the structure of the unknown protein is based upon the known structure(s) of related (i.e. homologous) protein(s). As part of this we will also describe some of the methods that can be used for sequence analysis.

10.3 First-principles Methods for Predicting Protein Structure

The most ambitious approaches to the protein folding problem attempt to solve it from first principles (*ab initio*). As such, the problem is to explore the conformational space of the molecule in order to identify the most appropriate structure. The total number of possible conformations is invariably very large and so it is usual to try to find only the very lowest energy structure(s). Some form of empirical force field is usually used, often augmented with a solvation term (see Section 11.12). The global minimum in the energy function is assumed to correspond to the naturally occurring structure of the molecule.

All of the conformational search methods that were described in Sections 9.2–9.7 have been used at some stage to explore the conformational space of small peptides. Here we will describe some of the methods designed specifically for tackling the problem for peptides and proteins.

H A Scheraga has devised many novel methods with his colleagues for exploring the conformational space of peptides and proteins [Scheraga 1993]. Each new method is rigorously evaluated using a standard test molecule, met-enkephalin (H–Tyr–Gly–Phy–Met–OH). One method is the 'build-up' approach, in which the peptide is constructed from three-dimensional amino acid templates [Gibson and Scheraga 1987]. Each template corresponds to a low-energy region of the Ramachandran map. To explore the conformational space of a peptide, a dipeptide fragment is first constructed by joining together all possible pairs of templates available to the first two amino acids. Each dipeptide fragment is minimised and the lowest-energy structures are retained for the next step, in which the third amino acid is connected. The peptide is gradually built up in this way, with energy minimisation and selection of the lowest-energy structures at each stage.

The simplest of Scheraga's random search methods is a random dihedral search, in which a single dihedral is selected at each iteration and randomly rotated [Li and Scheraga 1987]. The resulting structure is minimised and then accepted or rejected according to the Metropolis criterion. Such an algorithm is equally applicable to organic molecules as to proteins. The electrostatically driven Monte Carlo method [Ripoll and Scheraga 1988, 1989] is a more complex random search method which recognises the importance of long-range electrostatic interactions in polypeptides and proteins. It is based upon the observation that the local

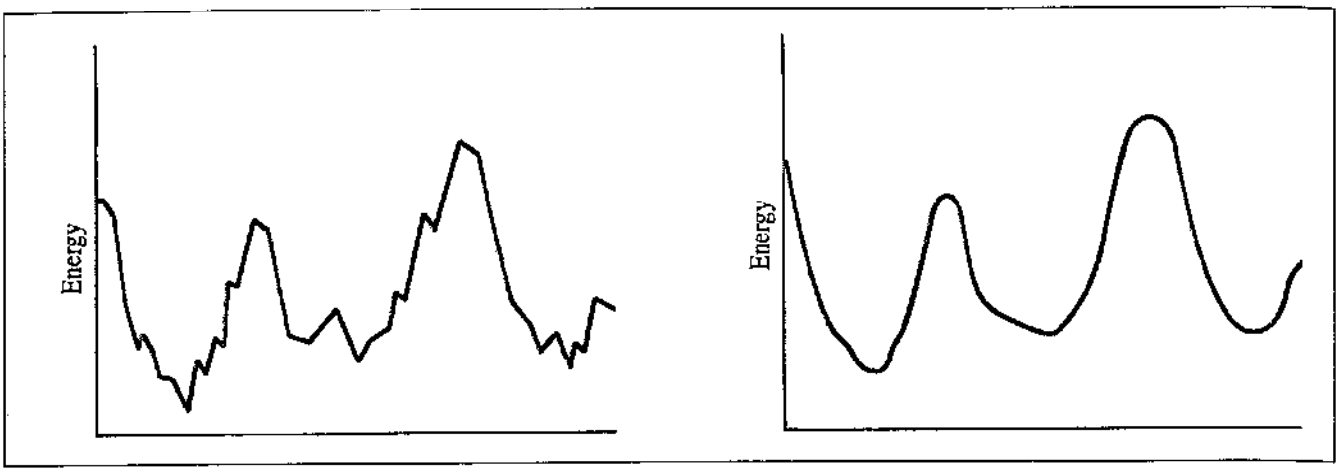


Fig. 10.7: Schematic energy surfaces for 'all-atom' (left) and simplified (right) models.

dipoles of amide units often adopt a favourable alignment in the electrostatic field of the protein. Two different types of move are thus used in the scheme. In the first type of move, an amide unit is randomly selected and the backbone (ϕ, ψ) torsion angles are changed to enable its dipole to be optimally aligned in its local electrostatic field. The resulting conformation is minimised and then accepted or rejected according to the Metropolis criterion. The second type of move involves a random change to a randomly selected dihedral followed by minimisation and acceptance or rejection in the usual way. This approach thus combines moves designed to optimise the long-range electrostatic interactions with moves that have a more local influence on the conformation.

Proteins have many more rotatable bonds than peptides, and so it is common to use some form of simplified energy model to make the problem tractable. The energy surface of a model with fewer degrees of freedom should have a smaller number of minima than the energy surface of a more detailed model; it must be assumed that the energy surface of the simplified model reproduces the general features of the more detailed representation, but without the fine structure (Figure 10. 7). Various simplified models have been developed for investigating the conformational space of proteins. Many of these models are analogous to the models used to perform Monte Carlo simulations of polymers, such as the lattice and 'bead' models. An optimisation procedure based on molecular dynamics/simulated annealing or a genetic algorithm is often used with such simplified models to first identify families of low-energy structures, which may then be converted into a more detailed representation for subsequent refinement.

10.3.1 Lattice Models for Investigating Protein Structure

One reason for the interest in lattice models is that they can be used to try to answer some of the fundamental questions about protein structure. For example, it may be feasible to enumerate all possible conformations for a chain of a given length on the lattice From this set of states statistical mechanics can be used to derive thermodynamic properties and to investigate the relationship between the structure and the sequence. In the 'HP' model [Chan and Dill 1993], a protein is modelled as a sequence of hydrophobic (H) and

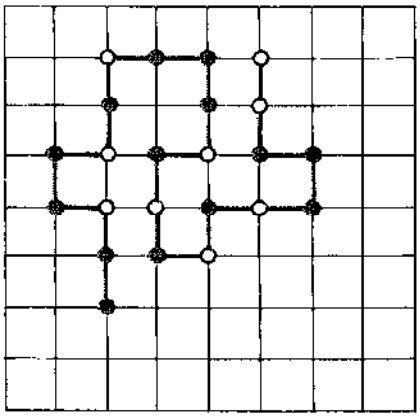


Fig 10.8. The HP model of Chan and Dill.

hydrophilic (P) monomers. The sequence is grown onto a two-dimensional lattice using a self-avoiding walk, and the energy of the resulting conformation is calculated by summing interactions between pairs of monomers that occupy adjacent lattice sites but are not covalently bonded (Figure 10.8). Such interactions between pairs of hydrophobic monomers are favoured by a constant energy increment with all other interaction energies set to zero. Exhaustive enumeration of all the conformations was possible for chains of 30 or so monomers, and the global energy minimum for each chain was determined Several interesting features arose from studies of this model. When the hydrophobic–hydrophobic interaction energy is small a large number of conformations are accessible. As the hydrophobic interaction energy increases there is a sharp decrease in the number of compact conformations containing hydrophobic cores. Another interesting feature of this and similar models is that α-helices and β-sheets naturally arise in the compact cores of such models. This suggests that the formation of secondary structure in a protein is not driven by specific hydrogen-bonding interactions between amino acids but rather by the compact nature of the core; conformations other than helices or sheets are not viable.

The simple lattice models are intended to address general questions about protein folding and structure. More sophisticated lattice models have been designed which are used to predict the actual structures of specific proteins. With such methods it is usually not possible to exhaustively explore the conformational space even on the lattice and so methods such as Monte Carlo simulated annealing are used to generate low-energy structures. Skolnick has developed several lattice models, which are used in a three-stage procedure to construct a model of the protein [Godzik *et al.* 1993; Skolnick *et al.* 1997]. In the first stage, a 'coarse' lattice is used in which five different types of move are permitted, excluded volume effects due to side-chain packing are taken into account and the interaction energy model contains a total of seven terms. A set of low-energy structures is obtained using Monte Carlo simulated annealing; these are then refined using a finer lattice model. This second model is closer to the actual structures of proteins and uses a more accurate representation of the side chains The conformations obtained from the finer lattice model may then be converted to a full atomic model for refinement using a technique such as energy minimisation with a standard force field.

Some simplified models of proteins represent each amino acid residue as one or more 'pseudo-atoms', according to the size and chemical nature of the amino acid. These models are analogous to the 'bead' polymer models and full conformational freedom is possible. All of the standard types of calculation can be performed with these simplified models, encompassing techniques such as energy minimisation and molecular dynamics. An empirical model is used to calculate the residue–residue interaction energies. The parameters for these empirical models can be derived in a variety of ways. One option is to parametrise the simple model to reproduce the results of a more detailed, all-atom model An early attempt to develop such a representation was made by Levitt [Levitt 1976] who used energy minimisation to predict the structures of small proteins. In this model the interaction between each pair of residues is equal to the average of the calculated interaction over all spatial orientations of the two residues. Minimisation of a polypeptide chain from an initial open structure resulted in compact conformations with the same size and shape as experimentally determined protein structures, together with features such as secondary structure and β-turns. Some of Levitt's observations are still very pertinent. In particular, he notes that the 'wrong' structure may still have a lower energy (as predicted by the energy function) than the 'correct' structure; this is also found to be the case with more complex molecular mechanics functions [Novotny *et al.* 1988].

To summarise, first-principles methods have been successfully used to predict the naturally occurring conformations of small peptides but are not yet sufficiently reliable to predict accurately the structures of proteins, though in some cases the general fold of the molecule is quite similar to the native structure. However, as we shall see later in Section 10.8 lattice models have been very useful in helping to understand protein folding.

10.3.2 Rule-based Approaches Using Secondary Structure Prediction

Most protein structures contain a significant amount of secondary structure (α-helices and β-strands). An obvious way to tackle the problem of predicting a protein's three-dimensional structure is first to determine which stretches of amino acids should adopt each type of secondary structure, and then pack these secondary structural elements together.

The first step in this procedure requires the secondary structural elements to be predicted. In other words, each amino acid must be assigned to one of three classes: α-helix, β-strand or coil (i.e. neither helix nor strand) Some approaches also predict whether an amino acid is present in a turn structure. One of the first methods for secondary structure prediction was devised by Chou and Fasman [Chou and Fasman 1978]. Theirs is a statistical method, based upon the observed propensity of each of the 20 amino acids to exist as α-helix, β-strand and coil. These propensities were originally determined by analysing 15 protein X-ray structures. The fractional occurrence of each amino acid in each of these three states was calculated, as was the fractional occurrence of the amino acid over all 15 structures. The propensity of that residue for a given type of secondary structure then equals the ratio of these two values. Each residue was also classified according to its propensity to act as an 'initiator' or as a 'breaker' of α-helices and β-strands. To predict the secondary structure, the amino acid sequence is searched for potential α-helix or β-strand initiating

residues. The helix or strand is then extended so long as the average propensity value for a window of five or six residues exceeds a threshold value. β-turns are also predicted using a statistical measure of the propensity of an amino acid to exist in such structures.

Many other methods have been proposed for predicting the secondary structure of a protein from its sequence, including approaches based upon information theory [Garnier *et al.* 1978] and neural networks [Ning and Sejnowski 1988]. However, the performance of even the best methods was often barely more than 65–70% (a success rate of 33% would be achieved purely by chance, if helix, sheet and coil structures were present in equal amounts). Moreover, some of these prediction rates are probably higher than they should be due to the use of the same protein structures to develop and evaluate the models. More recent methods for secondary structure prediction do not use just the sequence of interest but also other related sequences, in the form of a multiple sequence alignment. These related sequences can be found using a standard sequence-searching program such as BLAST (described below). The use of multiple sequences improves the performance of secondary structure prediction, because the algorithm is then able to search for a consensus over the aligned sequences rather than being misled by some chance effect if only a single sequence is analysed. Two methods that use multiple sequences are PHD [Rost and Sander 1993] (a neural network approach) and DSC [King *et al.* 1997]. Methods such as these are truly able to achieve 70% prediction accuracy using unrelated proteins for development and testing. Moreover, by combining the results from more than one method it is possible to make small improvements over any single individual method [Cuff and Barton 1999]. Nevertheless, it may be that there is an inherent upper limit to the performance of secondary structure prediction, because it only considers local interactions, neglecting interactions between amino acids that are far apart in the sequence but close in three-dimensional space.

Having predicted the secondary structural elements, it is then necessary to determine how they could pack together in order to achieve a low-energy structure [Cohen and Presnell 1996]. Cohen, Sternberg and Taylor analysed the packing of α-helices and β-sheets in a number of proteins and deduced a series of rules that could be used to derive favourable packing arrangements [Cohen *et al.* 1982]. For example, from an analysis of 18 protein structures they observed that an α-helix usually packs against a β-sheet in a parallel arrangement involving two rows of non-polar residues on the helix. These rules were then used to pack the α-helices and β-sheets into a stable core structure [Sternberg *et al.* 1982]. The number of possible packing arrangements was usually very large, but this number could be drastically reduced using two simple filters. First, there had to be a sufficient number of residues between sequential elements of secondary structure to span the distance between them, and second there should be no unfavourable interactions between the helices and sheets in the packed structure. Having generated one or more approximate structures the model was submitted to an energy refinement. The results from a secondary structure prediction can also be used as a restraint in lattice models [Ortiz *et al.* 1998].

The rule-based approach to protein structure prediction is obviously very reliant on the quality of the initial secondary structure prediction, which may not be particularly accurate The method tends to work best if it is known to which structural class the protein belongs; this can sometimes be deduced from experimental techniques such as circular dichroism.

For example, some proteins contain only α-helices, which obviously makes the problem of predicting the secondary structure considerably easier. Overall, such approaches have had variable success at predicting protein structure. Nevertheless, secondary structure prediction is increasingly used as part of more general approaches to predicting protein structure.

10.4 Introduction to Comparative Modelling

There are striking similarities between the three-dimensional structures of some proteins. For example, the three-dimensional structures of trypsin, chymotrypsin and thrombin are shown in Figure 10.9 (colour plate section), from which it is obvious that they adopt very similar conformations. These proteins are all members of the trypsin-like serine protease family of enzymes but it is also possible for biologically unrelated proteins to show significant structural similarity. For example, many proteins have a structure consisting of eight twisted parallel β-strands arranged in a barrel-like structure with the β-strands connected by α-helices (Figure 10.10). This structure is often referred to as a 'TIM barrel' after triosephosphate isomerase, which was the first protein with this framework to have its structure determined by X-ray crystallography.

Comparative modelling* exploits the structural similarities between proteins by constructing a three-dimensional structure based upon the known structure(s) of one or more related proteins To do this, it is necessary to decide which protein structure(s) to use as the 3D templates, and then to decide how to match the amino acids in the unknown structure with the amino acids in the known structure(s).

If the biological function of the protein is known it may be relatively straightforward to decide which protein(s) one might wish to consider to build the model. In other cases, the function of the protein may not be known, but it may be possible to deduce to which family it belongs by searching a sequence database for the presence of particular combinations of amino acids (called *motifs*) that often imply a particular function or structural feature. Sometimes the template is the protein whose sequence is the closest match for the unknown protein. Identifying and quantifying such matches is the role of sequence alignment methods.

10.5 Sequence Alignment

We have already seen that trypsin, chymotrypsin and thrombin have very similar 3D structures. If we now overlay the three-dimensional structures of these enzymes we find that identical amino acids are found at many positions in space, including the active site serine, histidine and aspartic acid residues, as shown in Figure 10.11 (colour plate section). The amino acid sequences of these proteins can be arranged into a *sequence alignment* as

* The term 'comparative' modelling is now preferred to the older name 'homology' modelling; the latter implies some similarity of function between the unknown protein and the template, but this may not necessarily be the case.

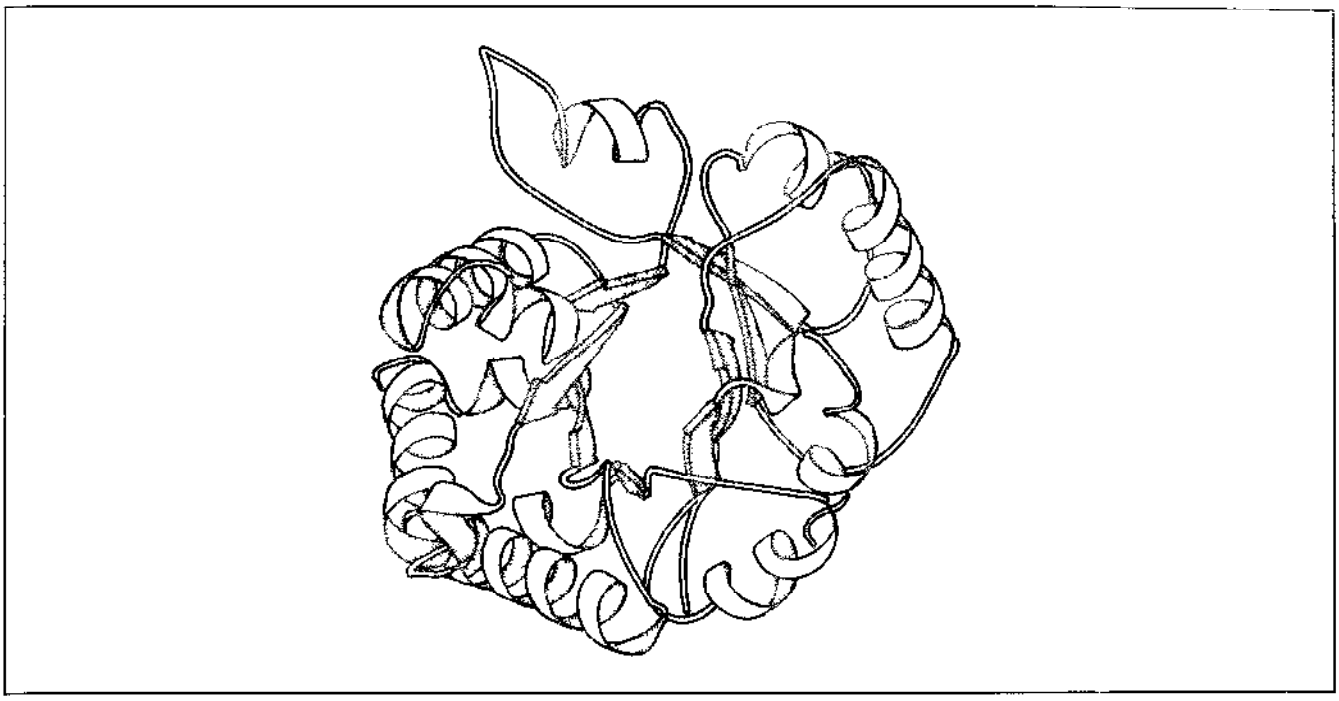


Fig 10 10: The 'TIM barrel' [Noble et al. *1991].*

shown in Figure 10.12, written using the one-letter codes for the amino acids. The relationship between sequence and structure was examined by Chothia and Lesk in 1986 [Chothia and Lesk 1986] who showed that proteins with similar sequences tend to have similar three-dimensional structures. The objective of a sequence-alignment algorithm is to position the amino acid sequences so that the matched stretches of amino acids correspond to common structural or functional features (such as the secondary structure or catalytic residues). Gaps in the aligned sequences correspond to regions where polypeptide loops are deleted or inserted. As such, sequence alignment is a key component of many procedures for predicting the structure of a new protein whose sequence has just been determined. In

```
Trypsin        SQWVVSAAHC  ... ..... YKSGIQVRLG EDNINVVEGN E.QFISASKS
Chymotrypsin   EDWVVTAAHC .... ..... GVTTSDVVVA GEFDQGLETE DTQVLKIGKV
Thrombin       DRWVLTAAHC LLYPPWDKNF TVDDLLVRIG KHSRTRYERK VEKISMLDKI

Trypsin        IVHPSYN.SN TLNNDIMLIK LKSAASLNSR VASISLP... TSCA .SAGT
Chymotrypsin   FKNPKFS.IL TVRNDITLLK LATPAQFSET VSAVCLP... SADEDFPAGM
Thrombin       YIHPRYNWKE NLDRDIALLK LKRPIELSDY IHPVCLPDKQ TAAKLLHAGF

Trypsin        QCLISGWGN. ....TKSSGT SYPDVLKCLK APILSDSSCK SAYPGQITSN
Chymotrypsin   LCATTGWGK. ....TKYNAL KTPDKLQQAT LPIVSNTDCR KYWGSRVTDV
Thrombin       KGRVTGWGNR RETWTTSVAE VQPSVLQVVN LPLVERPVCK ASTRIRITDN

Trypsin        MFCAGYLEGG ...KDSCQGD SGGPVV..CS GK.. .LQGI VSWGSGCAQK
Chymotrypsin   MICAG..ASG ...VSSCMGD SGGPLV..CQ KNGAWTLAGI VSWGSSTCST
Thrombin       MFCAGYKPGE GKRGDACEGD SGGPFVMKSP YNNRWYQMGI VSWGEGCDRD
```

Fig 10 12: Sequence alignment of trypsin, chymotrypsin and thrombin (bovine) The active sites histidine, aspartic acid and serine are highlighted

the following discussion, we will focus on the alignment of amino acid sequences but the algorithms can also be used (sometimes with small modifications) for DNA sequences.

Three general types of sequence-alignment method can be identified. Some algorithms attempt to match two sequences along their entire length. A modification of this approach is to search for local alignments involving sections (not necessarily continuous) from the sequences. The best-known examples of these first two methods were developed by Needleman and Wunsch [Needleman and Wunsch 1970] and Smith and Waterman [Smith and Waterman 1981], respectively. Both of these methods are fairly computationally intensive and not particularly suited to searching the large (and rapidly growing) sequence databases. For this purpose more approximate, heuristic methods are preferred, two major ones being BLAST and FASTA.

We will consider these algorithms in this chronological order, although in a typical comparative modelling exercise one would probably first use a heuristic algorithm to determine possible sequences of interest, then the Smith–Waterman method to identify appropriate sub-sequences, and finally the Needleman–Wunsch algorithm to derive the alignment to use in the actual construction of the model. It is always a good idea when possible to manually check any automatic alignment; the results of most automatic alignment programs can often be improved by some manual intervention. We will illustrate the various methods using the alignment of protein sequences but all of these algorithms can also be used to align nucleic acid sequences.

Any alignment algorithm requires a means for 'scoring' an arbitrary alignment of the two sequences. The objective is to find the alignment that gives the 'best' score. The simplest type of score is the *percentage sequence identity*, which gives the percentage of amino acids that are the same in the two sequences, thus identical pairs score 1 and all others score 0. An alternative approach recognises that topologically equivalent residues in two structurally homologous proteins may not be identical but often have very similar shape, electronic, hydrogen-bonding and hydrophobic properties. Such 'conservative' substitutions can frequently be made with little disruption to the three-dimensional structure of the protein and so it is desirable to take this into account in the scoring scheme. For example, in the alignment of the serine proteases position 54 is restricted to either threonine or serine due to the need to form a hydrogen bond to position 43. Dayhoff and co-workers analysed substitution frequencies in aligned sequences and have published a series of tables which give the probability of mutating one amino acid to another [Dayhoff 1978]. These probabilities are usually stored as 20×20 matrices known as PAM matrices (PAM stands for point-accepted mutation per 100 residues). One PAM corresponds to a change (on average) in 1% of all amino acid positions. The PAM concept can also be considered as a measure of 'evolutionary distance'. The best-known PAM matrix, and the one originally published by Dayhoff, is the PAM250, corresponding to 250 cycles of PAM evolution. The mutation probability matrices for evolutionary distances of 1 PAM and 250 PAM are given in Appendices 10.3 and 10.4, respectively. Each element M_{ij} of these matrices gives the probability that an amino acid in column i will have mutated to the amino acid in row j after the relevant amount of evolutionary time. It is important to realise that not every residue will necessarily have changed over this period; some will not have changed at all whereas others will have

mutated several times, possibly returning to the original state. Thus after 1 PAM there is a 0 23% probability that histidine will have mutated to glutamine whereas after 250 PAM the probability is 8%. The PAM250 matrix suggests that about 20% of the amino acids are the same after this period of evolution, with 55% of the tryptophan residues being unchanged but only 7% of the methionine residues. The matrix for any evolutionary period can be determined simply by multiplying the basic single PAM matrix an appropriate number of times. It is common to encounter a PAM matrix in a symmetrical 'log-odds' form, as shown in Figure 10.13. Each element S_{ij} of the log-odds matrix is obtained from the basic matrix by dividing M_{ij} by the relative frequency of occurrence of the amino acid i and then taking logarithms. Each element in the log-odds matrix thus represents the probability of amino acid replacement per occurrence of amino acid i per occurrence of amino acid j. An amino acid pair with S_{ij} greater than zero replaces each other more often (i.e. are likely mutations) than would be the case for random sequences of the same composition, whereas

		Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
Ala	A	2																			
Arg	R	−2	6																		
Asn	N	0	0	2																	
Asp	D	0	−1	2	4																
Cys	C	−2	−4	−4	−5	4															
Gln	Q	0	1	1	2	−5	4														
Glu	E	0	−1	1	3	−5	2	4													
Gly	G	1	−3	0	1	−3	−1	0	5												
His	H	−1	2	2	1	−3	3	1	−2	6											
Ile	I	−1	−2	−2	−2	2	−2	−2	−3	−2	5										
Leu	L	−2	−3	−3	−4	−6	−2	−3	−4	−2	2	6									
Lys	K	−1	3	1	0	−5	1	0	−2	0	−2	3	5								
Met	M	−1	0	−2	−3	−5	−1	−2	−3	−2	2	4	0	6							
Phe	F	−4	−4	−4	−6	−4	−5	−5	−5	−2	1	2	−5	0	9						
Pro	P	1	0	−1	−1	−3	0	−1	−1	0	−2	−3	−1	−2	−5	6					
Ser	S	1	0	1	0	0	−1	0	1	−1	−1	−3	0	−2	−3	1	3				
Thr	T	1	−1	0	0	−2	−1	0	0	−1	0	−2	0	−1	−2	0	1	3			
Trp	W	−6	2	−4	−7	−8	−5	−7	−7	−3	−5	−2	−3	−4	0	−6	−2	−5	17		
Tyr	Y	−3	−4	−2	−4	0	−4	−4	−5	0	−1	−1	−4	−2	7	−5	−3	−3	0	10	
Val	V	0	−2	−2	−2	−2	−2	−2	−1	−2	4	2	−2	2	−1	−1	−1	0	−6	−2	4

Fig 10.13 The PAM250 scoring matrix in the log-odds form [Dayhoff 1978]. Each element is given by $S_{ij} = 10(\log_{10} M_{ij}/f_i)$, where M_{ij} is the appropriate element of the mutation probability matrix (Appendix 10.4) and f_i is the frequency of occurrence of amino acid i (i e the probability that i will occur in a sequence by chance). The numbers are rounded to the nearest integer

a pair with S_{ij} less than zero replaces each other less often (i.e. are less likely mutations). With the log-odds matrix the probabilities can be summed when comparing sequences without having to multiply them. The lower PAM matrices tend to score very similar sequences highly, whereas the higher PAM matrices can be used to find more distant relationships. Indeed, it is sometimes suggested that combinations of PAM matrices should be used to cover both possibilities. The values in the PAM matrices were obtained by considering a small number of closely related sequences and counting the observed amino acid substitutions. The BLOSUM matrices ('block substitution matrix') are obtained in a similar fashion but are often considered superior as they were derived from analyses on sequences less similar than for the PAM matrices [Henikoff and Henikoff 1992]. More recently, 'definitive' mutation matrices were obtained using a computationally elegant procedure that enabled an exhaustive match of an entire protein sequence database to be performed [Gonnet *et al.* 1992].

10.5.1 Dynamic Programming and the Needleman–Wunsch Algorithm

The Needleman–Wunsch algorithm is widely used for aligning pairs of sequences; this algorithm guarantees to find the optimal alignment based upon the scoring matrix used [Needleman and Wunsch 1970]. The algorithm uses *dynamic programming*, which forms the basis for a number of widely used methods in bioinformatics. Sequence alignment is a 'hard' problem, because there are an extremely large number of possible solutions (of the order of 10^{30} for two sequences of length 100). Here we describe the basic algorithm as it is commonly implemented today; this is equivalent to but not exactly the same as the original Needleman–Wunsch approach

A matrix H is constructed with M rows and N columns to represent the M amino acids of protein A and the N amino acids of protein B. The elements of this matrix are filled in a sequential manner. Each element $H_{i,j}$ of this matrix corresponds to the optimal score for aligning two sub-sequences, $1 \ldots i$ from the first sequence and $1 \ldots j$ from the second ($1 \leq i \leq M$, $1 \leq j \leq N$). The algorithm works from the 'top left' to the 'bottom right' of the matrix (the original Needleman–Wunsch algorithm works in the reverse direction but gives the same result). The value assigned to each matrix element $H_{i,j}$ is determined from the three preceding elements $H_{i-1,j-1}$, $H_{i-1,j}$ and $H_{i,j-1}$ to the north-west, north and west, respectively, according to the following formula:

$$H_{i,j} = \max \begin{Bmatrix} H_{i-1,j-1} + w_{A_i,B_j} \\ H_{i-1,j} + w_{A_i,\Delta} \\ H_{i,j-1} + w_{\Delta,B_j} \end{Bmatrix} \tag{10.1}$$

These three moves to the point i, j are illustrated in Figure 10.14 and correspond to a match, a gap in sequence B and a gap in sequence A, respectively. The symbol Δ in Equation (10.1) is used to represent a gap. $w_{A_i B_j}$ represents the score associated with aligning residue A_i with residue B_j. In the simplest identity scoring scheme w_{A_i,B_j} would equal 1 if the residues were identical and zero otherwise. More typical would be the use of the PAM or BLOSUM scoring matrices. The remaining two scores, $w_{A_i,\Delta}$ and w_{Δ,B_j}, are *gap penalties*. The simplest scheme is

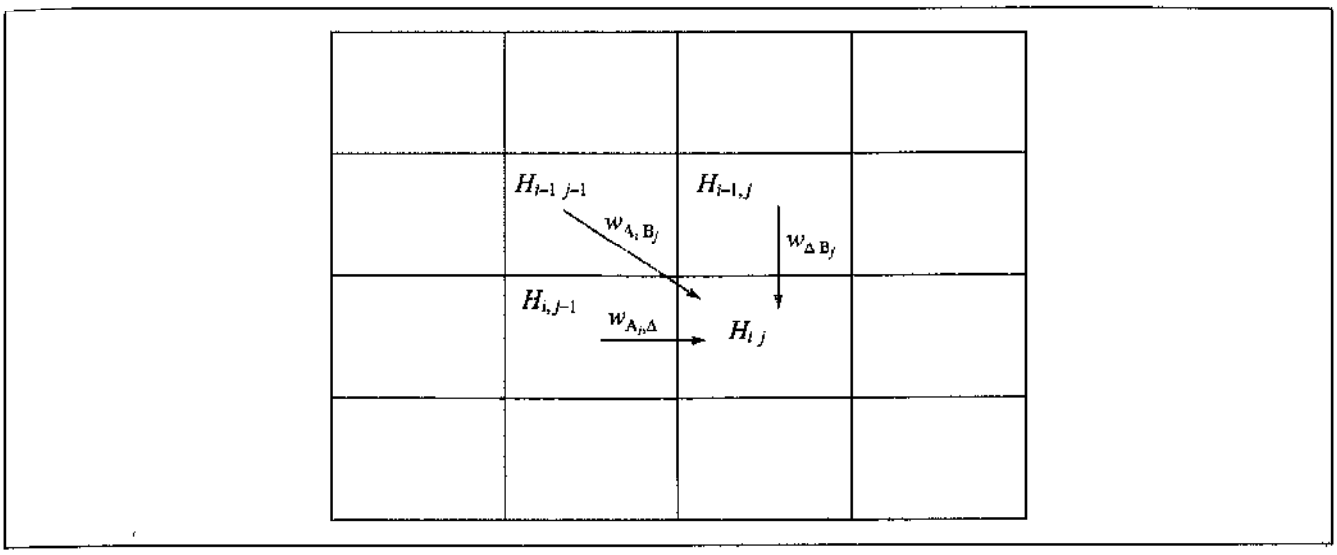


Fig 10.14: The three moves used in dynamic programming to update the matrix element H_{ij}.

to use no gap penalty. This is illustrated for two stretches of polypeptide with sequences AECENRCKCRDP (A) and AVCNERCKLCKPM (B). Sequence A thus has 12 residues and sequence B has 13. The matrix H is shown in Figure 10.15; as can be seen it is usual to introduce a 'zeroth' row and column. For each of these outer elements of the matrix there is only one predecessor matrix element, which corresponds to matching each residue with a gap. The algorithm starts at $H_{0,0}$ and fills up the matrix one row at a time. For our simple scoring scheme, which uses sequence identity as the scoring scheme and no gap penalty, these outer elements are all zero. Let us consider the element $H_{6,6}$. This corresponds to matching an arginine from sequence A with another arginine from sequence B. The three preceding elements from the matrix are $H_{5,5}$, $H_{5,6}$ and $H_{6,5}$, which all have values of 3. The value of $w_{6,6}$ is 1 (as the two residues are identical) and so the scores corresponding to the three possible moves in Equation (10.1) are 4, 3 and 3. The value of $H_{6,6}$ is thus set to 4. Now consider $H_{10\ 10}$. $w_{10,10}$ is zero (arginine from A and cysteine from B). The values of the relevant elements $H_{10,9}$, $H_{9,9}$ and $H_{9,10}$ are 6, 6 and 7, respectively, and so in this case the maximum score (7) derives from a vertical move. It is sometimes possible for more than one move to give the same score. An example of this is $H_{5,5}$ (asparagine in A and glutamic acid in B). As the two residues are not identical w_{A_i,B_j} is zero. The scores for the three types of move are 2 (diagonal, from $H_{4,4}$), 3 (from $H_{4,5}$) and 3 (from $H_{5,4}$) and so the value assigned to $H_{5,5}$ is also 3.

Having completed the scoring matrix the overall score for matching the two sequences corresponds to the value of the final matrix element, $H_{M,N}$. In our example this overall score is 8. To determine the actual alignment it is necessary to trace back through the matrix. This can be achieved by storing for each matrix element $H_{i,j}$ which of the three possibilities gave the maximum value. As we have seen, in some cases a tie results and so alternative alignments may have the same score. This is the case for our sequences, due to the presence of a tie at element $H_{5,5}$. These are shown in Figure 10.15.

When no gap penalty is used then the Needleman–Wunsch alignment may contain a large, unrealistic number of gaps. The simplest type of gap penalty (other than not to have one) is to use a length-dependent scheme in which one assigns a fixed negative value for each insertion

X	Δ	A	V	C	N	E	R	C	K	L	C	K	P	M
Δ	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	0	1	1	1	1	1	1	1	1	1	1	1	1	1
E	0	1	1	1	1	2	2	2	2	2	2	2	2	2
C	0	1	1	2	2	2	2	3	3	3	3	3	3	3
E	0	1	1	2	2	3	3	3	3	3	3	3	3	3
N	0	1	1	2	3	3	3	3	3	3	3	3	3	3
R	0	1	1	2	3	3	4	4	4	4	4	4	4	4
C	0	1	1	2	3	3	4	5	5	5	5	5	5	5
K	0	1	1	2	3	3	4	5	6	6	6	6	6	6
C	0	1	1	2	3	3	4	5	6	6	7	7	7	7
R	0	1	1	2	3	3	4	5	6	6	7	7	7	7
D	0	1	1	2	3	3	4	5	6	6	7	7	7	7
P	0	1	1	2	3	3	4	5	6	6	7	7	8	8

```
AEC ENRCK CRDP          AECEN RCK CRDP
AVCNE RCKLC KPM         AVC NERCKLC KPM
Score=8 (8 residues matched)
```

Fig. 10.15 Finding the optimal sequence alignment using dynamic programming with an identity scoring scheme and no gap penalty. Sequence A = AECENRCKCRDP, sequence B = AVCNERCKLCKPM. The value in each matrix element corresponds to the optimal score for the appropriate pair of sub-sequences. As can be seen, there are two alignments each with a score of 8.

or deletion (collectively known as *indels*). For example, if we introduce a gap penalty of −2 and make the score for a non-identical pair −1 then the dynamic programming matrix for our example changes to that shown in Figure 10.16. The alignment generated in this case happens to have no gaps (except at the end). Such an alignment is characterised by a straight diagonal.

The scoring scheme in this second example is somewhat artificial, designed primarily to illustrate the effect that a gap penalty can have on the alignment. More sophisticated types of gap penalty are possible. With the length-dependent scheme two isolated gaps contribute the same score as two consecutive gaps. Most dynamic programming methods permit a gap penalty of the form $v + uk$, where k is the gap length, v is the gap opening penalty and u is the gap extension penalty. It is most common to have a larger gap opening penalty and a smaller gap extension penalty. Yet more sophisticated are the position-specific penalty schemes. For example, if the 3D structure of one or both sequences is known then further improvements can be obtained by penalising even more severely gaps that occur in an α-helix or β-strand. The scoring matrix can also be modified to use position-specific weights. This would for example lead to an increase in the weight for aligning a residue known to be in the active site with a residue of the same type or reducing the penalty for gaps in solvent-exposed, peripheral regions.

	Δ	A	V	C	N	E	R	C	K	L	C	K	P	M
Δ	0	−2	−4	−6	−8	−10	−12	−14	−16	−18	−20	−22	−24	−26
A	−2	1	−1	−3	−5	−7	−9	−11	−13	−15	−17	−19	−21	−23
E	−4	−1	0	−2	−4	4	−6	−8	−10	−12	−14	−16	−18	−20
C	−6	−3	−2	1	−1	−3	−5	5	−7	−9	−11	−13	−15	−17
E	−8	−5	−4	−1	0	0	−2	−4	−6	−8	−10	−12	−14	−16
N	−10	−7	−6	−3	0	−1	−1	−3	−5	−7	−9	−11	−13	−15
R	−12	−9	−8	−5	−2	1	0	−2	−4	−6	−8	−10	−12	−14
C	−14	−11	−10	7	−4	−3	−2	1	−1	−3	5	−7	−9	−11
K	16	−13	−12	−9	−6	−5	−4	−1	2	0	−2	4	−6	−8
C	−18	−15	−14	11	−8	−7	−6	−3	0	1	1	−1	−3	−5
R	−20	−17	−16	−13	−10	−9	6	−5	−2	−1	0	0	−2	−4
D	−22	−19	−18	−15	−12	−11	−8	−7	−4	−3	−2	−1	1	−3
P	−24	−21	−20	−17	−14	−13	−10	−9	−6	−5	−4	−3	0	−2

```
AECENRCKCRDP
AVCNERCKLCKPM
Score=0 (6 residues matched)
```

Fig. 10.16 Finding the optimal sequence alignment using dynamic programming with a scoring scheme in which a match scores 1, a mismatch scores −1 and the gap penalty is −2.

10.5.2 The Smith–Waterman Algorithm

The Needleman–Wunsch algorithm finds a global alignment of the two sequences. This is appropriate for two sequences that are known to be similar over their whole lengths. However, it is quite common for sequences to show just local regions of similarity which would otherwise be missed in a global alignment. This can occur, for example, in multi-domain proteins, which contain a number of distinct folded sequences, each with a separate function. Even if our unknown sequence were homologous to one of the domains a global alignment against the multi-domain sequence might fail to correctly identify a match. The Smith–Waterman algorithm [Smith and Waterman 1981] is essentially the same as the method that we have described so far, except that a zero is added to the recurrence equation to give:

$$H_{i,j} = \max\left\{\begin{array}{c} H_{i-1,j-1} + w_{A_i,B_j} \\ H_{i-1,j} + w_{A_i,\Delta} \\ H_{i,j-1} + w_{\Delta,B_j} \\ 0 \end{array}\right\} \tag{10.2}$$

The zero prevents negative similarity. The pair of segments with the maximum similarity is found by first locating the matrix element with the maximum value of $H_{i,j}$ and then tracing

	Δ	A	V	C	N	E	R	C	K	L	C	K	P	M
Δ	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	0	1	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	1	0	0	0	0	0	0	0	0
C	0	0	0	1	0	0	0	1	0	0	1	0	0	0
E	0	0	0	0	0	1	0	0	0	0	0	0	0	0
N	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R	0	0	0	0	0	0	1	0	0	0	0	0	0	0
C	0	0	0	1	0	0	0	2	0	0	1	0	0	0
K	0	0	0	0	0	0	0	0	3	1	0	2	0	0
C	0	0	0	1	0	0	0	1	1	2	2	0	1	0
R	0	0	0	0	0	0	1	0	0	0	1	1	0	0
D	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Fig. 10.17 Finding the optimal local sequence alignment using the Smith–Waterman algorithm with a scoring scheme in which a match scores 1, a mismatch scores −1 and the gap penalty is −2. The algorithm identifies the conserved RCK motif.

back in the same way as before, ending with an element equal to zero. The next-best pair of segments can be found by tracing back from the second-largest element of H not associated with the first traceback. The Smith–Waterman matrix for our two sequences (scoring 1 for a match, −1 for a mismatch and −2 for a gap) is shown in Figure 10.17, from which it can be seen that the algorithm identifies the conserved RCK motif in the middle of the two sequences.

Many variants on the basic dynamic programming method are possible, some of which we have already discussed. Other modifications of a more practical nature can increase the speed and decrease the memory requirements of the procedure.

It is important to consider the significance of an alignment. A common way to quantify this is to compare the score for a given global or local alignment to the distribution of scores obtained from aligning pairs of random sequences of the same length and amino acid composition. Such a distribution can be obtained by generating a suitably large number of random sequences, calculating their alignment scores and then determining the mean and standard deviation. The 'true' alignment score is then expressed as the number of standard deviation units it is above the mean of the random distribution. These scores are referred to as SD scores or Z scores; for proteins with 100–200 amino acids a value above 15 corresponds to an almost ideal alignment, whereas scores below 5 should be treated with caution. However, the distribution of scores is often skewed and it is usually possible to identify high-scoring sequences that actually have no structural similarity to the target. This skewed distribution has important consequences, which will be discussed further in the next section.

10.5.3 Heuristic Search Methods: FASTA and BLAST

The dynamic programming methods for global or local sequence alignment guarantee to find the optimal solution and can be efficiently implemented so that they find all alignments within some cutoff score. However, such methods can take a significant amount of time to search a large sequence database. This is increasingly important due to the growth of the sequence databases. Heuristic alignment methods were developed to tackle this problem. They do not guarantee always to find the globally optimal solution but in practice they rarely miss a particularly significant match. They generally work by rapidly identifying regions of potential interest using fast look-up methods and then expand these regions locally to identify the alignment.

The FASTA algorithm [Pearson 1990; Pearson and Lipman 1988] (and its predecessor, FASTP [Lipman and Pearson 1985]) uses a look-up table in the initial step, which involves the identification of all exact matches of length k (known as a *k-tuple*, abbreviated to *ktup*). For amino acid sequences there are 20^k possible k-tuples (4^k possibilities for DNA). The location of every k-tuple within both the query and target sequences is stored in an array of length 20^k, which thus immediately enables all matches of the length k between the two sequences to be determined. The next step is to merge pairs of k-tuples that are present on the same diagonal of the pairwise sequence matrix. In some cases, there may be a continuous section of matching k-tuples along a diagonal; in other cases, there may be gaps between the k-tuples. A simple formula is used to determine which of these diagonal regions has the most significant number of k-tuple matches. These diagonal regions are scored using a scoring matrix (e.g. PAM250) and the top-scoring regions retained. It may then be possible to join together some of these regions in order to give a longer alignment via the introduction of a joining penalty (similar to a gap penalty). Finally, an optimised alignment can be recalculated centred on this highest-scoring initial region. The result of this alignment is reported as the overall score for that particular database sequence. The highest-scoring sequences from the database are then taken and dynamic programming is used to optimise the alignment, restricting the range of the dynamic programming search to a narrow band centred on the top-scoring diagonals. This four-stage process is illustrated in Figure 10.18.

FASTA only reports the single best local alignment of the query against each of the database sequences. This can mean that a region of high similarity (but which is biologically irrelevant) may mask a lower-scoring but more significant region. The ktup parameter is used to vary the speed and sensitivity of the search. For protein databases the standard value is ktup = 2, so only alignments where there is a pair of identical residues between the query and the database sequence are examined.

The BLAST program also searches for regions of local similarity but in its original form did not consider gaps (BLAST stands for Basic Local Alignment Search Tool [Altschul *et al.* 1990]) Given a query sequence and a database sequence the algorithm first finds all segment pairs of some length w (typically 3 for proteins) that have a score greater than some threshold (T) when using a suitable scoring matrix such as one of the PAM matrices. Each hit is then extended in both directions to check whether it lies within a longer alignment (called a maximal segment pair, or MSP) that has a 'significant' score. BLAST uses a parameter X

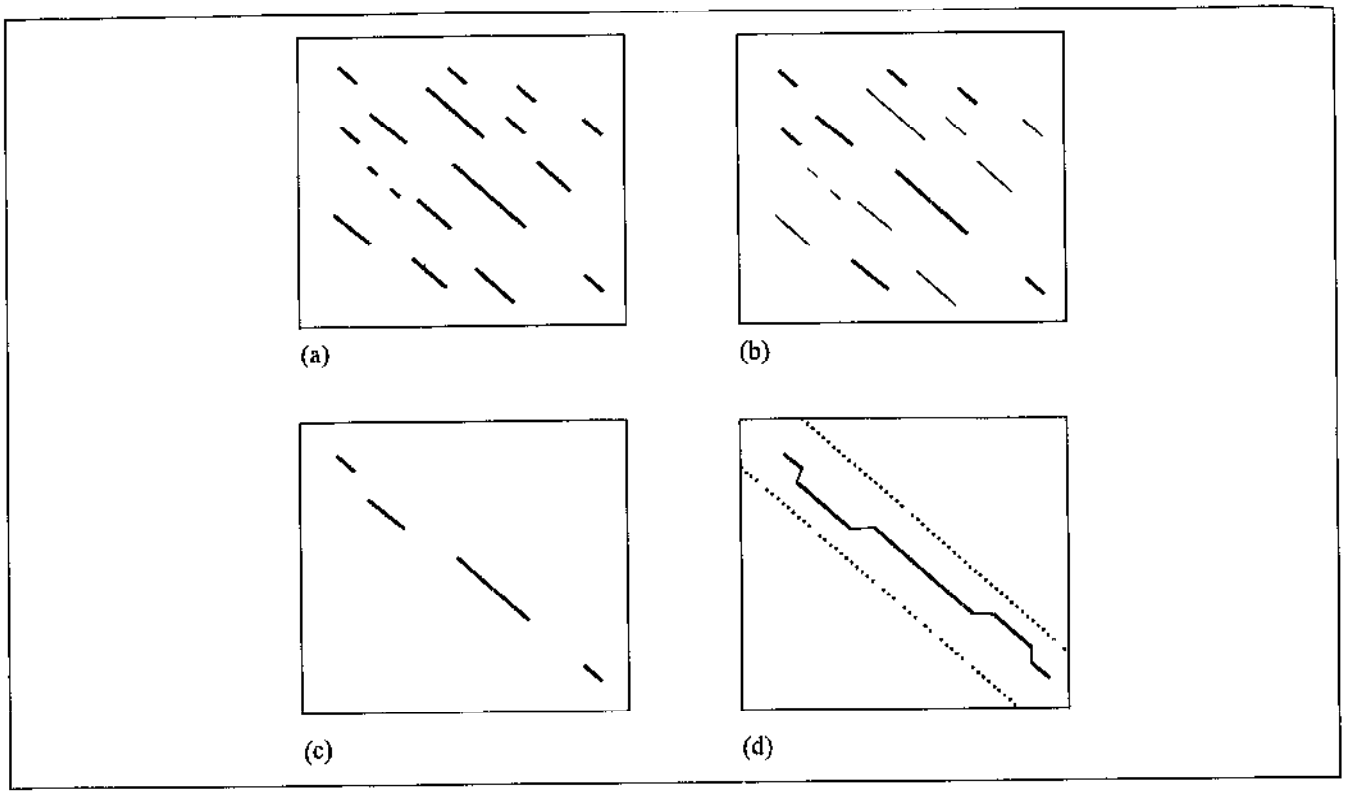


Fig 10 18: Operation of the FASTA algorithm. (a) Locate regions of identity, (b) scan these regions using a scoring matrix and save the best ones, (c) optimally join initial regions to give a single alignment, (d) recalculate an optimised alignment centred around the highest scoring initial region (Figure adapted from Pearson W R and D J Lipman 1988 Improved tools for biological sequence comparison. Proceedings of the National Academy of Sciences USA *85·2444–2448)*

to determine how long an extension will be attempted to raise the score above the required threshold score, S. The threshold score corresponds to the highest MSP score at which chance similarities are likely to appear (though in some cases the parameter X is used to reject segments that score more than X below the best found so far). The program can thus return one or more sets of local alignments that exceed the score S. The performance of the algorithm is dependent upon the values of the initial threshold, T, and the parameter X. A lower value of T reduces the possibility of missing MSPs at the expense of increasing the number of hits that proceed to the second, extension stage.

The threshold scores S used by BLAST are derived from the statistical analysis of a simple model in which the amino acids occur randomly with a probability P. The MSP scores obtained for pairs of random sequences do not follow a normal distribution (i.e. one which is symmetrical about the mean) but are skewed (formally known as an *extreme value distribution,* Figure 10.19). It turns out that the number of locally optimal segment pairs with a score of at least x is approximately distributed according to the Poisson distribution according to $KMN \exp[-\lambda x]$ where K and λ are constants that can be determined from the amino acid probabilities and the scoring matrix, and M and N are the lengths of the two sequences. This leads to the notion of a *p value,* which is the probability that a particular segment pair would occur by chance. A normalised score which enables all scoring schemes to be directly compared has also been defined. The normalised score S' is related to the basic

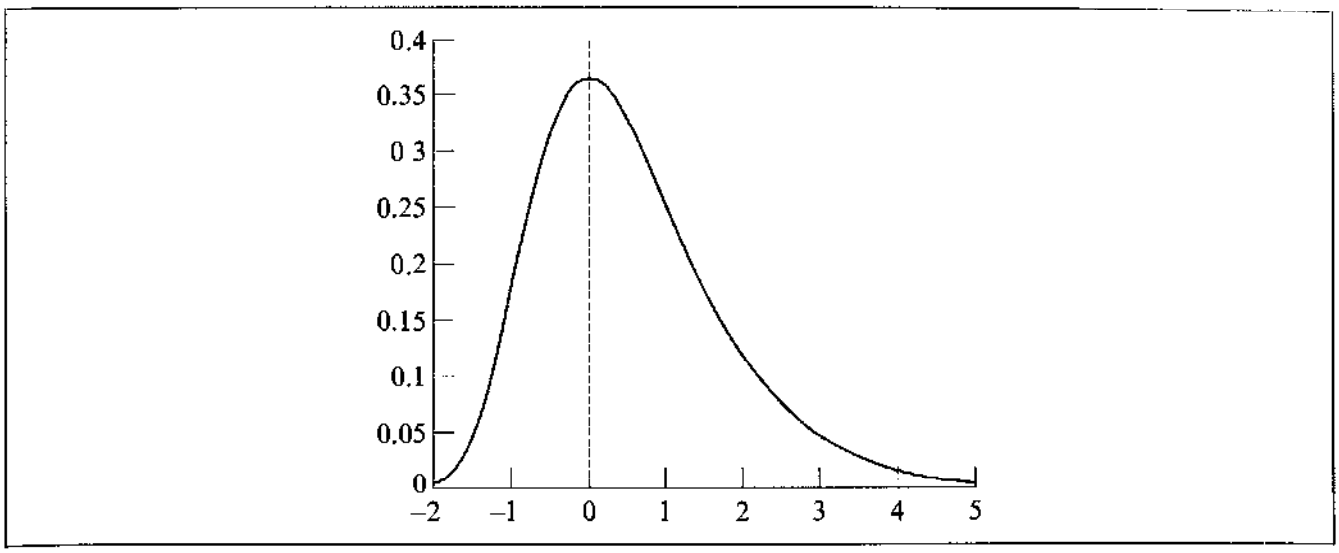


Fig 10.19: The probability density of the extreme value distribution typical of the MSP scores for random sequences The probability that a random variable with this distribution has a score of at least x is given by $1 - exp[-e^{-\lambda(x-u)}]$, where u is the characteristic value and λ is the decay constant The figure shows the probability density function (which corresponds to the function's first derivative) for u = 0 and λ = 1.

threshold score S by:

$$S' = \frac{\lambda S - \ln K}{\ln 2} \tag{10.3}$$

The number of distinct MSPs with a score of at least S' that are expected to occur by chance is represented by E, where E is:

$$E = \frac{MN}{2^{S'}} \tag{10.4}$$

Here, M is the length of the query sequence and N is the total length of the comparison sequence (which, for a database search, is obtained by adding the lengths of all the sequences in the database). The smaller the E value the more significant the match. Thus as the length of the query sequence or the size of the database increases so the normalised score S' must also increase to maintain a given significance level.

It is also possible to extend this concept to cover the presence of more than one distinct segment pair in a pair of sequences (for example, if there are three MSPs present with scores of 40, 45 and 50 then one can calculate the probability of finding three pairs with at least a score of 40 by chance). The ability of BLAST to provide a quantitative significance of any match found is a particularly useful feature of the program, which, with its continuing development and availability, has made it the most widely used method for sequence database searching

Gapped-BLAST and PSI-BLAST are two significant extensions to the basic BLAST algorithm [Altschul *et al.* 1997]. As we indicated above, the original BLAST method does not permit gaps to be introduced into the MSPs. This could lead to statistically significant matches being missed where the introduction of a gap could have enabled several local alignments to be combined. The ability to introduce gaps means that only one of the alignments need be found as it can then be extended to include the others. A dynamic programming method,

able to extend a central segment in both directions, is used for this. Another modification, introduced at the same time as Gapped-BLAST, was to require two nearby overlapping hits to be present along the same diagonal before extensions were performed. PSI-BLAST stands for Position-Specific Iterated BLAST. This method uses a score matrix that is sensitive to the position of the amino acid in the query sequence and not just its identity. In this case, an iterative procedure is used whereby the significant alignments found from the first BLAST run are used to define a position-specific score matrix for the second run, and so on. PSI-BLAST is much more sensitive than BLAST. For example, it was able to detect the similarity between histidine triad proteins and galactose-1-phosphate uridylyl-transferase proteins with E values of less than 10^{-4} whereas a BLAST search could not determine these relationships with an E value threshold as high as 0.01 [Altschul *et al.* 1997]. These particular families of proteins had previously been identified as having a possible evolutionary relationship from a comparison of the three-dimensional structures; the BLAST family of programs (as with all the sequence-alignment methods we have considered) only work with the 'one-dimensional' sequence. When compared with the rigorous Smith–Waterman method, Gapped-BLAST missed eight of the 1739 significant similarities found by the dynamic programming method but ran 100 times faster PSI-BLAST ran 40 times faster than the Smith–Waterman method and found all of the matches, together with many others. However, unsupervised use can sometimes be prone to introduce errors which may propagate over subsequent cycles.

10.5.4 Multiple Sequence Alignment

Simply put, a multiple sequence alignment is an alignment containing more than two sequences. If we know the sequences of other proteins that are suspected to be in the same family then it is usually preferable to create such an alignment. A multiple sequence alignment is often more reliable than a pairwise alignment as it is easier to detect any trends; with just two sequences it is easy to be misled by some chance correspondence. The alignment of the serine proteases in Figure 10.12 is an example of a multiple sequence alignment. Information from multiple sequences is also valuable even at the sequence-matching level, as the results with PSI-BLAST indicate (it uses the multiple hits to define the score matrix for the next iteration).

Perhaps the most obvious way to generate a multiple sequence alignment would be to extend the basic Needleman–Wunsch dynamic programming method to cover more than two sequences. Such a generalisation from two to N sequences is possible, though in practice the method is limited to comparisons of three sequences. With some approximations (such as the use of a 'window' centred on the diagonal to restrict which elements of the H matrix are considered) it is possible to increase the number of sequences that can be globally aligned. However, the most common approach to multiple sequence alignment is to use some form of hierarchical clustering approach. First, all pairwise sequence alignments are generated A hierarchical cluster analysis (see Section 9.13) is then used to group the most similar pair of sequences, then the next most similar pair, and so on Such an approach needs not only to align two sequences together but also to align one sequence with an alignment and one alignment with another alignment (an alignment contains two or more

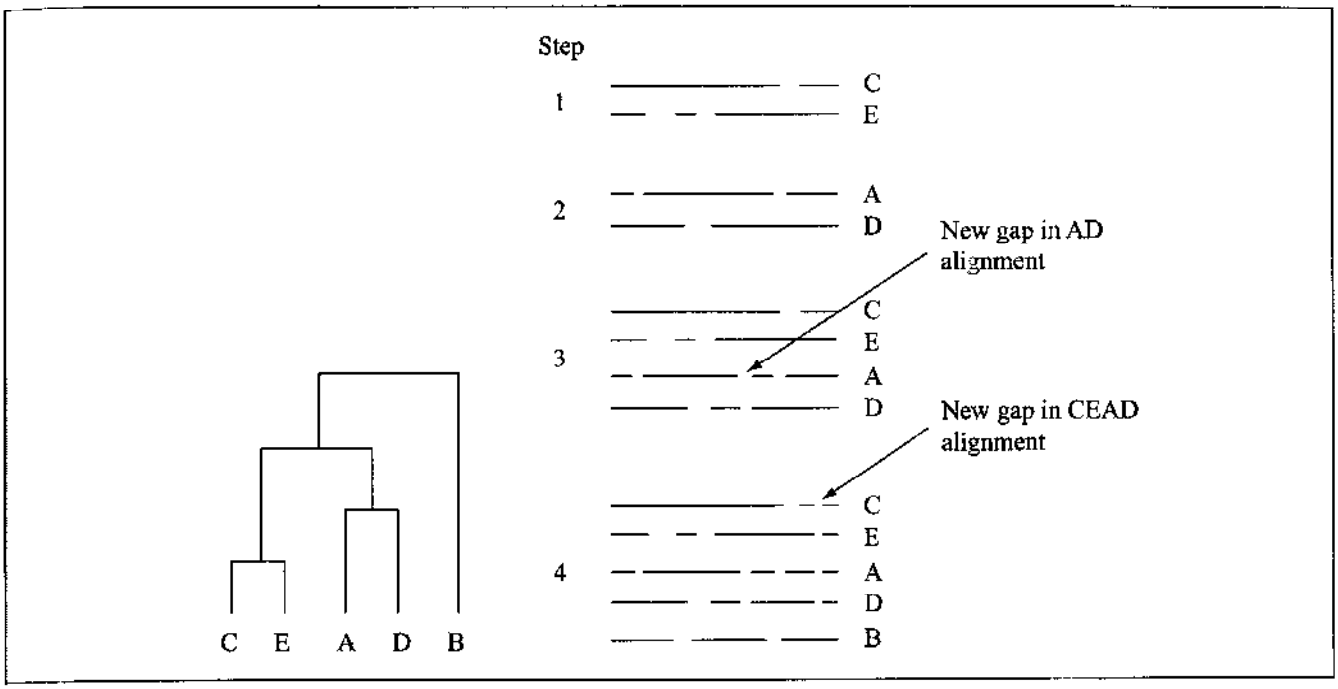


Fig 10.20 Schematic illustration of the creation of a multiple sequence alignment for five sequences A–E. In the first step sequences C and E are aligned. In the second step sequences A and D are aligned In the third step the pair CE is aligned with the pair AD. Finally, the quartet CEAD is aligned with B

individual sequences). Position-specific scoring matrices (also known as *profiles*) are used during these steps; these can be obtained by averaging the substitution values for the amino acids at a particular location. For example, suppose two sequences A and B are aligned in the first step and it is then desired to align a third sequence C with the alignment AB. If at some position A contains serine and B contains threonine then the score for matching (say) an alanine residue from C at this position would be the average score for alanine/serine and alanine/threonine. Once formed, gaps are usually maintained. Moreover, it may be necessary to introduce additional gaps into an alignment in order to achieve an optimal match in later stages. Under such circumstances, the gap is introduced into all of the sequences that form that particular alignment. The process is illustrated schematically in Figure 10.20.

More sophisticated approaches apply weighting schemes so that very similar sequences have lower weights because they contain duplicated information. In addition, special procedures can be used to handle gaps – for example, the gap penalty can be made position-specific (e.g. lower at existing gaps, higher near existing gaps, residue-specific) [Thompson *et al.* 1994]. Nevertheless, despite these developments most automatic alignments benefit from some manual intervention. Computer graphics programs which can display all of the sequences, colour-coded by amino acid type or properties, can greatly facilitate this process.

A multiple sequence alignment can suggest whether certain residues are conserved more frequently than others, and regions where insertions and deletions are more common This gives rise to the notion of a *profile* [Gribskov *et al.* 1987], which, as we indicated earlier, can be considered as a position-specific weighting scheme that indicates the score for matching an amino acid at a particular position, together with insertion and deletion penalties

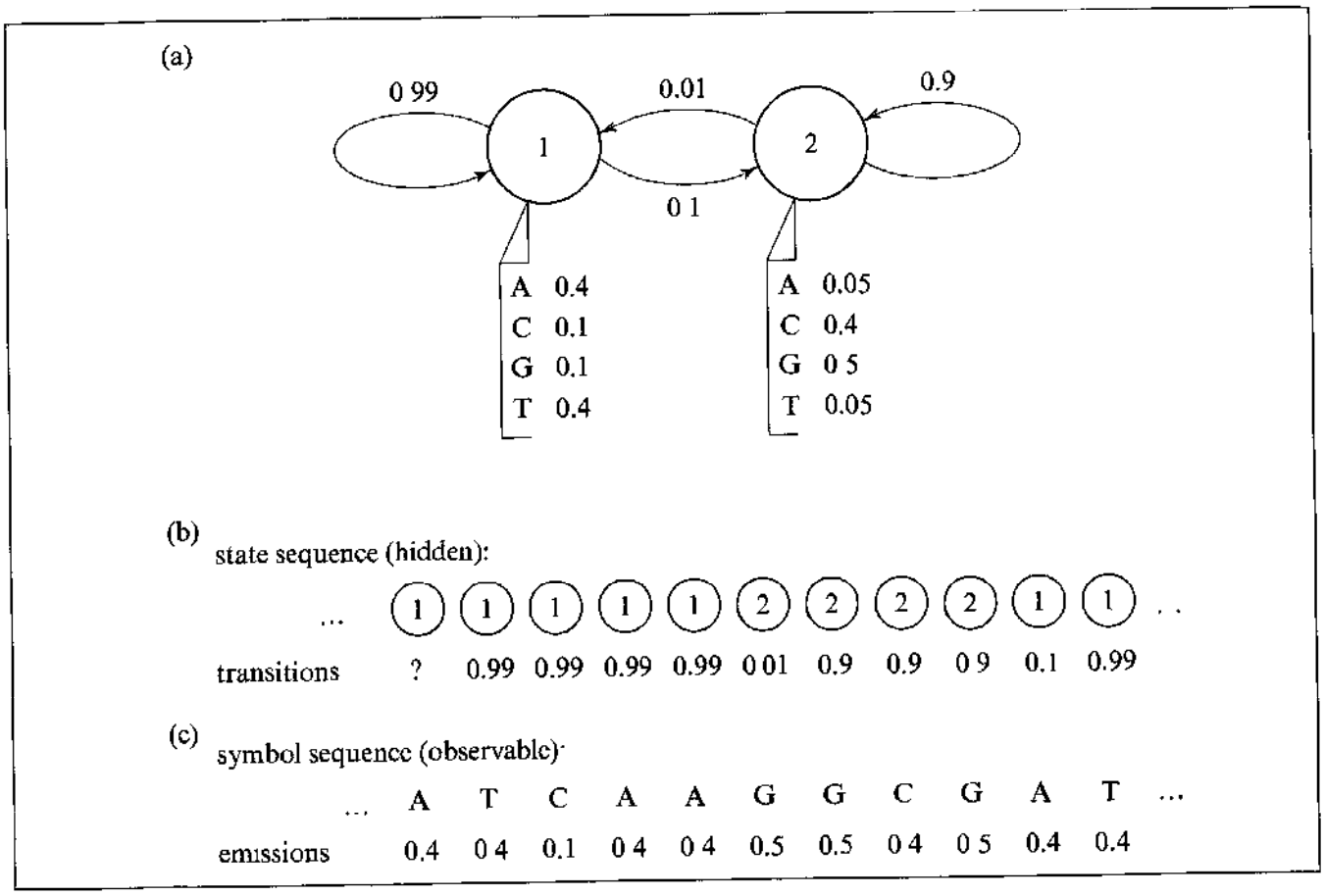


Fig 10 21. Simple two-state hidden Markov model for the generation of DNA sequences State 1 generates AT-rich sequences and state 2 generates GC-rich sequences according to the symbol emission probabilities. In addition, there are transition probabilities as indicated by the arrows A sample hidden state sequence is shown together with an (observed) symbol sequence together with the associated probabilities (Figure redrawn from Eddy S R 1996. Hidden Markov Models Current Opinion in Structural Biology *6 361–365).*

Hidden Markov models (HMMs) are a class of statistical modelling tools that can be used for multiple sequence alignment as well as other useful applications in bioinformatics. They have been extensively used in other areas of science and technology, particularly for speech recognition [Rabiner 1989]. The name derives from the fact that they are constructed from a series of 'states', each of which corresponds to the columns of a multiple alignment. These states are interconnected according to a series of transition probabilities; the choice of which state to occupy depends upon the current state and so the sequence of states is a Markov chain. They are 'hidden' because the sequence of states is not observed; rather, one observes the amino acid or nucleic acid sequence that it generates. One of the simplest HMMs with some biological relevance is shown in Figure 10 21 [Eddy 1996]; this has just two states, one of which preferentially generates AT-rich sequences and the other generates GC sequences. Having generated its symbol (according to the symbol emission probabilities) there is only a 1% probability of making a transition to the other state. This means that such a model will tend to generate sequences of either A and T or G and C with infrequent switches between the two.

For protein sequences a more complex model is used [Krogh *et al.* 1994]. First, there is a beginning and an end state with as many intervening states as there are columns in the

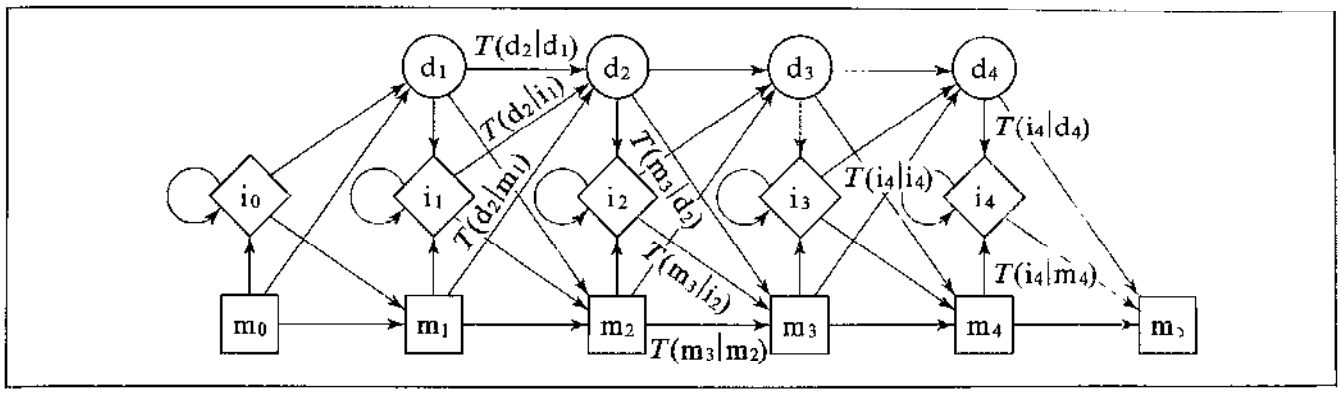


Fig 10 22 Hidden Markov model used for protein sequence analysis m_1–m_4 are match states (corresponding in this case to a four-position alignment) m_0 and m_5 are the begin and end states, and i and d are the insert and delete states There are three possible transitions from each state to other states. (Figure redrawn from Krogh A, M Brown, S Mian, K Sjölander and D Haussler 1994 Hidden Markov Models in Computational Biology Applications to Protein Modelling Journal of Molecular Biology *235 1501–1531)*

multiple alignment. At each position there are three possibilities. Either an amino acid is generated according to the distribution of that state, or the state is skipped (a deletion) or an amino acid might be inserted. There are probabilities for moving between the various states (Figure 10.22). The symbol generation and state-transition probabilities are determined by an iterative training process. A key feature of this training phase is that it is not necessary to provide aligned data, unlike other approaches. The other key feature is that an HMM inherently contains position-specific gap penalties that are learned from the data. The HMM builds its profile during the process of actually performing the multiple alignment, rather than this being a separate task that is performed once the alignment has been generated.

Having built a hidden Markov model for a particular family of proteins, it can then be used to search a database. A score is computed for each sequence in the database and those sequences that score significantly more than other sequences of a similar length are identified. This was demonstrated for two key families of proteins, globins and kinases in the original paper [Krogh *et al.* 1994]. For the kinases, 296 sequences with a Z score above 6 were identified from the SWISSPROT database of protein sequences. Of these 296 sequences, 278 were already known to be kinases or were classified as such by a battery of other procedures and were thus considered to constitute 'certain' kinases. Of the remainder, some were considered 'false positives' (i.e. were not kinases) whilst for others no definite conclusion could be drawn. In addition, a handful of sequences below the $Z = 6$ cutoff were also identified as 'false negatives'.

10.5.5 Protein Structure Alignment and Structural Databases

Despite the great progress in sequence-alignment methods there are still some cases where the similarity can only be identified by considering the three-dimensional structures of the proteins. In such cases, it is necessary to have some means of aligning two proteins based upon structural criteria. Perhaps the most basic way to achieve such a structural comparison is using computer graphics and manual superposition. However, automated methods have also been developed, of which the so-called *double dynamic programming* approach is

particularly popular [Taylor and Orengo 1989]. The method is so named because it uses two dynamic programming steps. In the first step, it is necessary to determine the score for each pair of residues, one from each structure. These scores are used to fill a rectangular H matrix, to which dynamic programming is applied to determine the optimal alignment.

In order to determine the pairwise residue score between residue i from protein A and residue j from protein B a second rectangular matrix is used. Each element (l, m) of this matrix corresponds to a pair of residues, l from protein A and m from protein B. The matrix element (l, m) is set to the similarity value:

$$s = \frac{a}{[({}^{A}\mathbf{V}_{il} - {}^{B}\mathbf{V}_{jm})^2 + b]} \tag{10.5}$$

${}^{A}\mathbf{V}_{il}$ is the 3D vector from residue i to residue l in protein A and ${}^{B}\mathbf{V}_{jm}$ is the vector from residue j to residue m in protein B (Figure 10.23). The more similar are the two vectors the greater the similarity score. a and b are constants. Having filled in the elements in the matrix corresponding to residues l and m, dynamic programming is used to find the optimal similarity S_{ij} between residues i and j. This is the value entered into the main matrix, which is used for the final dynamic programming step. Sequence information can be incorporated into the procedure by changing the numerator in Equation (10.5) from a to $(wD_{R_iR_j} + a)$, where D is one of the common scoring matrices used in normal sequence alignment for substitution of a residue R_i with residue R_j. w is a weighting factor that determines the relative contributions of structure and sequence. Other advances in the technique enable

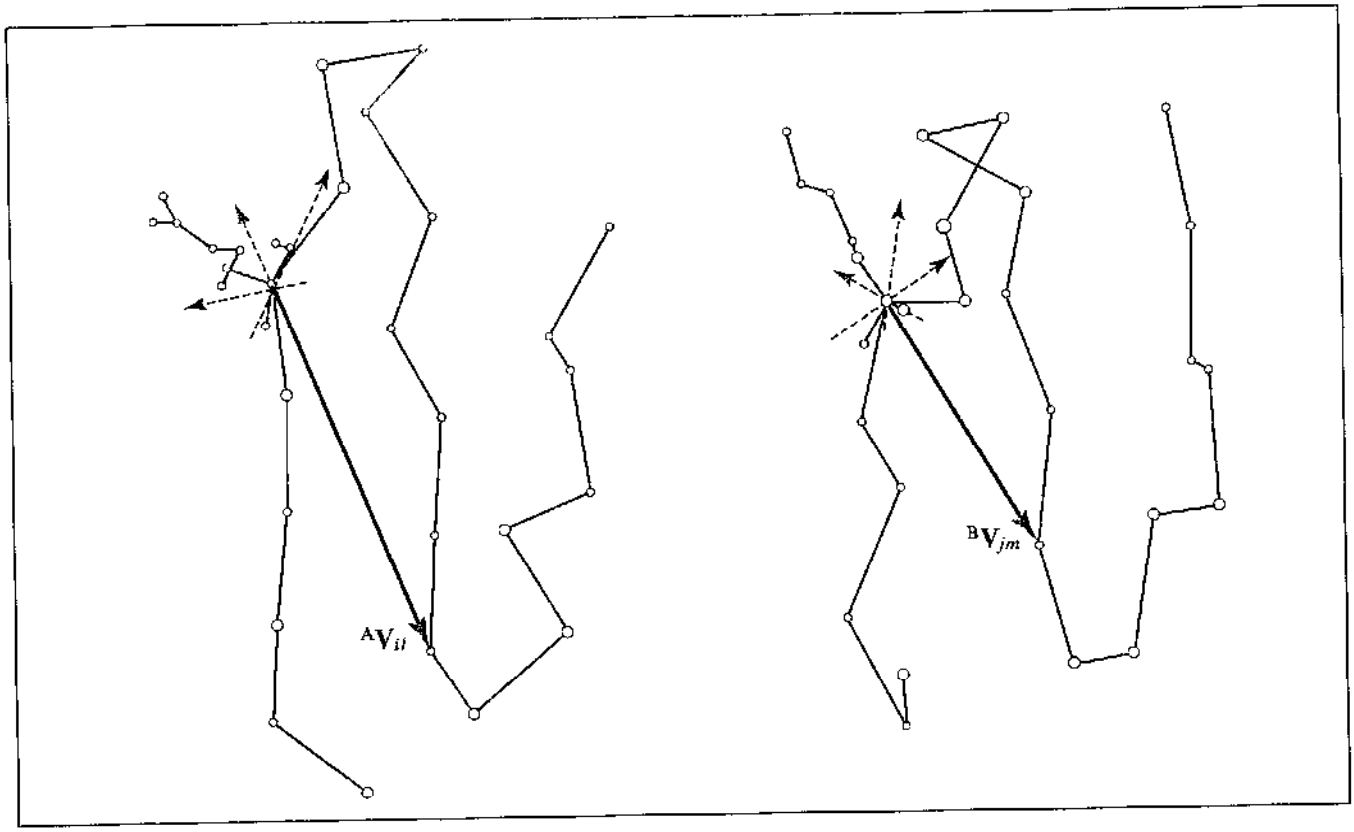


Fig 10.23. Vectors used to calculate the 3D similarity for the second matrix used in the double dynamic programming method.

local regions of structural similarity to be identified [Orengo and Taylor 1990, 1993]. A particularly fast implementation uses an initial secondary structural filter [Orengo *et al.* 1992].

A number of structural databases have been developed to classify proteins according to their three-dimensional structures. Many of these are accessible via the World Wide Web. The protein databank (PDB [Bernstein *et al.* 1977]) is the primary source of data about the structures of biological macromolecules and contains a large number of structures, but many of these are of identical proteins (complexed with different ligands or determined at different resolutions) or are of close homologues.

The SCOP (Structural Classification of Proteins) database adopts a hierarchical approach to protein structure, with several different levels [Murzin *et al.* 1995]. Unusually, the SCOP database is constructed from a visual inspection and comparison of structures. Multi-domain proteins are split into their individual domains, which are then classified into different families, superfamilies, folds and fold classes. A *family* comprises proteins with 30% or greater sequence identity or where there is a very close match of function and structure. A *superfamily* comprises those proteins with low sequence identities but which have similar structures and functional features. A *fold* is defined as a particular set of secondary structural elements joined together in a specific topology. Finally, the *fold class* is usually one of five higher-level classifications: (a) all-alpha (contains only α-helices), (b) all-beta (only β-sheets), (c) alpha and beta (usually written α/β, where α-helices and β-strands are intermixed), (d) alpha plus beta (written $\alpha+\beta$, where the α-helices and β-strands are mostly segregated), (e) multi-domain. In 1997, there were more than 7600 entries in the PDB but after removing duplicates (the same protein from the same organism) this number falls to 1729. If just one structure from each homologous family is included (defined so that no two proteins have more than 25% sequence identity) then this gives 652 proteins with 463 superfamilies and 327 folds [Brenner *et al.* 1997]. Classifications such as these have led some to suggest that there is an upper limit to the number of different protein families, with about 1000 being a commonly suggested limit. Other databases which are based upon a structural classification include CATH [Orengo *et al.* 1993] and FSSP [Holm and Sander 1994]. The latter uses an algorithm called DALI [Holm and Sander 1993] to compare protein structures based upon a comparison of residue–residue distance matrices. FSSP includes a representative set of three-dimensional structures which are used for other applications, such as threading (see below). It is also possible to combine structural and sequence information as in the HSSP database, which uses alignment methods to identify sequences that are related to proteins of known three-dimensional structure and so by implication have the same secondary and tertiary structure [Holm and Sander 1999].

10.6 Constructing and Evaluating a Comparative Model

A sequence alignment establishes the correspondences between the amino acids in the unknown protein and the template protein (or proteins) from which it will be built. The three-dimensional structures of two or more related proteins are conveniently divided into *structurally conserved regions* (SCRs) and *structurally variable regions* (SVRs). The structurally conserved regions correspond to those stretches of maximum sequence identity or sequence

similarity where one expects the conformation of the unknown protein to be very similar to that of the template protein(s). The structurally conserved regions are often found in the core of the protein or the active site. The structurally variable regions usually correspond to polypeptide loops which connect secondary structure elements together. These loops can show significant differences in sequence and be of completely different lengths.

There are currently three different classes of method for constructing the three-dimensional model [Šali 1995] The first method involves piecing together rigid bodies taken from the template protein(s). The second method assembles the target protein by joining together small segments or by reconstructing a set of coordinates. The third approach generates a series of spatial restraints from the templates, which are used in conjunction with an optimisation procedure to derive a structure of the target. Whilst each of these methods is in principle capable of producing a structure complete with loops and side chains, it is more common to consider the actual construction as a three-stage process. First, the amino acid backbone for the structurally conserved regions is generated. This gives the 'core' of the protein, to which the loops are then added. Finally, the side chains are placed. The model may then be subjected to some form of refinement, such as energy minimisation. Finally, the model should be validated to ensure that it conforms to a variety of rules about protein geometry derived from analyses of known protein structures.

The simplest type of rigid-body method involves simply transferring the backbone conformation of the core of the protein from a single template to the unknown protein. An alternative is to construct a framework by averaging the structures from a number of protein templates Each template can be given a weight related to its sequence similarity to the unknown target [Srinivasan *et al* 1996]

The segment-matching procedure starts with a basic framework, usually consisting of a set of alpha-carbon atoms. These coordinates are used to guide the fitting of the segments [Levitt 1992]. In the case of comparative modelling the initial framework would be derived from the structures of one or more homologous proteins. The conformations of the segments typically come from known protein structures, but an alternative is to use some form of geometrical algorithm to generate an energetically feasible set of atomic coordinates. The notion that the structure of a protein might be predicted by piecing together 'spare parts' from known structures was first demonstrated by Jones and Thirup [Jones and Thirup 1986]. This idea is at the heart of many methods used to construct frameworks, loops and side chains. More ambitious is the use of fragment assembly without using an underlying framework [Simons *et al.* 1997, 1999a, b]. In this case, the fragments are taken from proteins of known structure which show local sequence similarity to the unknown target. The initial structures resulting from this 'splicing' process are then subjected to simulated annealing using a scoring function that has sequence-dependent terms (representing factors such as the burial of hydrophobic residues and electrostatics) and sequence-independent terms (describing the packing of α-helices and β-strands). A number of runs are typically performed, from which the most promising are selected.

The third method, satisfaction of spatial restraints, adopts a rather different approach to the problem One possible method that would fall into this category would be distance geometry, with the distance constraints being derived from related template structures.

An alternative is to use a optimisation procedure in Cartesian space; this is the basis for a program called Modeller [Šali and Blundell 1993]. In this method, a large number of restraints are derived. Some of these restraints come from an analysis of the sequence alignment of the target protein to homologous proteins of known structure; others are derived from a statistical analysis of the relationships between various features of protein structure Typical features include the distribution of distances between alpha-carbon atoms, residue solvent accessibilities or side-chain torsion angles. Particularly relevant to comparative modelling are the associations between these features for two related proteins. Thus the backbone conformation of a particular residue may be restrained according to the residue type, the conformation of an equivalent residue in a related protein and the local sequence similarity between the two proteins. The restraints are expressed as *probability density functions* (pdf), each of which is a smooth function which gives the distribution of the feature as a function of the related variables. These individual probability density functions are combined to give a molecular function, which is then optimised. The optimisation uses a combination of conjugate gradients with molecular dynamics and simulated annealing. Local restraints are considered first, and then the global restraints.

For those methods which construct just a template for the structurally conserved regions the next task is to determine the conformations of the loop regions. These generally occur on the surface of the molecule. Each loop must obviously adopt a conformation that enables it to properly join together the appropriate parts of the core. The loop conformation should also have a low internal energy and not have any unfavourable interactions with the rest of the molecule. In certain cases, the loops may be restricted to a set of *canonical structures*. For example, it has been observed that the conformations of some antibody loops fall into a small number of classes [Chothia *et al.* 1989]. Similarly, the loops that connect certain types of secondary structure often show distinct conformations. The β-turns that connect strands of β-sheets have been classified into a small number of distinct families [Wilmot and Thornton 1988]. In other cases, we require an alternative method for predicting the loop conformations. Here we discuss just a few of the many methods that have been proposed for modelling polypeptide loops. These methods generally proceed by searching a database for suitable segments or by using some form of conformational search.

Loop conformations can be obtained by searching the protein databank for stretches of polypeptide chain that contain the appropriate number of amino acids and also have the correct spatial relationship between the two ends [Jones and Thirup 1986]. A test for amino acid homology may also be included in the criteria for loop selection. This procedure can be made very efficient by precalculating the necessary geometric information from loops in the protein databank and then using screening methods to identify the loops that can fit. This geometric screen uses information about interatomic distances between key atoms at the base of the loop. Loops that clash with the rest of the protein are rejected.

For loops that contain fewer than seven rotatable bonds, an algorithm devised by Go and Scheraga [Go and Scheraga 1970] can be used to calculate possible loop geometries directly. Go and Scheraga showed that it was possible to determine the torsion angles that would permit the end-to-end distance of the loop to achieve the desired value. The original Go and Scheraga method was developed for a model with fixed bond lengths and bond

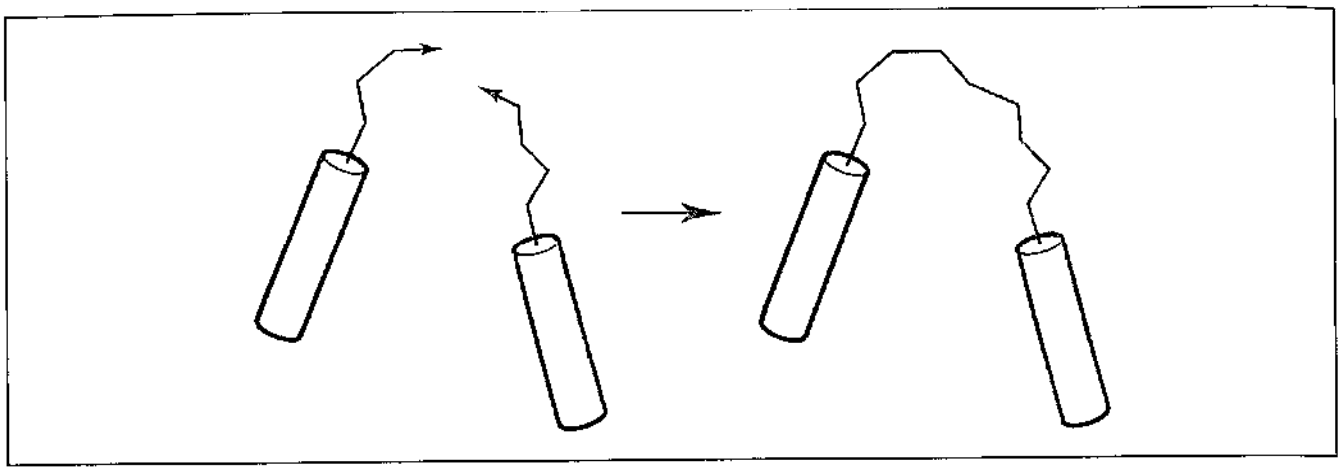


Fig. 10.24: An effective way to construct loops using a systematic search algorithm is to grow the two ends of the chain until they meet

angles; later variants permit the bond angles to deviate slightly from their equilibrium values and so have a higher chance of finding an acceptable match [Bruccoleri and Karplus 1985]. In the CONGEN program of Bruccoleri and Karplus a systematic search is used to explore the space of $N - 6$ rotatable bonds (where N is the number of ϕ and ψ torsions in the loop) [Bruccoleri and Karplus 1987]. For each conformation that is generated, the Go and Scheraga chain-closure algorithm is used to complete the structure. Purely systematic search methods can also be used to generate loop conformations. One interesting way to try to alleviate the combinatorial explosion is to construct the loop from both ends simultaneously; the half-complete loops are then joined in the middle (Figure 10.24).

Methods based on random algorithms have also been devised for modelling protein loops One interesting method is the random tweak algorithm [Shenkin *et al.* 1987], which calculates the changes in the backbone ϕ and ψ torsion angles that will enable a randomly generated loop conformation to fit a set of distance constraints. An advantage of the random tweak procedure is that almost every chain can be 'tweaked' so that it satisfies these constraints; it is also extremely fast because it scales with the number of constraints rather than with the length of the chain. However, no information is included about the interactions with the rest of the protein in the calculation and this has to be checked once a loop conformation has been generated.

Having defined one or more backbone conformations for the protein, including the loop regions, it is then necessary to assign conformations to the side chains. In the core region there may be a high degree of sequence identity between the unknown protein and the template, and the side-chain conformations can often be transferred directly from the template. Changes in amino acids in the core are often very conservative (e.g. a change from a phenylalanine to a tyrosine) and it is also easy to model the side chain in such cases Where there is less correspondence between the amino acid sequences (and especially for the loop regions) then the side chains must be added without reference to the template. A variety of systematic and random methods have been used to predict side-chain conformations; Monte Carlo, simulated annealing and genetic algorithm methods are particularly common [Vasquez 1996]. A popular tactic is to restrict the conformations of the side

chains to those that are observed in experimentally determined protein structures [Ponder and Richards 1987]; a further refinement of this approach recognises that side-chain conformations depend upon the conformation of the main chain [Dunbrack and Karplus 1993] Side-chain prediction methods invariably keep the backbone fixed.

The initial structures obtained from a comparative modelling exercise can often be rather high in energy. Energy minimisation is thus often performed to refine the structure, though one should be careful to ensure that the minimisation does not cause any drastic changes and some practitioners deprecate its use.

Once a protein model has been constructed, it is important to examine it for flaws. Much of this analysis can be performed automatically using computer programs that examine the structure and report any significant deviations from the norm. A simple test is to generate a Ramachandran map, in order to determine whether the amino acid residues occupy the energetically favourable regions. The conformations of side chains can also be examined to identify any significant deviations from the structures commonly observed in X-ray structures. More sophisticated tests can also be performed. One popular approach is Eisenberg's '3D profiles' method [Bowie *et al.* 1991; Lüthy *et al.* 1992]. This calculates three properties for each amino acid in the proposed structure: the total surface area of the residue that is buried in the protein, the fraction of the side-chain area that is covered by polar atoms and the local secondary structure. These three parameters are then used to allocate the residue to one of eighteen environment classes. The buried surface area and fraction covered by polar atoms give six classes (Figure 10.25) for each of the three types of secondary structure (α-helix, β-sheet or coil). Each amino acid is given a score that reflects the compatibility of that amino acid for that environment, based upon a statistical analysis of known protein structures. Specifically, the score for a residue i in an environment j is calculated using

$$\text{score} = \ln\left(\frac{P(i:j)}{P_i}\right) \tag{10.6}$$

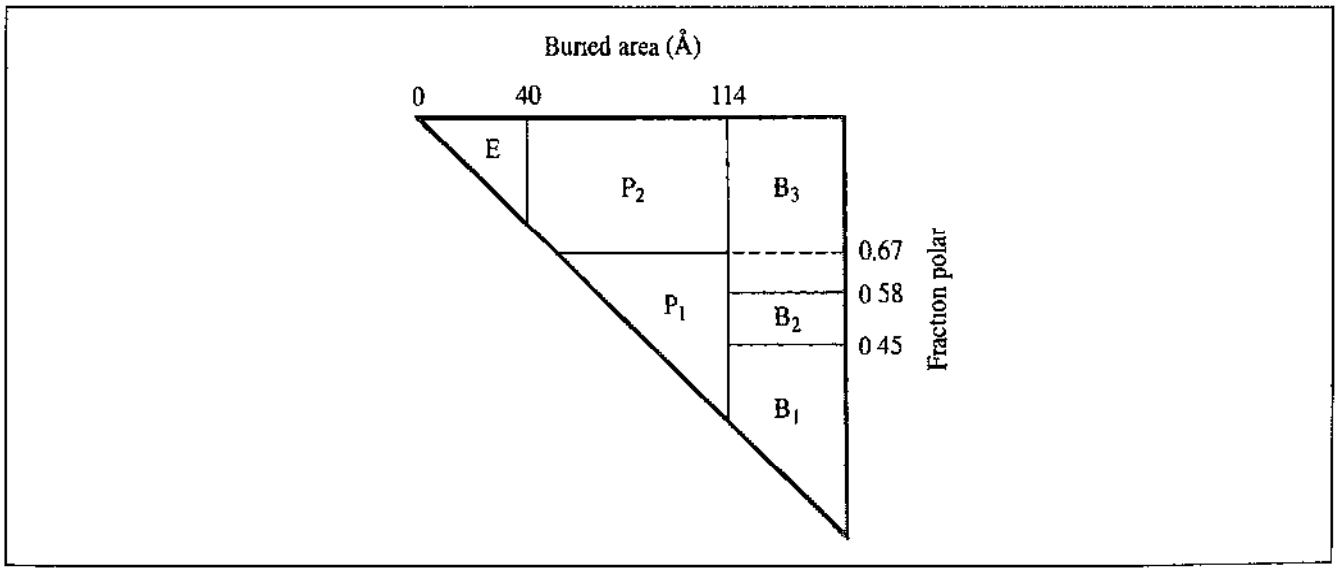


Fig 10.25 The six environment categories used by the 3D profiles method (Figure adapted from Bowie J U R Lüthy and D Eisenberg 1991 A Method to Identify Protein Sequences That Fold into a Known Three-Dimensional Structure Science 253:164–170.)

where $P(i:j)$ is the probability of finding residue i in environment j and P_i is the overall probability of finding residue i in any environment. For example, $P(i:j)$ is -0.45 for a valine residue in a partially buried environment with a high fraction (>67%) of the surface covered by polar atoms in an α-helix. The negative number indicates that this environment is not favoured for valine. However, this environment is more favoured by arginine, for which the $P(i:j)$ value is 0.50.

The 3D profiles method can be used to calculate an overall score for a protein model. It was found that deliberately misfolded protein models have low scores because they contain residues in environments with which they are not compatible. Such misfolded models often cannot be distinguished from the correct structures using molecular mechanics energies. The 3D profile can also be used to identify whether a generally correct model contains regions of incorrectly assigned residues. This is usually done by plotting the score as a function of the sequence, as shown in Figure 10.26. Any residues for which the score falls significantly below the average score should be investigated to check whether the model is faulty in that particular region.

Comparative modelling is a widely used method, with many models being published in the literature Of particular importance are those papers which retrospectively compare a predicted model with the subsequent experimental structure. An early example was the comparison of a model of the aspartyl protease renin built using the Composer program [Frazao *et al.* 1994]. Renin is an important enzyme in the control of high blood pressure and so is of potential interest as a pharmaceutical target. Composer uses the rigid-body

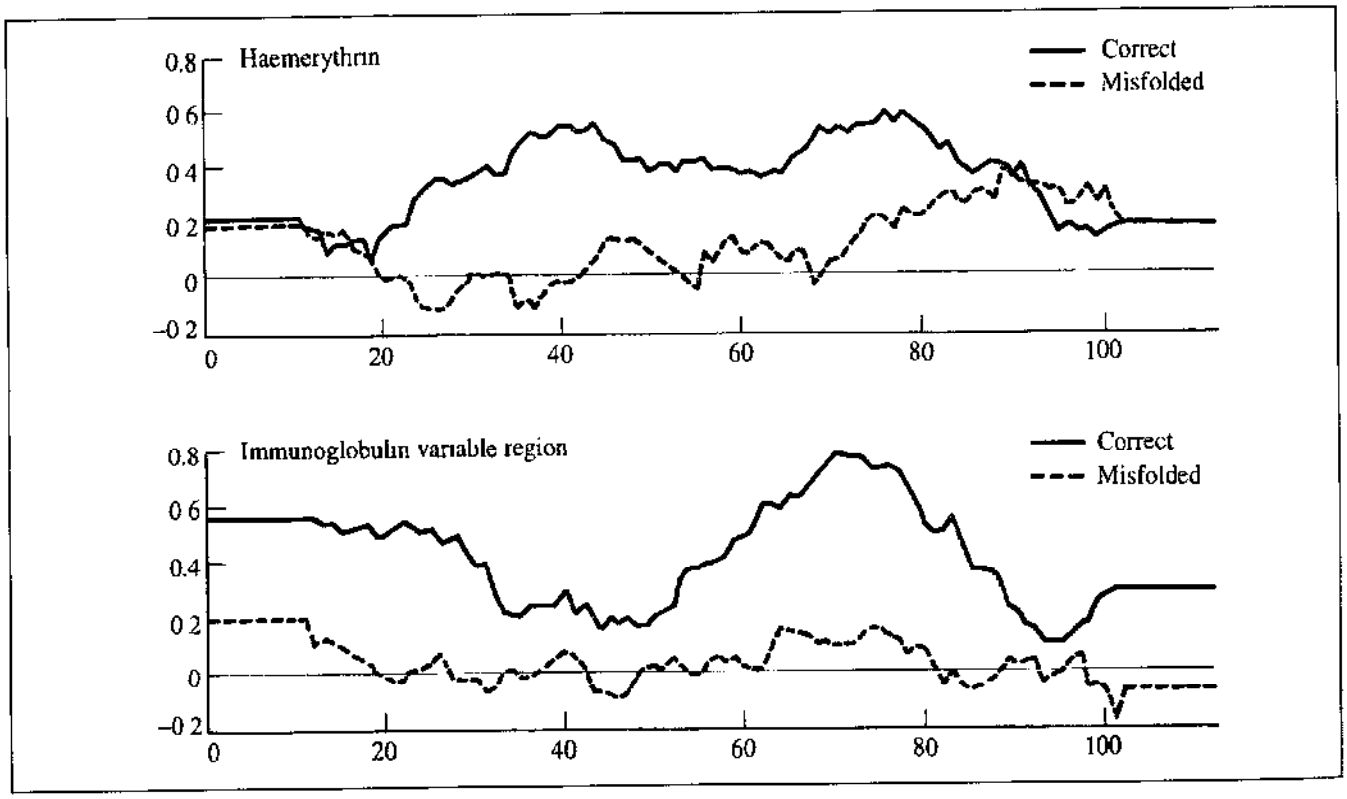


Fig 10 26 The 3D profiles output for incorrect and partially incorrect protein models compared to the correct structures The vertical axis gives the average profile score for a 21-residue window. (Figure redrawn from Lüthy R, J U Bowie and D Eisenberg 1992. Assessment of Protein Models with Three-Dimensional Profiles Nature 356·83–85)

approach; in this case, two homologous aspartyl proteases (pepsin and chymosin) were used as templates, followed by loop modelling using the 'spare parts' method. Side chains are assigned using a series of rules derived by examining topologically equivalent positions in homologous structures. This analysis leads to 1200 rules, one for each 20 by 20 amino acid replacement in each of the three types of secondary structure (α-helix, β-strand, neither). If no applicable rule exists then a conformation is chosen from a rotamer library. The model had an RMS fit for 280 alpha-carbon atoms of 0.84 Å (the atom pairs were selected by applying a cutoff of 3.5 Å). Of some interest was the fact that the model was closer to the X-ray structure than it was to either of the two structures used for its construction. However, the analysis did also highlight some areas for improvement, such as a proline-rich loop for which there were few representative examples in the database of known structures.

10.7 Predicting Protein Structures by 'Threading'

Threading (more formally known as 'fold recognition') is a method that may be used to suggest a general structure for a new protein [Jones *et al.* 1992; Jones and Thornton 1993]. The basic threading concept is very simple. Suppose we wish to predict the structure of an amino acid sequence, and that we have available a number of three-dimensional protein structures, typically chosen to represent common structural classes. We wish to know which structure is most compatible with the sequence of the unknown protein. This is done by 'threading' the sequence through each protein structure in turn (hence the name). Threading methods are closely related to *ab initio* approaches to protein structure prediction, but whereas the latter can effectively explore all of the conformational space (albeit a restricted lattice in many cases) threading methods inherently limit the search space to the conformations of known structures. As such, threading is doomed to fail for any protein which adopts a completely new fold. That threading can work is, of course, a consequence of the fact that there does appear to be a finite set of protein folds and that proteins with very weak sequence similarity can adopt very similar structures. Thus when two proteins have a sequence identity of more than 70% it should be straightforward to determine an alignment that leads to a reliable model. As the degree of similarity falls so the task becomes more difficult, and when one enters the so-called 'twilight zone' (corresponding to less than 20–30% sequence identity) then comparative modelling is often considered to be inappropriate (or at least, any model should be treated with caution). Threading should (in principle, at least) be particularly suited to such problems.

A naive threading implementation involves advancing each amino acid to occupy the location occupied in the previous iteration by its predecessor. A score is calculated for each structure so generated and the process is repeated until the sequence has been entirely threaded through the structure. The output is the structure or structures that correspond to the lowest value of the scoring function. As can be imagined there are very many possibilities to consider and so threading programs use special searching methods such as double dynamic programming to efficiently find the best ways to match the sequence to the structure. Even so, finding the optimal alignment is a very complex problem (particularly as one needs to consider gaps) and has resulted in some useful approximations that can be used to

make the problem more manageable. For example, in the *frozen approximation* each residue from the target sequence is scored according to the residues present in the actual template [Godzik *et al.* 1992]. It is also possible to use very high (or infinite) gap penalties in certain regions of the structure such as elements of secondary structure.

A variety of different scoring functions have been used for threading [Bryant and Lawrence 1993; Maiorov and Crippen 1994; Jernigan and Bahar 1996; Jones and Thornton 1996], but most share some common features. A threading calculation can require a large number of possibilities to be considered, and so the scoring functions are usually quite simple. This also reflects the low-resolution nature of the problem, in that one is usually attempting just to predict the basic fold of the protein. Each amino acid is typically treated as a single interaction site. Many of the scoring functions used in threading algorithms are potentials of mean force that provide an estimate of the free energy of interaction between two residues as a function of their separation. These potentials of mean force are calculated from statistical analyses of known protein structures For example, if one plots the distribution of distances observed in X-ray protein structures between amino acids that are separated by three other residues in the sequence (i.e. between residues i and $i + 4$) then a large peak is observed in the interval 5 9–6 5 Å and a broad shoulder at 11.4–13.3 Å. These reflect the presence of α-helix and β-strand conformations. It is from these distribution frequencies that one can determine the residue–residue potentials of mean force. Sippl has provided an example of the use of such potentials [Sippl 1990]. The pentapeptide sequence valine–asparagine–threonine–phenylalanine-valine (VNTFV in the one-letter amino acid code) adopts an α-helical conformation in the protein erythrocruorin but a β-strand conformation in ribonuclease. The potential of a mean force suggests that the β-strand is the more stable conformation for the isolated pentapeptide, but that when it is flanked by aspartic acid at one end and alanine at the other (as in erythrocruorin), the α-helix does indeed become the more stable conformation. For threading algorithms one is particularly interested in the interactions between amino acids that are close in three-dimensional space but far apart in the sequence, and the potentials used in such calculations are derived appropriately. In addition to the pairwise knowledge-based term a solvation contribution is often added. This measures the propensity of each amino acid for a certain degree of solvation, to help ensure that hydrophobic residues pack into the central core of the protein and hydrophilic residues are on the outside. As it tends to be just the cores that are conserved, the pairwise interaction term may be omitted for loop residues, which are therefore treated just by the solvation term.

Although knowledge-based potentials are most popular, it is also possible to use other types of potential function Some of these are more firmly rooted in the fundamental physics of interatomic interactions whereas others do not necessarily have any physical interpretation at all but are able to discriminate the correct fold from 'decoy' structures. These decoy structures are generated so as to satisfy the basic principles of protein structure such as a close-packed, hydrophobic core [Park and Levitt 1996]. The fold library is also clearly important in threading. For practical purposes the library should obviously not be too large, but it should be as representative of the different protein folds as possible. To derive a fold database one would typically first use a relatively fast sequence comparison method in conjunction with cluster analysis to identify families of homologues, which are assumed to have the same fold. A sequence identity threshold of about 30% is commonly

applied, with one representative being chosen from each cluster. Each of these representative structures is then compared to all other structures to enable a smaller set of representative folds to be identified. The number of unique folds is usually found to be about $\frac{1}{20}$th the total number of structures in the protein databank, as we saw for the SCOP database.

10.8 A Comparison of Protein Structure Prediction Methods: CASP

The utility of a protein model depends upon the use to which it is put. In some cases, one is only interested in the general fold that the protein adopts and so a relatively low-resolution structure is acceptable. For other applications, such as drug design, the model must be much more accurate, including the loops and side chains. In such cases, a poor model may often be far worse than no model at all, as it can be seriously misleading.

To evaluate the then available techniques for calculating protein models, a 'competition' was organised in 1994–1995. In this first CASP* challenge entrants were invited to predict the three-dimensional structures of seven proteins from their amino acid sequences [Mosimann *et al* 1995]. The structures of these seven proteins were simultaneously solved using X-ray crystallography, but these structures were not made available to the modellers. A total of 43 separate structures were submitted by 13 research groups, each model then being compared with the X-ray structure. The quality of each model was also assessed using a variety of methods, including the calculation of Ramachandran maps and 3D profiles.

Each competitor first had to decide which of the known protein structures they wished to use as the template; they then had to construct a sequence alignment (possibly using other protein sequences from the same family), and finally had to construct the model. The degree of sequence identity for the seven proteins ranged from 22% to 77%. One reassuring conclusion was that in favourable cases very accurate models could be constructed; the 'best' structure had an RMS difference with the X-ray structure of just 0.6 Å (for the protein NM23, which not surprisingly had the highest sequence identity with its template and indeed the same number of amino acid residues). Overall, the accuracy of the models largely depended upon two factors: the percentage sequence identity and the presence of substantial insertions or deletions between the template and target structures. The need for an accurate sequence alignment was evident; an incorrect alignment almost always resulted in an incorrect structure. For those proteins where there were large insertion loops, these were invariably predicted incorrectly, demonstrating a clear need for new strategies to tackle this problem. In some cases, models of the same protein were generated using both 'hands-on' approaches (where the modeller directed the construction of the structure) and wholly automatic procedures. In all cases, the manual structure was superior to the automatic model.

One disturbing finding was that a significant proportion of the models contained errors, even including amino acids with the wrong stereochemistry. In addition, significant deviations

* CASP stands for Critical Assessment of techniques for protein Structure Prediction

from planarity of the amide bond were noted in many structures, and the torsion angles of the side chains often deviated from the distributions observed in experimental protein structures. Some of the models contained high-energy steric interactions between non-bonded atoms and unlikely distributions of amino acids (i.e hydrophilic residues on the inside and hydrophobic residues on the outside). Most, if not all, of these problems can be identified very simply using publicly available software, and the organisers of the competition suggested that any models submitted for publication should be accompanied by output from a structure-verification program to allow an objective assessment of the quality of the model. The use of energy minimisation to refine models was identified as the cause of many of these problems, thereby highlighting the need for alternative protocols for the refinement stage of a comparative modelling exercise.

The first CASP competition was generally regarded as a great success and subsequent rounds have been organised, with increasing numbers of research groups submitting entries. At the time of writing the final reports from CASP3 have now been published [Moult *et al.* 1999], and the arrangements for CASP4 are well in hand. Three general categories were identified for CASP3. These comprise comparative modelling (which relies upon a clear relationship between the target protein and a protein of known structure), fold recognition (of which threading is one example), and *ab initio* prediction methods (which do not rely directly upon knowledge of any complete structures). Of the targets in CASP3, 15 were considered to be in the first category, 22 in the second and 15 in the third. A high level of participation ensured the success of the enterprise, which culminated in a meeting at Asilomar, California, to discuss the results Of especial value to the community is that participants are encouraged to assess not only what was successful but (more importantly) what lessons had been learned. Whilst it is difficult to identify any real trends from just three CASPs some of the key developments have been the continuing improvement of *ab initio* prediction methods, the introduction of advanced sequence-comparison methods such as PSI-BLAST and hidden Markov models, which often performed as well as more 'sophisticated' techniques for suggesting possible homologues, and a gradual improvement in comparative modelling methods. Perhaps *the* single most important message that has emerged from all of the CASP competitions (as well as other studies) is that the real key to comparative modelling is the quality of the alignment.

10.8.1 Automated Protein Modelling

As we mentioned in the introduction to this chapter, the Human Genome Project is generating thousands of protein sequences, at a far greater rate than the structures can be solved by experimental techniques. Given the close relationship between the three-dimensional structure and the function of a protein there is increasing interest in automating the process of predicting the structures of these proteins, as a prelude to assigning a tentative function A number of the methods that we considered for comparative modelling and fold recognition can be run in an automated manner. Indeed, one subsection of CASP3 was an assessment of automated methods, many of which have been made available over the World Wide Web (one of the earliest being Swiss-Model [Peitsch 1996]). This assessment confirmed that the most accurate models are those which benefit from some human input (especially during

the alignment stage), but the automated methods clearly have huge potential in exploiting the data being generated by the genome project.

An automated modelling procedure obviously needs to generate its model without the need for intervention: identifying structural templates related to the target sequence, aligning the templates with the target, building the model and finally evaluating and assessing the model. Of course, not all of the unknown proteins in a given genome will necessarily have a known structure that can be used as a template. For example, Sánchez and Šali considered the baker's yeast genome (*Saccharomyces cerevisiae*) [Sánchez and Šali 1998]. Of 6218 open reading frames (ORFs; a region of DNA that is converted into RNA and thence into protein) there were related structures for 2256 (36.3%), with an average pairwise sequence identity of 27%. Model building was performed with the Modeller program [Šali and Blundell 1993]. The models were then assessed for their quality, with 1071 of the original ORFs being considered to have a reliable model (17.2%).

Such large-scale modelling experiments can require significant computational resources, but the main bottleneck is generally considered to be the absence of structurally defined members of many protein families and the difficulty in detecting weak similarities, which would enable the appropriate template structures to be identified for more detailed comparative modelling. Above all, it is important to remember that no one single theoretical or experimental technique can predict protein function from sequence; rather, it is the application of an appropriate combination of methods that is required. Moreover, although our emphasis has been on the importance of the three-dimensional structure, such information is only one part of the jigsaw An illustration is provided by a study which compared all of the protein structures released in 1998 with all structures that were known by the end of 1997 [Koppensteiner *et al.* 2000]. Some 147 of the proteins (corresponding to 196 domains) solved in 1998 had no significant sequence similarity to any of the pre-1998 proteins. However, when the structures of these 196 domains were compared with the pre-1998 set it was found that 147 of the domains had significant structural similarity with a previously known protein fold. Moreover, in two-thirds of these cases the function was also the same. The implication from these and similar studies is that computational techniques can be very effective at processing and filtering the raw sequence information in order to identify proteins that may be of interest and thus to suggest what experiments should be performed in order to confirm the hypothesis.

10.9 Protein Folding and Unfolding

The mechanism by which a protein folds to its native state has long been a subject of considerable interest, from both experimental and theoretical perspectives. As we noted in the introduction to this chapter, proteins typically adopt a single structure, corresponding to the global minimum of free energy under physiological conditions Moreover, protein sequences can generally fold into this unique state in just a few seconds (or less) from any arbitrary starting conformation. Whilst exceptions to both of these two facts can be found, they do often hold true for small, water-soluble proteins such as enzymes, which have been the focus of most of the studies to date. The mechanism by which a protein is able

to fold into its unique fold was considered by Levinthal, who showed that folding could not occur via a systematic enumeration of all possible conformations [Levinthal 1969]. If it is assumed that there are three conformations for each amino acid then a polypeptide chain with (say) 100 amino acids would have about 10^{48} conformations. If the interconversion between conformations required just 10^{-11} seconds then it would take about 10^{29} years to explore them all. Of course, this is for the most basic of grid search algorithms, but even the most advanced systematic conformational search would still require an inordinate amount of time to identify the global minimum energy conformation. This discrepancy between the time for the exhaustive search and the observed timescale of protein folding is popularly known as the *Levinthal paradox*.

Two general types of computational model have been used to investigate protein folding: simple lattice models and atomistic models. These two approaches are complementary; lattice models attempt to capture the essential physics of the problem but do not provide information about specific interactions at the atomic level. It is often possible to exhaustively enumerate all possible states on the entire energy surface of a lattice model, in contrast to an atomistic model. Another important feature of atomistic simulations is that most (to date) have considered protein *unfolding*, rather than folding. The two processes are obviously linked through the principle of microscopic reversibility, though it has been argued that an unfolding pathway obtained with the 'strong' unfolding conditions (such as high temperature) that are often used in the simulations may not necessarily correspond to the 'true' physiological folding pathway [Finkelstein 1997]. It is just becoming possible to directly simulate the folding of proteins which, although containing a very small number of amino acids, do have recognisable secondary structure in solution. In the remainder of this section, we will consider both types of model and what they can tell us about the nature of protein folding and how a combination of experiment and theory has led to a 'new view' of protein folding that appears to resolve the Levinthal paradox.

This 'new view' considers an ensemble of structures through which a protein can fold to the native conformation rather than a single pathway involving a number of distinct intermediates. As such, a statistical description of the energy surface can be applied [Bryngelson and Wolynes 1987; Bryngelson *et al.* 1995; Onuchic *et al.* 1997]. The resulting theory suggests that the energy landscape of protein folding (i.e. the variation of free energy with the protein conformation, and the form of that free energy) generally resembles a funnel, but one which is 'rough', with local minima where the protein can transiently reside. Most of the molecular organisation occurs early in the procedure and can be described using a few parameters. Later in the folding process the protein may become trapped in the local minima, which can give rise to the semblance of a discrete folding pathway that is sensitive to the amino acid sequence and three-dimensional structure. A schematic representation of a folding landscape that conforms to these ideas is shown in Figure 10.27.

The essential features of protein lattice models were described in Section 10.3.1. Protein folding is usually studied with a self-avoiding chain on a cubic lattice with one residue per vertex and a simple interaction model which only includes interactions between pairs of monomers that are in contact on the lattice but are not successive in the sequence. Polymers of length 27 that occupy all sites of a $3 \times 3 \times 3$ cube are particularly popular. There are nearly

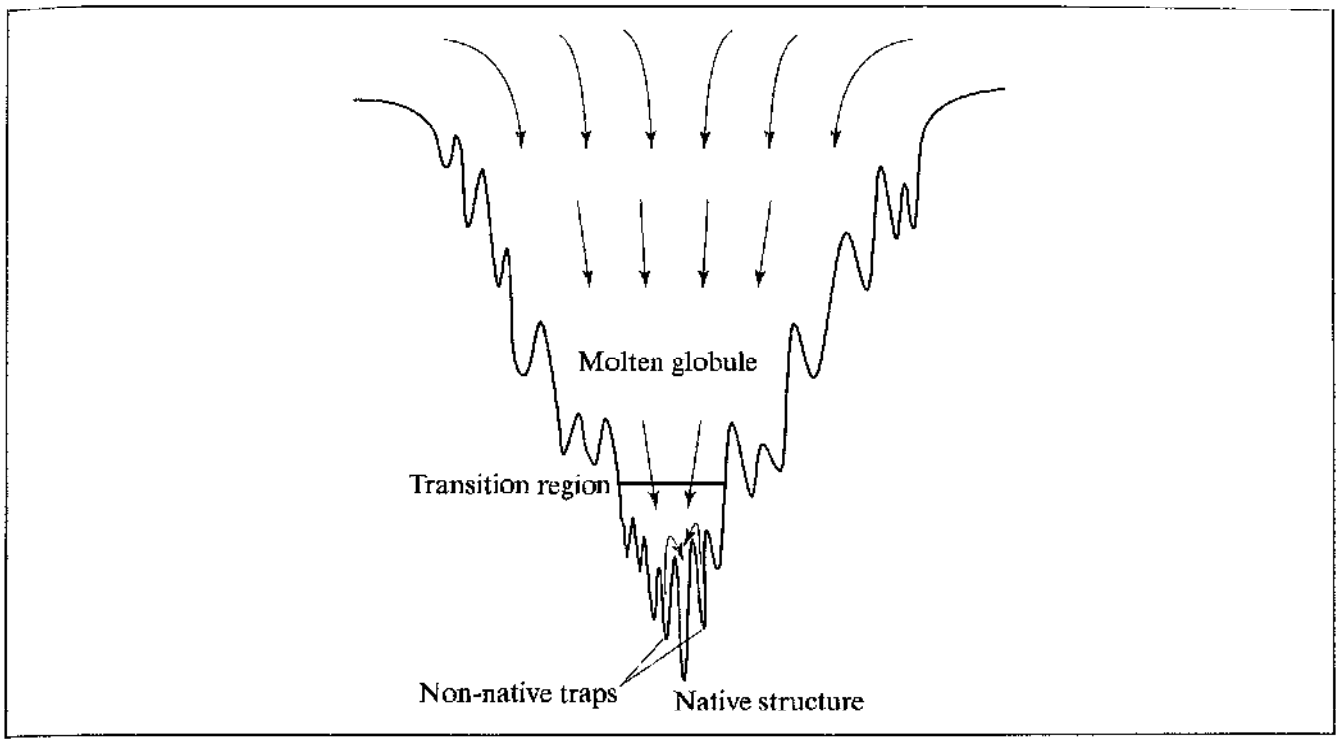


Fig. 10.27 Schematic representation of the energy landscape for protein folding. (Figure adapted from Onuchic J N, Z Luthey-Schulten and P Wolynes 1997 Theory of Protein Folding The Energy Landscape Perspective Annual Reviews in Physical Chemistry *48.545–600.)*

5 million possible structures for this system, 51 704 of which are unique (unrelated by rotation, reflection or reverse-labelling symmetries). In addition, it is estimated that there are about 10^{18} different non-compact arrangements.

In one study, Monte Carlo simulations were used to explore the conformational space of several hundred such sequences. The interactions between pairs of residues were selected at random from a Gaussian distribution [Šali *et al.* 1994a, b]. As such, this model corresponds to a heteropolymer with a random sequence of monomers of many different types. It was found that some sequences were able to find the global energy minimum (the native state) within a relatively short number of steps, whereas others did not. The key difference between the folding and non-folding sequences was the presence in the folding sequences of a pronounced global energy minimum, with a relatively large energy separation to the next lowest state. A three-stage folding pathway was suggested from these studies. The first stage involves a rapid collapse to a semi-compact random globule which contains about 30% of the contacts observed in the global minimum. In the second, rate-limiting, stage the protein searches for a transition state. There are about 1000 transition states which are structurally similar to the native state, having 80–95% of the native contacts. In the third stage, the chain rapidly progresses from one of the transition states to the native conformation. The transition region is key to the folding mechanism as it enables the search time to be reduced to a realistic value.

A related study was the exhaustive enumeration of the global minimum energy structures for all 2^{27} possible sequences of the 'HP' model described in Section 10.3.1 [Li *et al* 1996]. This showed that 4.75% of these sequences have a unique ground state (i.e. just one

conformation of the polymer on the lattice gives rise to the minimum energy). From this data it was also possible to determine how many sequences had a given structure as their unique ground state. Some structures are adopted by many sequences (the best being represented by 3794 sequences) whereas other structures correspond to the ground state of just a few sequences. Of additional interest was the fact that these 'highly designable' structures tended to have a larger gap between the global energy minimum and the next most stable structure, as found by the earlier studies. Of course, one could argue that it is inappropriate to extrapolate from the results of a small $3 \times 3 \times 3$ cube to 'real' proteins, and some workers consider the HP interaction model to be too simplistic. Nevertheless, such investigations invariably have the benefit of stimulating debate, which often leads to further advances.

Atomistic simulations of protein unfolding have most frequently been performed using high-temperature molecular dynamics. As we have already noted, the results from such calculations might not always be directly relevant to the physiological folding mechanism. However, simulations under other denaturing conditions such as high or low pH or non-aqueous solvents or even high-concentration urea (a common laboratory denaturant) are also possible. Most simulations are for a time of at least 1 ns; some are performed using explicit solvent, whilst others use an implicit solvent model. Some of the most interesting and fruitful work in this field is on systems that have also been studied experimentally, with NMR spectroscopy being a particularly widely used technique. The NMR data from an unfolded protein is usually more difficult to interpret than for a folded protein, and simulations can be useful in helping its interpretation. Two examples where this was possible are the characterisation of partially unfolded states of ubiquitin in 60% methanol [Alonso and Daggett 1995] and in thermally denatured barnase [Bond *et al.* 1997].

The ultimate goal for some in this area is a full atomistic simulation of the folding process, starting from an arbitrary structure, with explicit representation of the solvent. Such simulations are currently at the very limit of what can be achieved due to the length of simulation required and the large number of particles involved. One example of the current state-of-the-art is the 1 μs simulation of a 36-residue peptide starting from a fully extended state [Duan and Kollman 1998]. This peptide is one of the smallest proteins that can fold autonomously, with folding estimated to take between 10 μs and 100 μs. It contains three short α-helices. The simulation involved in addition to the protein about 3000 water molecules and was performed in a truncated octahedron simulation box with a time step of 2 fs. About four months of computing time on a 256 massively parallel supercomputer was required for the 1 μs simulation. Whilst the protein did not actually fold into the known experimental structure, a marginally stable state which showed significant resemblance to the native conformation was observed. This state had a lifetime of about 150 ns. A variety of metrics were used to monitor the simulation, including the RMS deviation to the experimental structure, the radius of gyration, the fraction of native contacts present and the solvation free energy. A high degree of fluctuation in all of these features was observed, characteristic of a relatively shallow free-energy landscape. Cluster analysis of the trajectory was used to identify the major conformations visited during the simulation and also to characterise the pathways between these conformations. Ready transitions were observed between the early states especially, giving a 'tangled' network of pathways. As computer power increases we are likely to see more studies of this type

Appendix 10.1 Some Common Abbreviations and Acronyms Used in Bioinformatics

This is necessarily an incomplete list; more comprehensive glossaries can be found elsewhere (particularly on the World Wide Web).

A,G,C,T, (U)	Adenine, guanine, cytosine, thymine – the four bases present in DNA. Uracil replaces thymine in RNA
Bp	Base pair
cDNA	Complementary DNA, synthesised from messenger RNA
Chromosome	Discrete unit of the genome consisting of a single molecule of DNA that carries many genes
Clone	Genetically identical copy (of a gene, cell or organism)
Codon	Sequence of three nucleotides that codes for a single amino acid (or a termination signal)
Contig	A group of pieces of DNA, derived from a cloning experiment (often a series of ESTs, see below), that represent overlapping regions of a chromosome
Deletion	One or more nucleotides that are not copied during DNA replication
DNA	Deoxyribose nucleic acid
Domain	Sequence of a polypeptide chain that can independently fold into a stable three-dimensional structure
Dynamic programming	Technique widely used in sequence alignment
EST	Expressed sequence tag. An EST is a partial sequence (typically less than 400 bases) selected from cDNA and used to identify genes expressed in a particular tissue
Eukaryote	Organism whose cells have a discrete nucleus and other subcellular compartments (*cf.* prokaryote)
Exon	Translated sequence of DNA
Gap	A break in a DNA or protein sequence which enables two or more sequences to be aligned
Gene	A sequence of DNA at a particular position on a specific chromosome that encodes a precise functional product (usually protein)
Genome	All of the genetic material in the chromosomes of an organism
Indel	Insertion or deletion required to optimise sequence alignment
Intron	Non-translated sequence of DNA
Kb	Kilobase – one thousand nucleotide bases
ktup	*k*-tuple. Parameter used in FASTA and FASTP sequence-alignment methods
Mb	Megabase – one million nucleotide bases

Appendix 10.1 Continued

mRNA	Messenger RNA
Mutation	A change in the DNA sequence
Nucleotide	Three components that make up the basic building block in DNA and RNA: a nitrogenous base (A, T, G, C, U), a phosphate and a sugar
Oligonucleotide	A molecule composed of a small number of nucleotides
Orthologue	Homologous proteins that perform the same function in different organisms
ORF	Open Reading Frame – region of DNA that is transcribed into RNA. Delineated by an initiator codon at one end and a stop codon at the other end
PAM	Point Accepted Mutation per 100 residues
Paralogue	Homologous proteins that perform different but related functions in one organism
PCR	Polymerase Chain Reaction. Widely used method for amplifying a DNA base sequence
Polymorphism	Differences in DNA sequence between individuals
Prokaryote	Organism lacking a nucleus and subcellular compartments (*cf* eukaryote). Includes bacteria and viruses
RNA	Ribonucleic acid
SNP	Single Polynucleotide Polymorphism – single base-pair variations in DNA
STS	Sequence tagged site. A short DNA sequence that occurs just once in the human genome and whose location and base sequence are known
Transcription	First step in gene expression, corresponding to the generation of mRNA from the original DNA
Translation	Second step in gene expression, the synthesis of proteins from mRNA
tRNA	Transfer RNA

Appendix 10.2 Some of the Most Common Sequence and Structural Databases Used in Bioinformatics

GenBank (NCBI, USA) EMBL Nucleotide Sequence Database (Europe) DDBJ (Japan)	The three main nucleotide sequence databases, which are synchronised daily
PIR-International Protein Sequence Database	Redundant protein sequence database
Swiss-Prot, TrEMBL	Annotated non-redundant protein sequence database. TrEMBL is a computer-annotated supplement to Swiss-Prot. TrEMBL contains the translations of all coding sequences present in the EMBL Nucleotide Sequence Database which are not yet integrated into Swiss-Prot
GenPept	Compendium of amino acid translations derived from GenBank
PDB, NRL3D	Protein Data Bank – protein structures (mostly from X-ray crystallography). NRL3D is a derived sequence database in PIR format
SCOP	Structural Classification of Proteins. Hierarchical protein structure database
CATH, FSSP	Sequence-structure classification databases
Prosite	Motif database

Appendix 10.3 Mutation Probability Matrix for 1 PAM

		Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
Ala	A	867	2	9	10	3	8	17	21	2	6	4	2	6	2	22	35	32	0	2	18
Arg	R	1	914	1	0	1	10	0	0	10	3	1	19	4	1	4	6	1	8	0	1
Asn	N	4	1	822	36	0	4	6	6	21	3	1	13	0	1	2	20	9	1	4	1
Asp	D	6	0	42	859	0	6	53	6	4	1	0	3	0	0	1	4	3	0	0	1
Cys	C	1	1	0	0	973	0	0	0	1	1	0	0	0	0	1	5	1	0	3	2
Gln	Q	3	9	4	5	0	876	27	1	23	1	3	6	4	0	6	2	2	0	0	1
Glu	E	10	0	7	56	0	35	864	4	2	3	1	4	2	0	3	4	2	0	1	2
Gly	G	21	1	12	11	1	2	7	935	1	0	1	2	2	1	3	21	3	0	0	5
His	H	1	8	18	3	1	20	1	0	912	0	1	1	0	2	3	1	1	1	4	1
Ile	I	2	2	3	1	2	1	2	0	0	872	9	2	12	7	0	1	7	0	1	32
Leu	L	3	1	3	0	0	6	1	1	4	22	947	2	45	13	3	1	3	4	2	15
Lys	K	2	37	25	6	0	12	7	2	2	4	1	925	19	0	3	8	11	0	1	1
Met	M	1	1	0	0	0	2	0	0	0	5	8	4	875	1	0	1	2	0	0	4
Phe	F	1	1	1	0	0	0	0	1	2	8	6	0	4	945	0	2	1	3	28	0
Pro	P	13	5	2	1	1	8	3	2	5	1	2	2	1	1	925	12	4	0	0	2
Ser	S	28	11	34	7	11	4	6	16	2	2	1	7	4	3	17	840	38	5	2	2
Thr	T	22	2	13	4	1	3	2	2	1	11	2	8	6	1	5	32	871	0	2	9
Trp	W	0	2	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	976	1	0
Tyr	Y	1	0	3	0	3	0	1	0	4	1	1	0	0	21	0	1	1	2	945	1
Val	V	13	2	1	1	3	2	2	3	3	57	11	1	17	1	3	2	10	0	2	902

Each element M_{ij} of this matrix corresponds to the probability that the amino acid in column i will mutate to the amino acid in row j after a period of 1 PAM The values have been multiplied by 10 000. (Based on Dayhoff M O 1978 *Atlas of Protein Sequence and Structure* Volume 5 Supplement 3. Dayhoff M O (Editor) Georgetown University Medical Center, National Biomedical Research Foundation Figure 82)

Appendix 10.4 Mutation Probability Matrix for 250 PAM

		Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
Ala	A	13	6	9	9	5	8	9	12	6	8	6	7	7	4	11	11	11	2	4	9
Arg	R	3	17	4	3	2	5	3	2	6	3	2	9	4	1	4	4	3	7	2	2
Asn	N	4	4	6	7	2	5	6	4	6	3	2	5	3	2	4	5	4	2	3	3
Asp	D	5	3	8	11	1	7	10	5	6	3	2	5	3	1	4	5	5	1	2	3
Cys	C	2	1	1	1	52	1	1	2	2	2	1	1	1	1	2	3	2	1	4	2
Gln	Q	3	5	5	6	1	10	7	3	8	2	3	5	3	1	4	3	3	1	2	2
Glu	E	5	4	7	11	1	9	12	5	6	3	2	5	3	1	4	5	5	1	2	3
Gly	G	12	5	10	10	4	7	9	27	5	5	4	6	5	3	8	11	9	2	3	7
His	H	2	5	5	4	2	7	4	2	15	2	2	3	2	2	3	3	2	2	3	2
Ile	I	3	2	2	2	2	2	2	2	2	10	6	2	6	5	2	3	4	1	3	9
Leu	L	6	4	4	3	2	6	4	3	5	15	34	4	20	13	5	4	6	6	7	13
Lys	K	6	18	10	8	2	10	8	5	8	5	4	24	9	2	6	8	8	4	3	5
Met	M	1	1	1	1	0	1	1	1	1	2	3	2	7	2	1	1	1	1	1	2
Phe	F	2	1	2	1	1	1	1	1	3	5	6	1	4	32	1	2	2	4	20	3
Pro	P	7	5	5	4	3	5	4	5	5	3	3	4	3	2	19	6	5	1	2	4
Ser	S	9	6	8	7	7	6	7	9	6	5	4	7	5	3	9	10	9	4	4	6
Thr	T	8	5	6	6	4	5	5	6	4	6	4	6	5	3	6	8	11	2	3	6
Trp	W	0	2	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	55	1	0
Tyr	Y	1	1	2	1	3	1	1	1	3	2	2	1	2	15	1	2	2	3	31	2
Val	V	7	4	4	4	4	4	4	5	4	15	10	4	10	5	5	5	7	2	4	17

Each element M_{ij} of this matrix corresponds to the probability that the amino acid in column i will mutate to the amino acid in row j after a period of 250 PAM. The values have been multiplied by 100. (Based on Dayhoff M O 1978. *Atlas of Protein Sequence and Structure* Volume 5 Supplement 3 Dayhoff M O (Editor) Georgetown University Medical Center, National Biomedical Research Foundation: Figure 83.)

Further Reading

Altschul S F 1996. Sequence Comparison and Alignment. In Sternberg M E (Editor) *Protein Structure Prediction – A Practical Approach* Oxford, IRL Press, pp. 137–167.

Altschul S F, M S Boguski, W Gish and J C Wootton 1994. Issues in Searching Molecular Sequence Databases. *Nature Genetics* 6:119–129.

Attwood T K and D J Parry-Smith 2000 *Introduction to Bioinformatics* Harlow, Addison Wesley Longman.

Barton G J 1996. Protein Sequence Alignment and Database Scanning. In Sternberg M E (Editor) *Protein Structure Prediction – A Practical Approach* Oxford, IRL Press, pp 31–63

Barton G J 1998. Protein Sequence Alignment Techniques. *Acta Crystallographica* D54:1139–1146

Blundell T L, B L Sibanda, M J E Sterbnerg and J M Thornton Knowledge-based Prediction of Protein Structures and the Design of Novel Molecules. *Nature* **326** 347–352

Branden C and J Tooze 1991. *Introduction to Protein Structure.*
Chatfield C and A J Collins 1980 *Introduction to Multivariate Analysis* London, Chapman & Hall.
Dobson C M, A Šali and M Karplus 1998. Protein Folding: A Perspective from Theory and Experiment *Angewandte Chemie International Edition* **37**.868–893
Perutz M 1992. *Protein Structure New Approaches to Disease And Therapy* New York, W H Freeman
Schulz G E and R H Schirmer 1979. *Principles of Protein Structure.* New York, Springer-Verlag

References

Alonso D O V and V Daggett 1995 Molecular Dynamics Simulations of Protein Unfolding and Limited Refolding. Characterisation of Partially Unfolded States of Ubiquitin in 60% Methanol and in Water *Journal of Molecular Biology* **247**:501–520.

Altschul S F, W Gish, W Miller, E W Myers and D J Lipman 1990. Basic Local Alignment Search Tool *Journal of Molecular Biology* **215**·403–410.

Altschul S F, T L Madden, A A Schäffer, J Zhang, Z Zhang, W Miller and D J Lipman 1997. Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs. *Nucleic Acids Research* **25** 3389–3402.

Bernstein F C, T F Koetzle, G J B Williams, E Meyer, M D Bryce, J R Rogers, O Kennard, T Shikanouchi and M Tasumi 1977 The Protein Data Bank: A Computer-Based Archival File for Macromolecular Structures. *Journal of Molecular Biology* **112**:535–542

Birktoft J J and D M Blow 1972. The structure of Crystalline Alpha-Chymotrypsin V. The Atomic Structure of Tosyl-Alpha-Chymotrypsin at 2 Ångstroms Resolution *Journal of Molecular Biology* **68**·187–240

Bond C J, K-B Wong, J Clarke, A R Fersht and V Daggett 1997 Characterisation of Residual Structure in the Thermally Denatured State of Barnase by Simulation and Experiment. Description of the Folding Pathway. *Proceedings of the National Academy of Sciences USA* **94** 13409–13413.

Bowie J U, R Lüthy and D Eisenberg 1991 A Method to Identify Protein Sequences that Fold into a Known Three-Dimensional Structure. *Science* **253**:164–170

Brenner S E, C Chothia and T J P Hubbard 1997. Population Statistics of Protein Structures: Lessons from Structural Classifications. *Current Opinion in Structural Biology* **7** 369–376.

Bruccoleri R E and M Karplus 1985. Chain Closure with Bond Angle Variations. *Macromolecules* **18**:2767–1773.

Bruccoleri R E and M Karplus 1987. Prediction of the Folding of Short Polypeptide Segments by Uniform Conformational Sampling. *Biopolymers* **26**:137–168.

Bryant S H and C E Lawrence 1993. An Empirical Energy Function for Threading Protein Sequences Through the Folding Motif. *Proteins: Structure, Function and Genetics* **16**·92–112

Bryngelson J D, J N Onuchic, N D Socci and P G Wolynes 1995. Funnels, Pathways, and the Energy Landscape of Protein Folding A Synthesis. *Proteins: Structure, Function and Genetics* **21**.167–195.

Bryngelson J D and P G Wolynes 1987. Spin Glasses and the Statistical Mechanics of Protein Folding. *Proceedings of the National Academy of Sciences USA* **84**·7524–7528.

Chan H S and K A Dill 1993. The Protein Folding Problem. *Physics Today* **Feb**·24–32.

Chothia C and A M Lesk 1986. The Relation Between the Divergence of Sequence and Structure in Proteins. *EMBO Journal* **5**·823–826.

Chothia C, A M Lesk, A Tramontano, M Levitt, S J Smith-Gill, G Air, S Sheriff, E A Padlan and D Davies 1989. Conformations of Immunoglobulin Hypervariable Regions *Nature* **342**:877–883

Chou P Y and G D Fasman 1978. Prediction of the Secondary Structure of Proteins from Their Amino Acid Sequence. *Advances in Enzymology* **47**:45–148.

Cohen F E and S R Presnell 1996. The Combinatorial Approach In Sternberg M J E (Editor) *Protein Structure and Prediction*. Oxford, IRL Press, pp. 207–227.

Cohen F E, M J E Sternberg and W R Taylor 1982 Analysis and Prediction of the Packing of α-Helices against a β-Sheet in the Tertiary Structure of Globular Proteins. *Journal of Molecular Biology* **156** 821–862.

Cuff J A and G J Barton 1999. Evaluation and Improvement of Multiple Sequence Methods for Protein Secondary Structure Prediction. *Proteins: Structure, Function and Genetics* **34**:508–519

Dayhoff M O 1978. A Model of Evolutionary Change. In Dayhoff M O (Editor) *Proteins in Atlas of Protein Sequence and Structure* Volume 5 Supplement 3 Georgetown University Medical Center, National Biomedical Research Foundation, pp. 345–358

Duan Y and P A Kollman 1998 Pathways to a Protein Folding Intermediate Observed in a 1-Microsecond Simulation in Aqueous Solution. *Science* **282**:740–744.

Dunbrack R L Jr and M Karplus 1993. Backbone-dependent Rotamer Library for Proteins. Application to Side-chain Prediction. *Journal of Molecular Biology* **230**:543–574

Eddy S R 1996 Hidden Markov Models. *Current Opinion in Structural Biology* **6**:361–365.

Finkelstein A V 1997. Can Protein Unfolding Simulate Protein Folding? *Protein Engineering* **10**:843–845.

Frazao C, C Topham, V Dhanaraj and T L Blundell 1994. Comparative Modelling of Human Renin: A Retrospective Evaluation of the Model with Respect to the X-ray Crystal Structure. *Pure and Applied Chemistry* **66** 43–50.

Garnier J, D Osguthorpe and B Robson 1978. Analysis of the Accuracy and Implications of Simple Methods for Predicting the Secondary Structure of Globular Proteins *Journal of Molecular Biology* **120**:97–120.

Gibson K D and H A Scheraga 1987. Revised Algorithms for the Build-up Procedure for Predicting Protein Conformations by Energy Minimization *Journal of Computational Chemistry* **8**:826–834.

Go N and H A Scheraga 1970 Ring Closure and Local Conformational Deformations of Chain Molecules *Macromolecules* **3**:178–187

Godzik A, A Kolinski and J Skolnick 1993 *De Novo* and Inverse Folding Predictions of Protein Structure and Dynamics *Journal of Computer-Aided Molecular Design* **7**:397–438

Godzik A, J Skolnick and A Kolinski 1992 Simulations of the Folding Pathway of Triose Phosphate Isomerase-type α/β Barrel Proteins. *Proceedings of the National Academy of Sciences USA* **89**:2629–2633.

Gonnet G H, M A Cohen and S A Benner 1992. Exhaustive Matching of the Entire Protein Sequence Database. *Science* **256**:1443–1445.

Gribskov M, A D McLachlan and D Eisenberg 1987. Profile Analysis: Detection of Distantly Related Proteins. *Proceedings of the National Academy of Sciences USA* **84**:4335–4358.

Havelka W A, R Henderson and D Oesterhelt 1995 3-Dimensional Structure of Halorhodopsin at 7-Ångstrom Resolution *Journal of Molecular Biology* **247**:726–738

Henderson R, J M Baldwin, T A Ceska, F Zemlin, E Beckmann and K H Downing 1990. Model for the Structure of Bacteriorhodopsin Based on High-resolution Electron Cryo-microscopy. *Journal of Molecular Biology* **213**:899–929.

Henikoff S and J G Henikoff 1992. Amino Acid Substitution Matrices from Protein Blocks *Proceedings of the National Academy of Sciences USA* **89**:10915–10919.

Holm L and C Sander 1993. Protein Structure Comparison by Alignment of Distance Matrices. *Journal of Molecular Biology* **233**:123–138.

Holm L and C Sander 1994. The FSSP Database of Structurally Aligned Protein Fold Families *Nucleic Acids Research* **22** 3600–3609.

Holm L and C Sander 1999 Protein Folds and Families: Sequence and Structure Alignments *Nucleic Acids Research* **27**:244–247

Jernigan R L and I Bahar 1996. Structure-derived Potentials and Protein Simulations *Current Opinion in Structural Biology* **6**:195–209.

Jones D and J Thornton 1993 Protein Fold Recognition *Journal of Computer-Aided Molecular Design* **7**:439–456

Jones D T, W R Taylor and J M Thornton 1992. A New Approach to Protein Fold Recognition *Nature* **358**.86–89

Jones D T and J M Thornton 1996 Potential Energy Functions for Threading. *Current Opinion in Structural Biology* **6**:210–216.

Jones T A and S Thirup 1986 Using Known Substructures in Protein Model Building and Crystallography *EMBO Journal* **5**:819–822

King, R D, M Saqi, R Sayle and M J E Sternberg 1997. DSC: Public Domain Protein Secondary Structure Prediction. *Computer Applications in the Biosciences* **13** 473–474

Koppensteiner W A, P Lackner, M Wiederstein and M J Sippl 2000. Characterization of Novel Proteins Based on Known Protein Structures *Journal of Molecular Biology* **296**.1139–1152

Kovacs H, A E Mark and W F van Gunsteren 1997 Solvent Structure at a Hydrophobic Protein Surface. *Proteins. Structure, Function and Genetics* **27**:395–404.

Krogh A, M Brown, S Mian, K Sjölander and D Haussler 1994. Hidden Markov Models in Computational Biology Applications to Protein Modeling. *Journal of Molecular Biology* **235** 1501–1531.

Levinthal C 1969. In Debrunner P, J C M Tsibris and E Munck (Editors) *Mössbauer Spectroscopy in Biological Systems*, Proceedings of a Meeting held at Allerton House, Monticello, Illinois, University of Illinois Press, Urbana, p 22

Levitt M 1976. A Simplified Representation of Protein Conformations for Rapid Simulation of Protein Folding *Journal of Molecular Biology* **104** 59–107

Levitt M 1992 Accurate Modeling of Protein Conformation by Automatic Segment Matching. *Journal of Molecular Biology* **226**.507–533

Li H, R Helling, C Tang and N Wingreen 1996. Emergence of Preferred Structures in a Simple Model of Protein Folding *Science* **273**:666–669

Li Z Q and H A Scheraga 1987 Monte Carlo Minimization Approach to the Multiple Minima Problem in Protein Folding *Proceedings of the National Academy of Sciences USA* **84**.6611–6615.

Lipman, D J and W R Pearson 1985 Rapid and Sensitive Protein Similarity Searches. *Science* **227**:1435–1441

Luthy R, J U Bowie and D Eisenberg 1992. Assessment of Protein Models with Three-Dimensional Profiles *Nature* **356**.83–85

Maiorov V N and G M Crippen 1994. Learning About Protein Folding via Potential Functions. *Proteins. Structure, Function and Genetics* **20**.167–173.

Mosimann S, S Meleshko and M N G Jones 1995 A Critical Assessment of Comparative Molecular Modeling of Tertiary Structures of Proteins. *Proteins: Structure, Function and Genetics* **23**:301–317.

Moult J, T Hubbard, K Fidelis and J T Pedersen 1999. Critical Assessment of Methods of Protein Structure Prediction (CASP) Round III. *Proteins: Structure, Function and Genetics* Suppl. 3:2–6.

Murzin A G, S E Brenner, T Hubbard and C Chothia 1995. SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures. *Journal of Molecular Biology* **247**:536–540

Needleman S B and C D Wunsch 1970 A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins. *Journal of Molecular Biology* **48**:443–453.

Ning Q and T J Sejnowski 1988 Predicting the Secondary Structure of Globular Proteins Using Neural Network Models. *Journal of Molecular Biology* **202**:865–888.

Noble M E M, R K Wierenga, A-M Lambeir, F R Opperdoes, W H Thunnissen, K H Kalk, H Groendijk and W G J Hol 1991. The Adaptability of the Active Site of Trypanosomal Triosephosphate Isomerase as Observed in the Crystal Structures of Three Different Complexes. *Proteins: Structure, Function and Genetics* **10**:50–69

Novotny J, A A Rashin and R E Bruccoleri 1988 Criteria that Discriminate between Native Proteins and Incorrectly Folded Models. *Proteins: Structure, Function and Genetics* 4:19–30

Onuchic J N, Z Luthey-Schulten and P Wolynes 1997. Theory of Protein Folding. The Energy Landscape Perspective. *Annual Reviews in Physical Chemistry* **48** 545–600.

Orengo C A, N P Brown and W R Taylor 1992 Fast Structure Alignment for Protein Databank Searching *Proteins. Structure, Function and Genetics* **14**:139–167

Orengo C A and W R Taylor 1990 A Rapid Method of Protein Structure Alignment *Journal of Theoretical Biology* **147** 517–551.

Orengo C A and W R Taylor 1993 A Local Alignment Method for Protein Structure Motifs *Journal of Molecular Biology* **233**:488–497.

Orengo C A, F P Flores, W R Taylor and J M Thornton 1993 Identification and Classification of Protein Fold Families *Protein Engineering* **6**:485–500

Ortiz A R, A Kolinski and J Skolnick 1998 Fold Assembly of Small Proteins Using Monte Carlo Simulations Driven by Restraints Derived from Multiple Sequence Alignments. *Journal of Molecular Biology* **277**:419–446

Park B and M Levitt 1996. Energy Functions that Discriminate X-ray and Near-native Folds from Well-constructed Decoys. *Journal of Molecular Biology* **258** 367–392.

Pauling L, R B Corey and H R Bronson 1951 The Structure of Proteins Two Hydrogen-bonded Helical Configurations of the Polypeptide Chain *Proceedings of the National Academy of Sciences USA* **37**:205–211

Pearson W R 1990 Rapid and Sensitive Sequence Comparison with FASTP and FASTA. *Methods in Enzymology* **183**:63–98

Pearson W R and D J Lipman 1988. Improved Tools for Biological Sequence Comparison. *Proceedings of the National Academy of Sciences USA* **85**:2444-2448.

Peitsch M C 1996. ProMod and Swiss-Model: Internet-based Tools for Automated Comparative Protein Modelling. *Biochemical Society Transactions* **24**:274–279

Ponder J W and F M Richards 1987 Tertiary Templates for Proteins Use of Packing Criteria in the Enumeration of Allowed Sequences for Different Structural Classes *Journal of Molecular Biology* **193**.775–791.

Rabiner L R 1989 A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE* **77**:257–286.

Ripoll D R and H A Scheraga 1988 On the Multiple-Minimum Problem in the Conformational Analysis of Polypeptides. II. An Electrostatistically Driven Monte Carlo Method: Tests on Poly(L-Alanine). *Biopolymers* **27**:1283–1303

Ripoll D R and H A Scheraga 1989. On the Multiple-Minimum Problem in the Conformational Analysis of Polypeptides. III. An Electrostatically Driven Monte Carlo Method: Tests on met-Enkephalin. *Journal of Protein Chemistry* **8**:263–287

Rost B and C Sander 1993 Prediction of Protein Secondary Structure at Better than 70% Accuracy *Journal of Molecular Biology* **232**:584–599.

Šali A 1995. Modelling Mutations and Homologous Proteins *Current Opinion in Biotechnology* **6** 437–451

Šali A and T L Blundell 1993 Comparative Protein Modelling by Satisfaction of Spatial Restraints *Journal of Molecular Biology* **234**.779–815.

Šali A, E Shakhnovich and M Karplus 1994a. How Does a Protein Fold? *Nature* **369**.248–251

Šali A, E Shakhnovich and M Karplus 1994b. Kinetics of Protein Folding. A Lattice Model Study of the Requirements for Folding to the Native State. *Journal of Molecular Biology* **235**.1614–1636

Sánchez R and A Šali 1998 Large-scale Protein Structure Modelling of the *Saccharomyces cerevisiae* Genome *Proceedings of the National Academy of Sciences USA* **95**:13597–13602.

Scheraga H A 1993. Searching Conformational Space. In van Gunsteren W F, P K Weiner and A J Wilkinson (Editors) *Computer Simulation of Biomolecular Systems* Volume 2 Leiden, ESCOM

Shenkin P S, D L Yarmusch, R M Fine, H Wang and C Levinthal 1987. Predicting Antibody Hypervariable Loop Conformation. I Ensembles of Random Conformations for Ringlink Structures. *Biopolymers* **26**:2053–2085.

Simons K T, R Bonneau, I Ruszinski and D Baker 1999b. *Ab Initio* Protein Structure Prediction of CASP III Targets Using ROSETTA. *Proteins Structure, Function and Genetics Supplement* **3** 171–176.

Simons K T, C Kooperberg, E Huang and D Baker 1997 Assembly of Protein Tertiary Structures from Fragments with Similar Local Sequences Using Simulated Annealing and Bayesian Scoring Functions. *Journal of Molecular Biology* **268**:209–225

Simons K T, I Ruczinski, C Kooperberg, B A Cox, C Bystroff and D Baker 1999a Improved Recognition of Native-Like Protein Structures Using a Combination of Sequence-Dependent and Sequence-Independent Features of Proteins *Proteins Structure, Function and Genetics* **34**.82–95

Sippl M J 1990 Calculation of Conformational Ensembles from Potentials of Mean Force. An Approach to the Knowledge-Based Prediction of Local Structures in Globular Proteins *Journal of Molecular Biology* **213**.859–883.

Skolnick J, A Kolinski and A R Ortiz 1997 MONSSTER: A Method for Folding Globular Proteins with a Small Number of Distance Restraints *Journal of Molecular Biology* **265**:217–241.

Smith T F and M S Waterman 1981. Identification of Common Molecular Subsequences. *Journal of Molecular Biology* **147**.195–197.

Srinivasan N, K Gurprasad and T L Blundell 1996. Comparative Modelling of Proteins. In Sternberg M E (Editor) *Protein Structure Prediction – A Practical Approach.* Oxford, IRL Press, pp 111–140.

Sternberg M J E, F E Cohen and W R Taylor 1982 A Combinatorial Approach to the Prediction of the Tertiary Fold of Globular Proteins. *Biochemical Society Transactions* **10**.299–301.

Summers N L, W D Carlson and M Karplus 1987 Analysis of Side-Chain Orientations in Homologous Proteins *Journal of Molecular Biology* **196** 175–198

Taylor W R and C A Orengo 1989 Protein Structure Alignment *Journal of Molecular Biology* **208** 1–22.

Thompson J D, D G Higgins and T J Gibson 1994 CLUSTAL W: Improving the Sensitivity of Progressive Multiple Sequence Alignment Through Sequence Weighting, Position-specific Gap Penalties and Weight Matrix Choice *Nucleic Acids Research* **22**.4673–4680

Turk D, H W Hoeffken, D Grosse, J Stuerzebecher, P D Martin, B F P Edwards and W Bode 1992 Refined 2 3 Ångstroms X-Ray Crystal Structure of Bovine Thrombin Complexes Formed with the 3 Benzamidine and Arginine-Based Thrombin Inhibitors NAPAP, 4-TAPAP and MQPA. A Starting Point for Improving Antithrombotics. *Journal of Molecular Biology* **226**:1085–1099

Turk D, J Sturzebecher and W Bode 1991 Geometry of Binding of the *N*-Alpha-Tosylated Piperidides of *meta*-Amidino-Phenylalanine, Para Amidino-Phenylalanine and *para*-Guanidino-Phenylalanine to Thrombin and Trypsin – X-ray Crystal Structures of Their Trypsin Complexes and Modeling of their Thrombin Complexes *FEBS Letters* **287**:133–138.

Vasquez M 1996. Modeling Side-chain Conformation. *Current Opinion in Structural Biology* **6**:217–221

Wilmot C M and J M Thornton 1988. Analysis and Prediction of the Different Types of β-turn in Proteins. *Journal of Molecular Biology* **203**:221–232.

Zarembinski T I, L-W Hung, H-J Mueller-Dieckmann, K-K Kim, H Yokota, R Kim and S-H Kim 1998 Structure-based Assignment of the Biochemical Function of a Hypothetical Protein: A Test Case of Structural Genomics. *Proceedings of the National Academy of Sciences USA* **95**:15189–15193.

CHAPTER ELEVEN

Four Challenges in Molecular Modelling: Free Energies, Solvation, Reactions and Solid-state Defects

In this chapter we shall consider four important problems in molecular modelling. First, we discuss the problem of calculating free energies. We then consider continuum solvent models, which enable the effects of the solvent to be incorporated into a calculation without requiring the solvent molecules to be represented explicitly. Third, we shall consider the simulation of chemical reactions, including the important technique of *ab initio* molecular dynamics. Finally, we consider how to study the nature of defects in solid-state materials.

11.1 Free Energy Calculations

11.1.1 The Difficulty of Calculating Free Energies by Computer

The free energy is often considered to be the most important quantity in thermodynamics. The free energy is usually expressed as the Helmholtz function, A, or the Gibbs function, G. The Helmholtz free energy is appropriate to a system with constant number of particles, temperature and volume (constant NVT), whereas the Gibbs free energy is appropriate to constant number of particles, temperature and pressure (constant NPT). Most experiments are conducted under conditions of constant temperature and pressure, where the Gibbs function is the appropriate free energy quantity.

Unfortunately, the free energy is a difficult quantity to obtain for systems such as liquids or flexible macromolecules that have many minimum energy configurations separated by low-energy barriers. Associated quantities such as the entropy and the chemical potential are also difficult to calculate. As we showed in Section 6.3, the free energy cannot be accurately determined from a 'standard' molecular dynamics or Monte Carlo simulation, because such simulations do not adequately sample from those regions of phase space that make important contributions to the free energy. Specifically, we showed that the Helmholtz free energy is given by:

$$A = k_B T \ln \left(\iint d\mathbf{p}^N \, d\mathbf{r}^N \exp \left(\frac{+\mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{k_B T} \right) \rho(\mathbf{p}^N, \mathbf{r}^N) \right) \tag{11.1}$$

The term $\exp[+\mathscr{H}(\mathbf{p}^N,\mathbf{r}^N)/k_BT]$ makes important contributions to the integral. However, a simulation using either Monte Carlo or molecular dynamics sampling seeks out the *lower-energy* regions of phase space. Such simulations will never adequately sample the important high-energy regions and so to calculate the free energy using a conventional simulation will lead to poorly converged and inaccurate values. The grand canonical and particle-insertion methods do provide a route to the free energy, but they are not applicable to many of the systems of interest, which contain complex molecules at high densities.

11.2 The Calculation of Free Energy Differences

Let us consider a closely related but slightly different problem: the calculation of the free energy difference of two states. As an example, we will consider the problem of calculating the free energy difference between ethanol (CH_3CH_2OH) and ethane thiol (CH_3CH_2SH) in water. As we shall see, this is a problem that can be tackled using methods that use Monte Carlo or molecular dynamics sampling. Three methods have been proposed for calculating free energy differences: thermodynamic perturbation, thermodynamic integration and slow growth. We shall consider each of these in turn.

11.2.1 Thermodynamic Perturbation

Consider two well-defined states X and Y. For example, X could be a system comprising a molecule of ethanol in a periodic box of water and Y could be ethane thiol in water. X contains N particles interacting according to the Hamiltonian $\mathscr{H}_X$ Y contains N particles interacting according to $\mathscr{H}_Y$ The free energy difference (ΔA) between the two states is as follows:

$$\Delta A = A_Y - A_X = -k_BT\ln\frac{Q_Y}{Q_X} \tag{11.2}$$

$$\Delta A = -k_BT\left\{\frac{\int\int d\mathbf{p}^N\,d\mathbf{r}^N\exp[-\mathscr{H}_Y(\mathbf{p}^N,\mathbf{r}^N)/k_BT]}{\int\int d\mathbf{p}^N\,d\mathbf{r}^N\exp[-\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]}\right\} \tag{11.3}$$

Substituting 1 in the form $\exp[+\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]\exp[-\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]$ into the numerator gives:

$$\Delta A = -k_BT\left\{\frac{\iint d\mathbf{r}^N\,d\mathbf{p}^N\exp\left(-\frac{\mathscr{H}_Y(\mathbf{r}^N,\mathbf{p}^N)}{k_BT}\right)\exp\left(+\frac{\mathscr{H}_X(\mathbf{r}^N,\mathbf{p}^N)}{k_BT}\right)\exp\left(-\frac{\mathscr{H}_X(\mathbf{r}^N,\mathbf{p}^N)}{k_BT}\right)}{\iint d\mathbf{r}^N\,d\mathbf{p}^N\exp\left(\frac{-\mathscr{H}_X(\mathbf{r}^N,\mathbf{p}^N)}{k_BT}\right)}\right\} \tag{11.4}$$

Equation (11.4) can be written in terms of an ensemble average, as follows:

$$\begin{aligned}\Delta A &= -k_BT\left\{\frac{\int\int d\mathbf{p}^N\,d\mathbf{r}^N\exp[-\mathscr{H}_Y(\mathbf{p}^N,\mathbf{r}^N)/k_BT]\exp[+\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]\exp[-\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]}{\int\int d\mathbf{p}^N\,d\mathbf{r}^N\exp[-\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]}\right\}\\ &= -k_BT\langle\exp-[\mathscr{H}_Y(\mathbf{p}^N,\mathbf{r}^N)-\mathscr{H}_X(\mathbf{p}^N,\mathbf{r}^N)/k_BT]\rangle_0\end{aligned} \tag{11.5}$$

The subscript 0 indicates averaging over the ensemble of configurations representative of the initial state X. If the averaging is over the ensemble corresponding to the final state Y (indicated by the subscript 1) then we are effectively simulating the reverse process, from which ΔA can be determined by:

$$\Delta A = k_B \ln\langle \exp[-(\mathscr{H}_X - \mathscr{H}_Y)/k_B T]\rangle_1 \tag{11.6}$$

This approach to the calculation of free energy differences, Equation (11.6), is generally attributed to Zwanzig [Zwanzig 1954]. To perform a thermodynamic perturbation calculation we must first define $\mathscr{H}_Y$ and $\mathscr{H}_X$ and then run a simulation at the state X, forming the ensemble average of $\exp[-(\mathscr{H}_Y - \mathscr{H}_X)/k_B T]$ as we proceed. Analogously, we could run a simulation at the state Y and obtain the ensemble average of $\exp[-(\mathscr{H}_X - \mathscr{H}_Y)/k_B T]$. Thus, if X corresponds to ethanol and Y to ethane thiol, the free energy difference could be obtained from a simulation of ethanol in a periodic box of water as follows. For each configuration we calculate the value of the energy for every instantaneous conformation of ethanol in which the oxygen atom is temporarily assigned the potential energy parameters of sulphur. Alternatively, we could simulate ethane thiol and for each configuration calculate the energy of the system in which the sulphur is 'mutated' into oxygen.

If X and Y do not overlap in phase space then the value of the free energy difference calculated using Equation (11.6) will not be very accurate, because we will not adequately sample the phase space of Y when simulating X. This problem arises when the energy difference between the two states is much larger than $k_B T$: $|\mathscr{H}_Y - \mathscr{H}_X| \gg k_B T$. How then can we obtain accurate estimates of the free energy difference under such circumstances? Consider what happens if we introduce a state that is intermediate between X and Y, with a Hamiltonian $\mathscr{H}_1$ and a free energy $A(1)$:

$$\begin{aligned}
\Delta A &= A(Y) - A(X) \\
&= (A(Y) - A(1)) + (A(1) - A(X)) \\
&= -k_B T \ln\left[\frac{Q(Y)}{Q(1)} \cdot \frac{Q(1)}{Q(X)}\right] \\
&= -k_B T \ln\langle \exp[-(\mathscr{H}_Y - \mathscr{H}_1)/k_B T]\rangle - k_B T \ln\langle \exp[-(\mathscr{H}_1 - \mathscr{H}_X)/k_B T]\rangle
\end{aligned} \tag{11.7}$$

If we define region 1 so that it overlaps with X and Y we may improve the sampling and obtain a more reliable value. This is shown in Figure 11.1.

An obvious extension is to use several different intermediate states in progressing from $\mathscr{H}_X$ to $\mathscr{H}_Y$:

$$\begin{aligned}
\Delta A &= A(Y) - A(X) \\
&= (A(Y) - A(N)) + (A(N) - A(N-1)) + \cdots \\
&\quad + (A(2) - A(1)) + (A(1) - A(X)) \\
&= -k_B T \ln\left[\frac{Q(Y)}{Q(N)} \cdot \frac{Q(N)}{Q(N-1)} \cdot \frac{Q(N-1)}{Q(N-2)} \cdots \frac{Q(2)}{Q(1)} \frac{Q(1)}{Q(X)}\right]
\end{aligned} \tag{11.8}$$

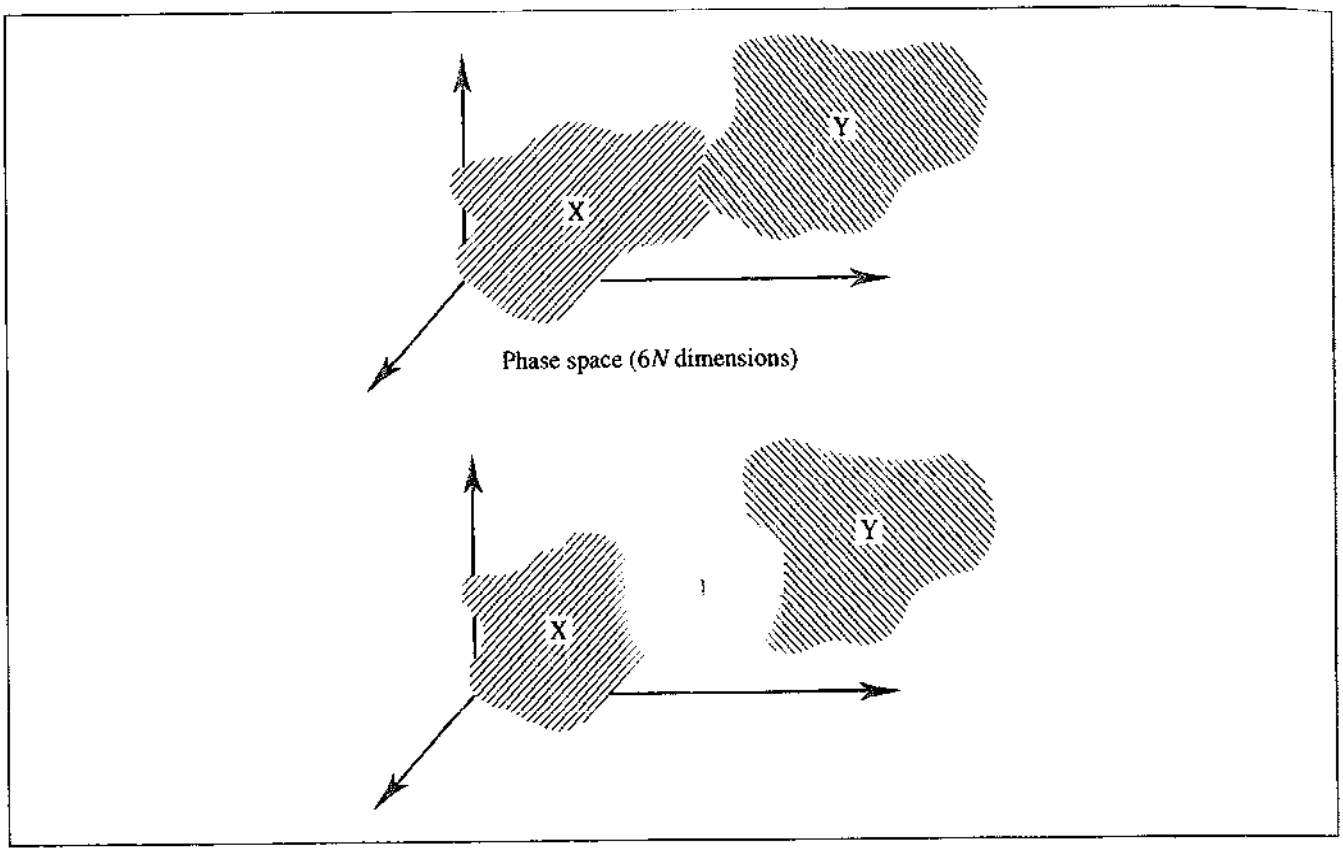


Fig 11.1 An intermediate state (labelled 1) can improve the degree of overlap in phase space and lead to improved sampling

The important point to notice is that the intermediate terms cancel out and so we are free to choose as many intermediate states as are necessary to get good overlaps and thus reliable values of the free energy differences.

11.2.2 Implementation of Free Energy Perturbation

Suppose we are using an empirical energy function such as the following to describe the inter- and intramolecular interactions in our ethanol/ethane thiol system:

$$\mathscr{V}(\mathbf{r}^N) = \sum_{\text{bonds}} \frac{k_i}{2}(l_i - l_{i,0})^2 + \sum_{\text{angles}} \frac{k_i}{2}(\theta_i - \theta_{i,0})^2 + \sum_{\text{torsions}} \frac{V_n}{2}(1 + \cos(n\omega - \gamma))$$
$$+ \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(4\varepsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} \right) \tag{11.9}$$

The force-field model for ethanol contains C–O and O–H bond-stretching contributions; in ethane thiol these are replaced by C–S and S–H parameters. Similarly, in ethanol there will be angle-bending terms due to C–O–H, C–C–O and H–C–O angles; in ethane thiol these will be C–S–H, C–C–S and H–C–S. The torsional contribution will be modified appropriately, as will the van der Waals and electrostatic interactions (both those within the

solute and between the solute and solvent). The partial atomic charges for all of the atoms in ethanol may all be different from those for ethane thiol.

The relationship between the initial, final and intermediate states is usefully described in terms of a *coupling parameter*, λ. As λ is changed from 0 to 1, the Hamiltonian varies from $\mathscr{H}_X$ to $\mathscr{H}_Y$. Each of the terms in the force field for an intermediate state λ can be written as a linear combination of the values for X and Y:

1. Bonds:
$$k_l(\lambda) = \lambda k_l(\mathrm{Y}) + (1-\lambda)k_l(\mathrm{X}) \tag{11.10}$$
$$l_0(\lambda) = \lambda l_0(\mathrm{Y}) + (1-\lambda)l_0(\mathrm{X}) \tag{11.11}$$
2. Angles:
$$k_\theta(\lambda) = \lambda k_\theta(\mathrm{Y}) + (1-\lambda)k_\theta(\mathrm{X}) \tag{11.12}$$
$$\theta_0(\lambda) = \lambda \theta_0(\mathrm{Y}) + (1-\lambda)\theta_0(\mathrm{X}) \tag{11.13}$$
3. Dihedrals:
$$v_\omega(\lambda) = \lambda v_\omega(\mathrm{Y}) + (1-\lambda)v_\omega(\mathrm{X}) \tag{11.14}$$
4. Electrostatics:
$$q_i(\lambda) = \lambda q_i(\mathrm{Y}) + (1-\lambda)q_i(\mathrm{X}) \tag{11.15}$$
5. van der Waals:
$$\varepsilon(\lambda) = \lambda \varepsilon(\mathrm{Y}) + (1-\lambda)\varepsilon(\mathrm{X}) \tag{11.16}$$
$$\sigma(\lambda) = \lambda \sigma(\mathrm{Y}) + (1-\lambda)\sigma(\mathrm{X}) \tag{11.17}$$

For each value of λ (λ_i) a simulation is performed (using either Monte Carlo or molecular dynamics as appropriate) with the appropriate force field parameters. First, the system is equilibrated using the force field parameters appropriate to λ_i. A production phase is then performed during which the free energy difference $\Delta A(\lambda_i \rightarrow \lambda_{i+1})$ is accumulated as $-k_\mathrm{B}T \ln\langle \exp(-\Delta\mathscr{H}_i/k_\mathrm{B}T)\rangle$, where $\Delta\mathscr{H}_i = \mathscr{H}_{i+1} - \mathscr{H}_i$. The total free energy change for $\lambda = 0$ to $\lambda = 1$ is then the sum of the free energy changes for the various values of λ_i, as shown in Figure 11.2.

The approach that we have described so far is known as *forward sampling*, because the free energy is determined for $\lambda_i \rightarrow \lambda_{i+1}$. In *backward sampling*, the free energy difference between λ_i and λ_{i-1} is determined. The coupling parameter λ still increases from 0 to 1; it is just that the free energies are accumulated in a different manner. In *double-wide sampling*, the free energy differences for both $\lambda_i \rightarrow \lambda_{i+1}$ and $\lambda_i \rightarrow \lambda_{i-1}$ are obtained from a simulation as

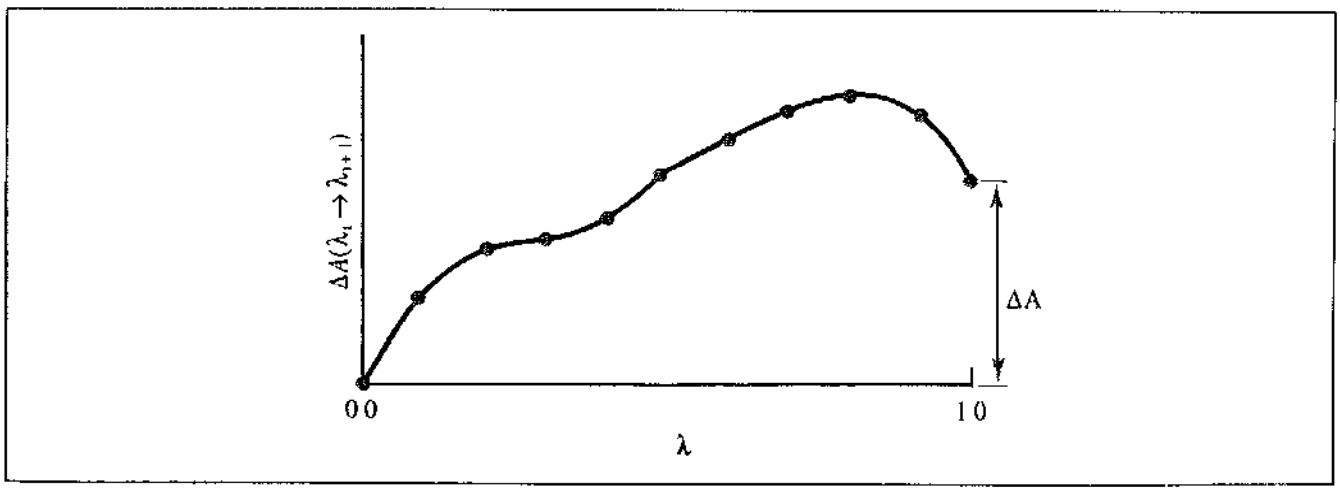


Fig. 11.2: Calculation of the free energy difference using perturbation

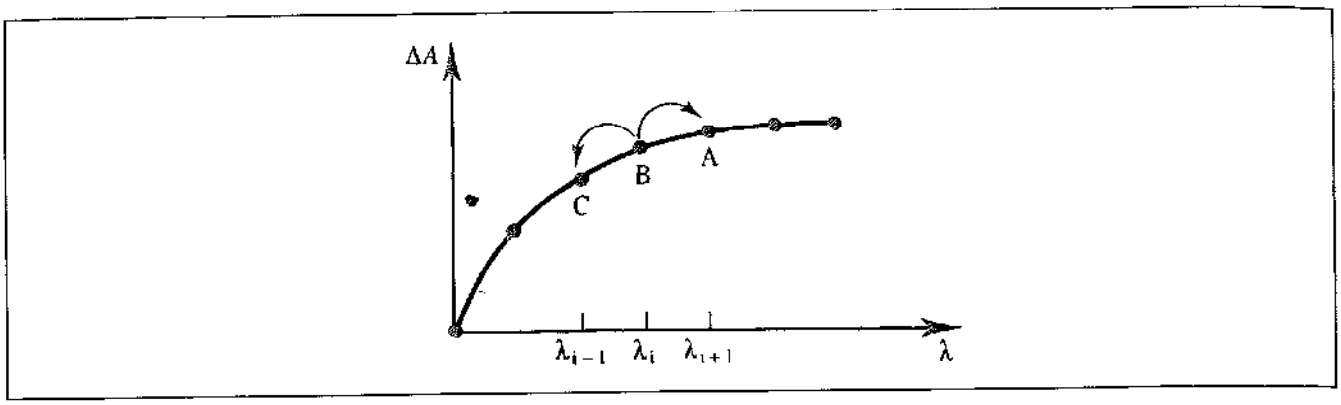


Fig 11 3 Double wide sampling enables two free energies to be accumulated from a single simulation.

illustrated in Figure 11.3. Consider point B in Figure 11.3, which corresponds to a coupling parameter λ_i. A simulation performed using λ_i can be used to obtain values for both the free energy difference $\Delta A(\lambda_i \rightarrow \lambda_{i+1})$ and the free energy difference $\Delta A(\lambda_i \rightarrow \lambda_{i-1})$. This is clearly a more efficient way to obtain the desired free energy as twice as many free energy differences can be obtained from a single simulation.

11.2.3 Thermodynamic Integration

An alternative way to calculate the free energy difference uses thermodynamic integration. The formula for the free energy difference is derived in Appendix 11.1 and is:

$$\Delta A = \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \right\rangle_\lambda d\lambda \tag{11 18}$$

To calculate a free energy difference using thermodynamic integration we thus need to determine the integral in Equation (11.18). In practice, this is achieved by performing a series of simulations corresponding to discrete values of λ between 0 and 1. For each value of λ, the average of

$$\left\langle \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \right\rangle_\lambda \tag{11 19}$$

is determined. These partial derivatives are calculated analytically in some programs but in others a finite difference approximation is used ($\partial \mathscr{H}/\partial \lambda \approx \Delta \mathscr{H}/\Delta \lambda$). The total free energy difference ΔA is then equal to the area under the graph of

$$\left\langle \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \right\rangle_\lambda \tag{11.20}$$

versus λ, as illustrated in Figure 11.4.

11.2.4 The 'Slow Growth' Method

A third approach for the calculation of free energy differences from computer simulation is the slow growth method. Here, the Hamiltonian changes by a very small, constant amount at

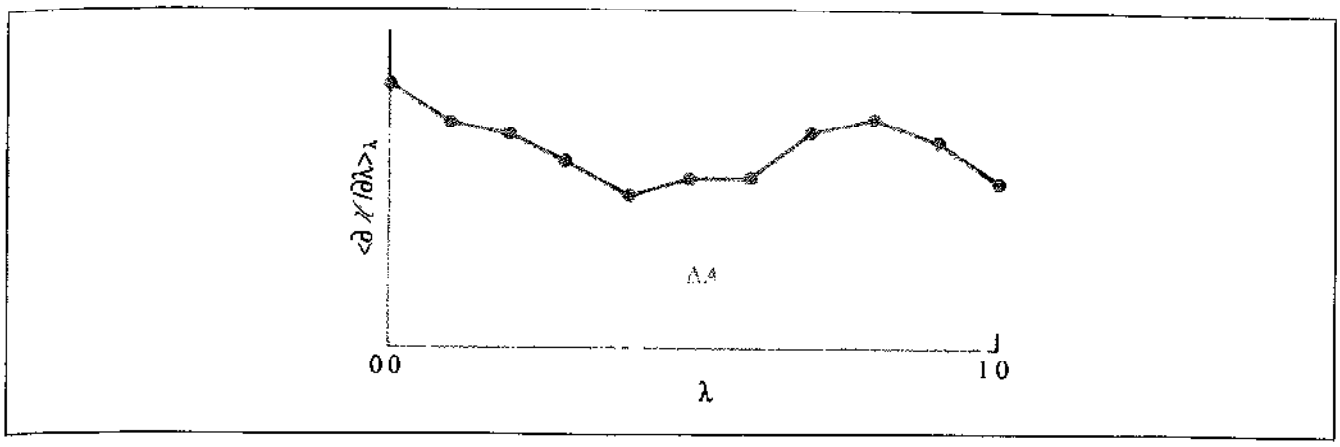


Fig 11 4: Calculation of free energy differences by thermodynamic integration

each step of the calculation. This means that at each stage the Hamiltonian $\mathscr{H}(\lambda_{i+1})$ is very nearly equal to $\mathscr{H}(\lambda_i)$. The free energy difference is given by:

$$\Delta A = \sum_{i=1,\lambda=0}^{i=N_{\text{step}};\lambda=1} (\mathscr{H}_{i+1} - \mathscr{H}_i) \tag{11.21}$$

This expression is derived in Appendix 11.2.

In principle, all three methods for calculating the free energy difference should give the same result, as the free energy is a state function and so independent of path. However, there may be practical reasons for choosing one method or another, as we shall discuss in Section 11.6. The other point to note at this stage is that our formulation of the free energy has been in terms of the partition function Q and the Hamiltonian $\mathscr{H}(\mathbf{p}^N,\mathbf{r}^N)$, which have contributions from both kinetic and potential energies. When the kinetic energy contributions are integrated out they cancel and so the various equations can be written in terms of the difference between the potential function, $\mathscr{V}(\mathbf{r}^N)$, rather than the Hamiltonian, $\mathscr{H}(\mathbf{p}^N,\mathbf{r}^N)$. Q is then replaced by the configurational integral, Z, and the free energy values that are obtained are excess free energies, relative to an ideal gas.

Our discussion so far has considered the calculation of Helmholtz free energies, which are obtained by performing simulations at constant *NVT*. For proper comparison with experimental values we usually require the Gibbs free energy, G. Gibbs free energies are obtained from a simulation at constant *NPT*.

11.3 Applications of Methods for Calculating Free Energy Differences

11.3.1 Thermodynamic Cycles

An early application of the free energy perturbation method was the determination of the free energy required to create a cavity in a solvent. Postma, Berendsen and Haak determined the free energy to create a cavity ($\lambda = 1$) in pure water ($\lambda = 0$) using isothermal–isobaric

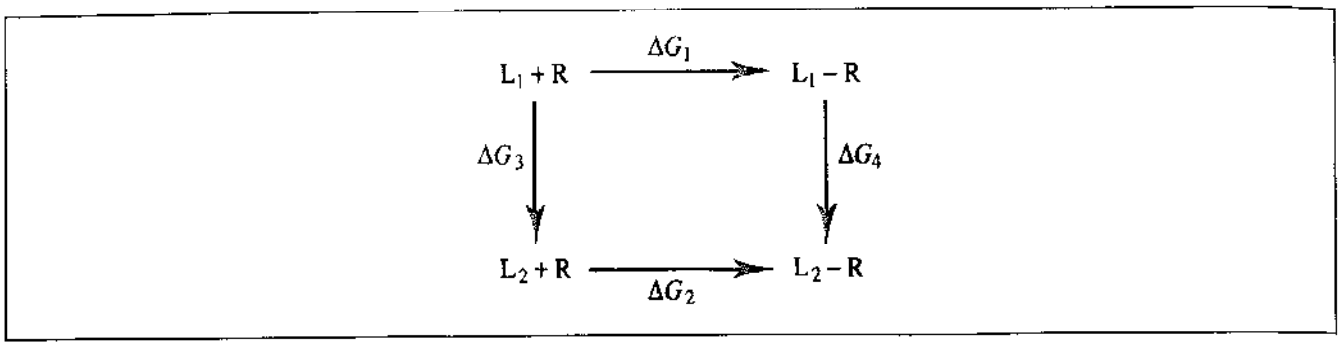


Fig. 11.5 Thermodynamic cycle for binding ligands L_1 and L_2 to receptor R

molecular dynamics simulations [Postma *et al.* 1982]. Five different cavity sizes were considered and the results showed that, as expected, the free energy of cavity formation increased with the size of the cavity, the results being in good agreement with analytical theories. For small cavities (< 1 Å radius) the results were inaccurate due to poor sampling. The calculations provided not only the free energy of cavity formation for the different cavity sizes but also structural and dynamic properties of the water molecules around the cavity. For example, the water structure varied with the cavity size. A cavity of radius 1.78 Å had the most pronounced shell structure, with a high first-neighbour peak and a significant second-neighbour peak in the cavity–water pair distribution function.

Many processes of interest to the molecular modeller involve an equilibrium between molecules that interact via non-covalent forces, the free energy being related to the equilibrium constant by $\Delta G = -RT \ln K$. Let us consider the binding of two different ligands (L_1 and L_2) to a receptor molecule (R). L_1 and L_2 could be putative inhibitors of an enzyme R or two 'guests' for a host R. The thermodynamic cycle for the two binding processes is shown in Figure 11.5. The relative binding affinity of L_1 and L_2 equals $\Delta G_2 - \Delta G_1$ and is commonly written $\Delta\Delta G$. In principle, it would be possible to calculate values of ΔG_1 and ΔG_2 by simulating the actual association process. To do this we would bring the ligand and the receptor together from an initial large separation to gradually form the intermolecular complex. However, in most cases this would involve such a major reorganisation of the receptor, the ligand and the solvent that it would be difficult to ensure adequate sampling of phase space.

The free energy is a state function, and so its value round a thermodynamic cycle must be zero. Thus $\Delta G_2 - \Delta G_1 = \Delta G_4 - \Delta G_3$ (Figure 11.5). ΔG_3 corresponds to the free energy difference of the two ligands in solution; ΔG_4 is the free energy difference of the two intermolecular complexes. The changes ΔG_3 and ΔG_4 do not correspond to any transformation that can be performed in the laboratory, but they are quite feasible in the computer. The free energy difference only depends upon the endpoints, and so we are at liberty to change the Hamiltonians in any way we wish. The free energy differences obtained from such non-physical pathways are likely to be much more reliable than the 'physically plausible' processes as they should involve much less reorganisation of the system This is particularly so if the two ligands L_1 and L_2 have similar structures. To calculate the relative free energy of binding of the two ligands we would therefore 'mutate' L_1 into L_2 in solution and L_1 to L_2 within the receptor. This is the *thermodynamic cycle perturbation approach* to calculating relative free energies.

11.3.2 Applications of the Thermodynamic Cycle Perturbation Method

One of the first applications of the thermodynamic cycle perturbation approach to the calculation of relative binding constants was the study by Lybrand, McCammon and Wipff of the synthetic macrocycle SC24, which, when protonated, can bind halide ions (Figure 11.6) [Lybrand *et al.* 1986]. SC24 binds Cl^- 4.30 kcal/mol more strongly than Br^-. Two simulations were performed to determine a theoretical value for this relative free energy using the free energy perturbation method with molecular dynamics. First, Cl^- was mutated to Br^- in aqueous solution, giving a free energy difference of 3.35 kcal/mol. The same mutation was then performed within the macrocycle, in a periodic box of water. The value obtained for this step was 7.50 kcal/mol, giving an overall relative free energy of binding of 4.15 kcal/mol. The experimental value was approximately 4.3 kcal/mol. Thus, although the free energy to desolvate Cl^- is unfavourable compared with Br^-, this is more than compensated for by favourable interactions between Cl^- and the host; Br^- is slightly too large to fit comfortably in the relatively inflexible SC24 molecule.

One of the most attractive applications of the free energy techniques is for predicting the relative free energies of binding of inhibitors of biological macromolecules such as proteins or DNA. If we know the binding constant of an inhibitor then we can, in principle at least, calculate the binding constant of a related inhibitor. The free energy cycle used to perform this calculation is analogous to that shown in Figure 11.5: we perform two separate free energy calculations: ligand L_1 is mutated to ligand L_2 in solution and within the binding site. An early calculation of this type was performed by Bash and colleagues, who studied two inhibitors of thermolysin (an enzyme which cleaves the amide bonds in peptides and proteins) [Bash *et al.* 1987]. The two inhibitors investigated had the general formula carbobenzoxy-Gly^P(X)-L-Leu-L-Leu [Barlett and Marlowe 1987] (Figure 11.7). The experimentally determined binding constants (K_i) of the X ≡ NH and X ≡ O inhibitors were 9.1 nM and 9000 nM, i.e. the former binds 1000 times more strongly. This difference in binding constants is equivalent to 4.1 kcal/mol. X-ray crystallographic analysis showed that the two inhibitors bind in almost identical positions. The calculated free energy difference was determined to

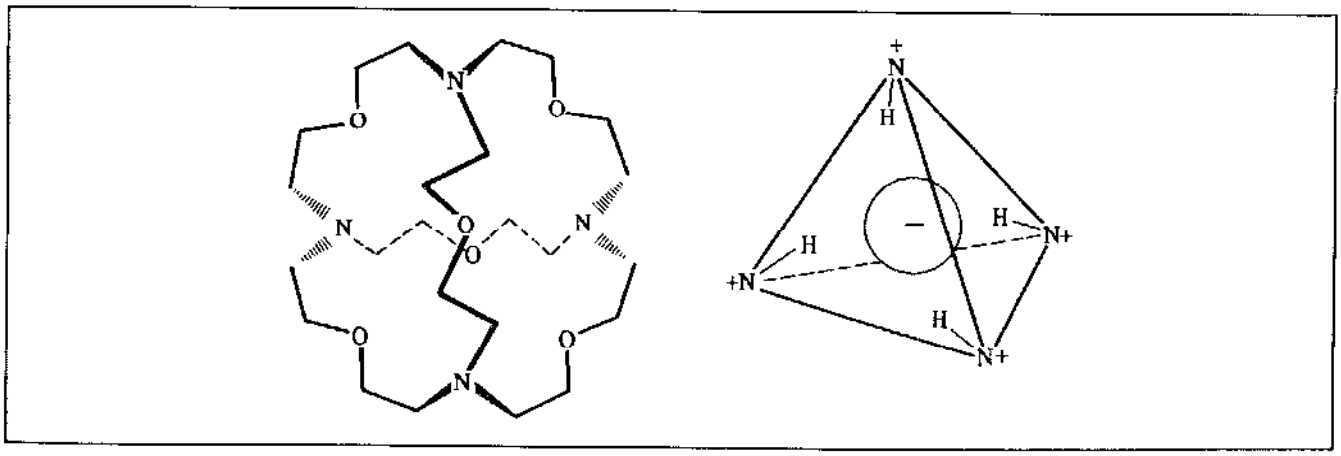


Fig. 11.6: The SC24/halide system. (Figure adapted from Lybrand T.P., J.A. McCammon and G. Wipff 1986 Theoretical Calculation of Relative Binding Affinity in Host–Guest Systems. Proceedings of the National Academy of Sciences USA *83 833–835.)*

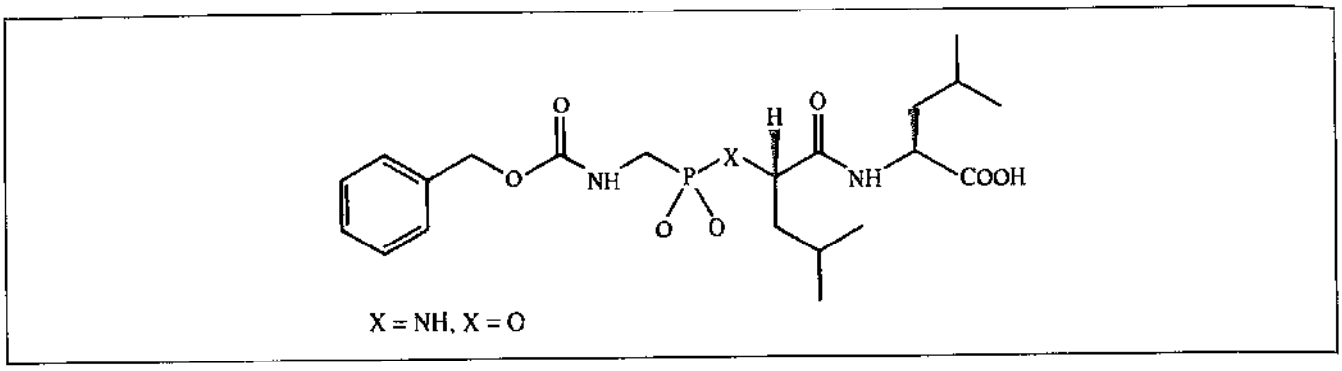


Fig. 11.7: Thermolysin inhibitors [Bartlett and Marlowe 1987]

be 4.2 ± 0.5 kcal/mol, in good agreement with the experimental result. In the active site of the enzyme the X group of the inhibitor interacts with the backbone carbonyl oxygen of one of the amino acids (Ala 113). The ester oxygen interacts unfavourably with this carbonyl, but the amide can form a hydrogen bond. The relative free energy of binding of the amide inhibitor to the protein was calculated to be 7.6 kcal/mol lower than the ester, but this was counteracted by the difference in the free energies of solvation, which was calculated to be 3.4 kcal/mol. The amide inhibitor thus incurs a greater desolvation penalty than the ester.

This study obviously gave very satisfactory agreement with the experimental data. However, a subsequent calculation by Merz and Kollman showed that the results were very sensitive to the charge model used for the inhibitor [Merz and Kollman 1989]. The charges for the inhibitor were obtained by electrostatic potential fitting in each case, though with different basis sets. This second calculation gave a free energy difference of 5.9 kcal/mol. Other studies have also shown that calculated free energies can be very sensitive to the charge model used; we will discuss some of the problems with performing free energy calculations in Section 11.6.

As one final example of the application of free energy calculations we will examine the determination of relative partition coefficients. The partition coefficient (P) is the equilibrium constant for the transfer of a solute between two solvents. The logarithm of the partition coefficient ($\log P$) for transfer between water and a variety of solvents (primarily 1-octanol) is widely used to derive structure–activity relationships (see Section 12.9), in which the biological activity of a molecule is correlated with its physicochemical properties. The thermodynamic cycle for the partition of two solutes, A and B, between two solvents is shown in Figure 11.8. If it were possible to calculate the free energy of transfer from one solvent to another (i.e. ΔG_1 or ΔG_2 in Figure 11.8) then this would give the partition coefficient directly. However, such a simulation would require an inordinate amount of time and probably be very inaccurate. A relative partition coefficient can be determined by mutating one solute into the other in the two separate solvents.

Calculations of relative partition coefficients have been reported using the free energy perturbation method with the molecular dynamics and Monte Carlo simulation methods. For example, Essex, Reynolds and Richards calculated the difference in partition coefficients of methanol and ethanol partitioned between water and carbon tetrachloride with molecular dynamics sampling [Essex *et al.* 1989]. The results agreed remarkably well with experiment

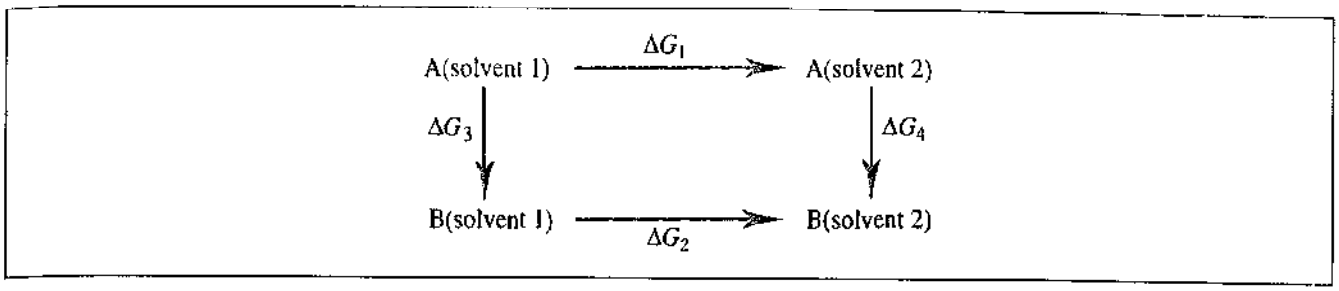


Fig 11.8: Thermodynamic cycle for calculating relative partition coefficients

(within 0.06 kcal/mol). Jorgensen, Briggs and Contreras used Monte Carlo methods to calculate the relative partition coefficients for eight pairs of solutes (including methanol/methylamine, acetic acid/acetamide and pyrazine/pyridine) between water and chloroform [Jorgensen *et al.* 1990]. For these eight systems good qualitative agreement with experimental data was obtained. However, the results involving acetic acid gave too broad a spread of values. This was traced to the relative free energies of hydration, which varied over too wide a range and indicated some areas for improvement in the force field model.

11.3.3 The Calculation of Absolute Free Energies

In some cases, it is possible to devise thermodynamic cycles which enable the absolute free energy of a change to be determined using free energy perturbation methods [Jorgensen *et al.* 1988]. Figure 11.9 shows a thermodynamic cycle for the association of L and R to give a complex LR in both the gas phase and in solution. ΔG_{ass} is the free energy of association in solution and is given by:

$$\Delta G_{ass} = \Delta G_{gas}(L + R \rightarrow LR) + \Delta G_{sol}(LR) - \Delta G_{sol}(L) - \Delta G_{sol}(R) \tag{11.22}$$

$\Delta G_{sol}(X)$ is the solvation free energy of the species X (the free energy of transfer from the gas phase to solvent). The solvation free energy can be written in terms of perturbations where the species disappear to nothing in the gas phase and in solution, $\Delta G_{sol}(X) = \Delta G_{gas}(X \rightarrow 0) - \Delta G_{sol}(X \rightarrow 0)$. The free energy of association, ΔG_{ass}, can then be written:

$$\begin{aligned}\Delta G_{ass} = {} & \Delta G_{gas}(L + R \rightarrow LR) - \Delta G_{gas}(L \rightarrow 0) + \Delta G_{sol}(L \rightarrow 0) \\ & - \Delta G_{gas}(R \rightarrow 0) + \Delta G_{sol}(R \rightarrow 0) + \Delta G_{gas}(LR \rightarrow 0) \\ & - \Delta G_{sol}(LR \rightarrow 0)\end{aligned} \tag{11.23}$$

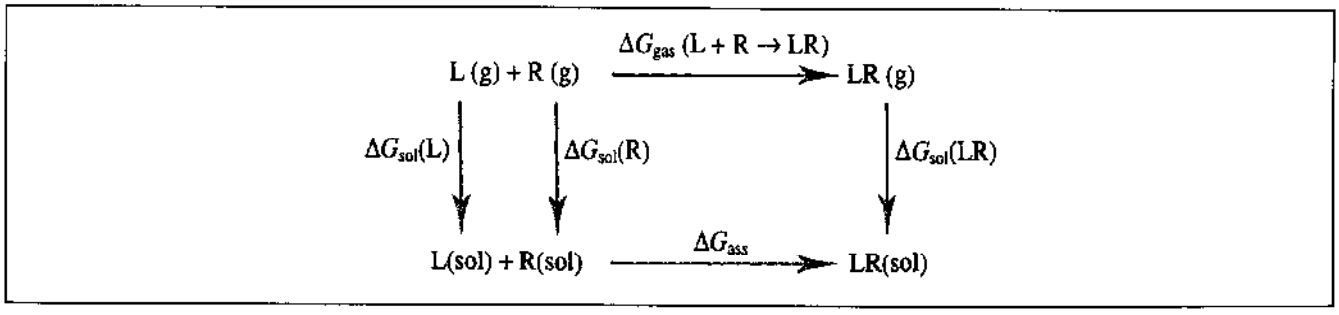


Fig 11 9 Thermodynamic cycle used to calculate absolute free energies [Jorgensen et al *1988]*

The gas-phase terms cancel and $\Delta G_{sol}(LR \rightarrow 0)$ can be written as the sum of two separate calculations:

$$\Delta G_{sol}(LR \rightarrow 0) = \Delta G_{sol}(LR \rightarrow R) + \Delta G_{sol}(R \rightarrow 0) \tag{11.24}$$

Thus, the overall free energy change can be written:

$$\Delta G_{ass} = \Delta G_{sol}(L \rightarrow 0) - \Delta G_{sol}(LR \rightarrow R) \tag{11.25}$$

We thus need perform only two simulations, L to nothing in water and L to nothing in the LR complex. The first application of this approach was to the association of two methane molecules in water, where both species (L and R) are identical. In general, L should be chosen as the smaller component.

11.4 The Calculation of Enthalpy and Entropy Differences

Free energy changes can now be routinely calculated with errors of less than 1 kcal/mol in favourable cases. How does this compare with the error with which the enthalpy or entropy difference can be determined? One way to determine the enthalpy change would be to perform two separate simulations, one of the initial system and one of the final system. For example, the difference in the enthalpy of solvation of ethanol and ethane thiol in water could be determined by simulating the two species separately and then taking the difference in the total enthalpies of the two systems. These total energies are invariably large numbers, with relatively large errors. The error in the calculated enthalpy difference would be comparable in magnitude to the error in the energy of each system. By contrast, the free energy is determined solely in terms of interactions involving the solute. This means that the free energy can be calculated much more accurately. More efficient ways to calculate the enthalpy and entropy change have been proposed for use with both free energy perturbation and thermodynamic integration schemes [Fleischman and Brooks 1987; Yu and Karplus 1988]. The uncertainties in the enthalpies and entropies so calculated are better than would be obtained by subtracting the differences in total energies, but they are still about one order of magnitude larger than the corresponding free energies.

11.5 Partitioning the Free Energy

The overall free energy can be partitioned into individual contributions if the thermodynamic integration method is used [Boresch *et al.* 1994; Boresch and Karplus 1995]. The starting point is the thermodynamic integration formula for the free energy:

$$\Delta A = \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \right\rangle_\lambda d\lambda \tag{11.26}$$

The Hamiltonian can be written as a sum of contributions from bond stretching, angle bending, and so on:

$$\left\langle \frac{\partial \mathscr{H}(\lambda)}{\partial \lambda} \right\rangle_\lambda = \left\langle \frac{\partial \mathscr{H}_{bonds}(\lambda)}{\partial \lambda} + \frac{\partial \mathscr{H}_{angles}(\lambda)}{\partial \lambda} + \cdots \right\rangle_\lambda \tag{11.27}$$

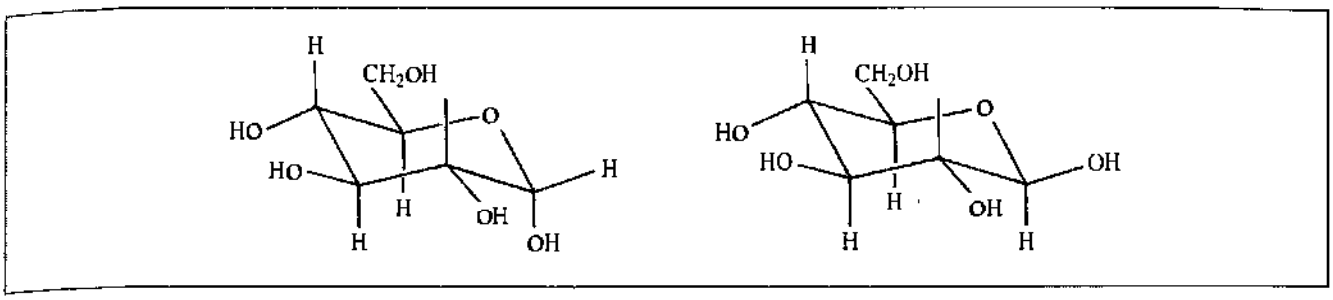


Fig 11.10 The α and β anomers of D-glucose

So the free energy is given by:

$$\Delta A = \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial \mathscr{H}_{\text{bonds}}(\lambda)}{\partial \lambda} \right\rangle_{\lambda} d\lambda + \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial \mathscr{H}_{\text{angles}}(\lambda)}{\partial \lambda} \right\rangle_{\lambda} d\lambda + \cdots$$
$$= \Delta A_{\text{bonds}} + \Delta A_{\text{angles}} + \cdots \tag{11.28}$$

We should remember that only the sum of the contributions is truly meaningful, as the individual contributions are not state functions. This has led some to criticise any use of such partitioning schemes [Smith and van Gunsteren 1994b], though they may be useful to indicate which interactions contribute the most to the overall free energy, and may also suggest the source of most of the error in the calculation. It is not possible to perform such a partitioning using thermodynamic perturbation.

An example of this partitioning scheme is the study by Ha and colleagues of the anomeric equilibrium between the α and β anomers of D-glucose [Ha *et al.* 1991]. D-glucose can exist in two tautomeric forms: α-D-glucose, in which the C_1 hydroxyl group is axial; and β-D-glucose, in which it is equatorial (Figure 11.10). In the gas phase, the axial α isomer is more stable than the equatorial β isomer, due to the anomeric effect which is considered to arise from unfavourable dipole–dipole interactions and delocalisation of the lone pair on the ring oxygen into an anti-bonding σ^* orbital. However, the β-D (equatorial) anomer is more stable than the α-D (axial) anomer by 0.3 kcal/mol in aqueous solution. The free energy difference between the two isomers in water was calculated by Ha *et al.* using both free energy perturbation and thermodynamic integration to be -0.3 ± 0.4 kcal/mol for $\beta \rightarrow \alpha$. A partitioning of the free energy showed that this small difference arose from the cancellation of two large terms: the α isomer was predicted to be 3.6 kcal/mol more favourable than the β isomer in the gas phase, due mainly to electrostatic effects. However, the β isomer was favoured over the α isomer in aqueous solution, again due to electrostatic effects, such as the enhanced hydrogen-bonding capability of the β isomer with the solvent. The small free energy difference, the difficulties of obtaining a reliable force field model and the large number of accessible conformations makes this equilibrium particularly difficult to tackle. One way in which the sampling problem has been tackled is by the use of a method called locally enhanced sampling (LES), which uses multiple copies of those parts of the system that can exist in more than one conformation. In the case of glucose these are the hydroxyl hydrogens and the hydroxymethyl group. In LES, each of the copies does not interact with the other copies of the same group and each atom 'sees' the mean force from all the copies. The first application of the technique was the study of the diffusion of

carbon monoxide within the protein myoglobin [Elber and Karplus 1990], but there are many other potential applications. LES reduces the barriers to conformational transitions, which leads to more rapid transitions between the conformational minima [Simmerling and Elber 1995]. However, the free energies calculated for the LES energy surface do need to be corrected to give the corresponding result for the single-copy system. In the case of glucose, it was found that the α isomer was favoured by 0.5–1.0 kcal/mol in the gas phase but that the β isomer was favoured in solution by 0.2 kcal/mol [Simmerling *et al.* 1998]. The gas-phase result was suggested to be a consequence of the tendency of the O–C–O–C linkage to adopt a gauche conformation, whereas the solution result was due to solvation effects. Many quantum mechanical studies have also been performed on this system, some of which have also included solvation effects (using the methods to be discussed in Section 11.10.2). For example, Barrows and co-workers performed high-level *ab initio* calculations on 11 varied low-energy conformations on glucose using large basis sets and including the effects of electron correlation [Barrows *et al.* 1998]. From this data, they calculated a gas-phase equilibrium constant of 0.4 kcal/mol in favour of the α isomer (using a Boltzmann-weighted average), whereas the equivalent value in solution was 0.6 kcal/mol in favour of the β isomer.

Another common practice is to partition the free energy into contributions from van der Waals and electrostatic interactions. This can be achieved rather easily by first perturbing the electrostatic and then the van der Waals parameters One system that has been studied in this way is the biotin/streptavidin complex. This protein–ligand complex is of particular interest due to the extremely strong association constant (−18 3 kcal/mol). The chemical structure of biotin is shown in Figure 11 11. The separate electrostatic and van der Waals free-energy calculations suggested that the largest contribution to the very negative free energy of binding was due to the non-polar van der Waals contribution rather than the electrostatic component [Miyamoto and Kollman 1993a, b]. Despite the presence of many hydrogen bonds between the ligand and the protein in the complex it was suggested that, whilst there was a large and favourable electrostatic interaction between biotin and streptavidin, this was almost cancelled by the free energy of interaction of biotin with water. By contrast, the van der Waals interaction gave a much greater contribution in the protein–ligand complex than for the ligand in water, so leading to its dominance. Indeed, the ligand is almost completely buried within the protein cavity, as can be seen in the structure of the intermolecular complex shown in Figure 11.12 (colour plate section).

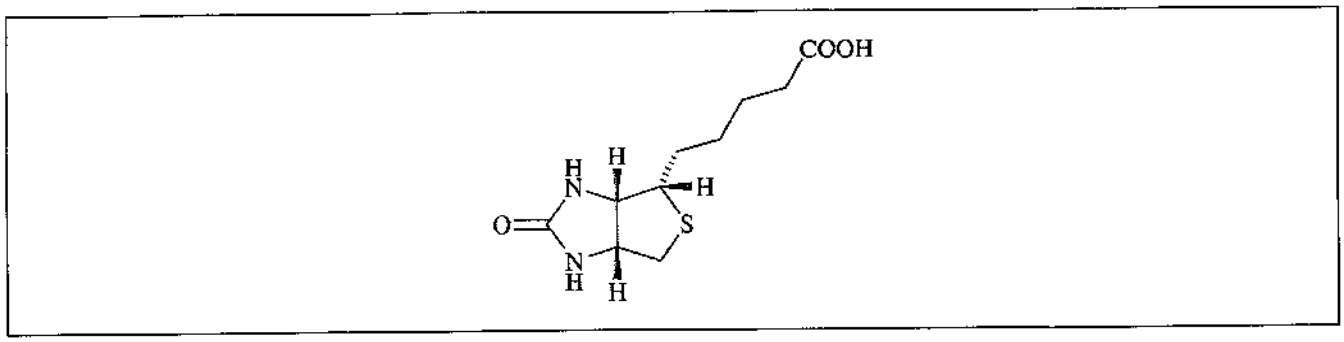


Fig 11 11· Biotin.

11.6 Potential Pitfalls with Free Energy Calculations

There are two major sources of error associated with the calculation of free energies from computer simulations. Errors may arise from inaccuracies in the Hamiltonian, be it the potential model chosen or its implementation (the treatment of long-range forces, etc.) The second source of error arises from an insufficient sampling of phase space.

Unfortunately, there is no set recipe that guarantees adequate coverage of phase space and thus reliable free energy values [Mitchell and McCammon 1991]. The errors associated with inadequate sampling may be identified by running the simulation for longer periods of time (molecular dynamics) or for more iterations (Monte Carlo); the perturbation can be performed in both forward and reverse directions; a different scheme could be used to determine the free energy difference (e.g. thermodynamic perturbation and thermodynamic integration). At the very least, the simulation should be run in both directions; the difference in the calculated free energy values (often referred to as the *hysteresis*) gives a lower-bound estimate of the error in the calculation

One possible pitfall to be aware of when estimating errors is that an excessively short simulation may give an almost zero difference between the forward and reverse directions. If the time of the simulation is much longer than the relaxation time of the system then the change can be performed reversibly. If the simulation time is of the same order of magnitude as the relaxation time then one would expect a significant degree of hysteresis However, if the simulation is much shorter than the relaxation time then approximately zero hysteresis may result, due to the inability of the system to adjust to the changes. In such a situation, the free energies for both forward and reverse directions may be approximately the same, but quite likely incorrect.

11.6.1 Implementation Aspects

The allure of methods for calculating free energies and their associated thermodynamic values such as equilibrium constants has resulted in considerable interest in free energy calculations. A number of decisions must be made about the way that the calculation is performed. One obvious choice concerns the simulation method. In principle, either Monte Carlo or molecular dynamics can be used; in practice, molecular dynamics is almost always used for systems where there is a significant degree of conformational flexibility, whereas Monte Carlo can give very good results for small molecules which are either rigid or have limited conformational freedom.

One must choose from the thermodynamic perturbation, thermodynamic integration and slow growth methods. Each of these methods has been extensively used, but the slow growth method is not now recommended. This method suffers from a phenomenon known as 'Hamiltonian lag'; the system never has time to properly equilibrate for a given value of the coupling parameter, because the potential function changes at every step. An additional advantage of the integration and perturbation approaches is that, should one decide at the end of a simulation that more sampling needs to be done for particular values of λ, or that more λ values are required over a particular range, then this can

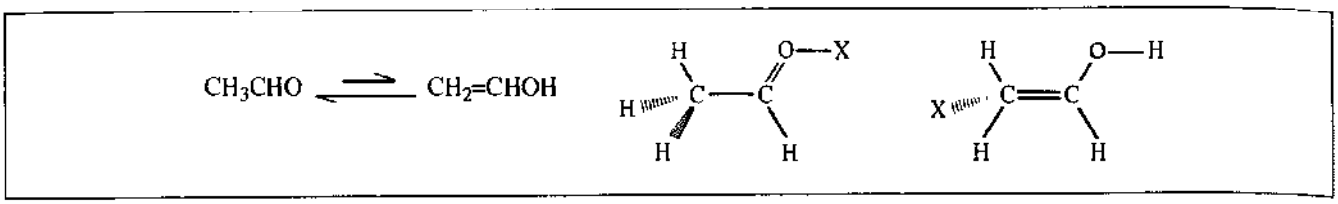


Fig. 11.13 To calculate the free energy difference between the aldehyde and enol forms of acetaldehyde, the single topology method uses dummy atoms (X)

easily be done without losing information from other parts of the calculation. With slow growth, one would have to redo the simulation from scratch.

Prior to the calculation, the increment $\delta\lambda$ in the coupling parameter must be specified. Traditionally, $\delta\lambda$ is set to a constant value before the simulation commences. It is important that there is enough overlap between successive states λ_i and λ_{i+1} so that reliable values can be obtained. An alternative approach is to use small changes in λ when the free energy is changing quickly and a larger change in λ when the free energy is changing more slowly. This is the basis of a method called *dynamically modified windows*, in which the slope of the free energy versus λ curve is used to determine the value of $\delta\lambda$ to use in the next iteration [Pearlman and Kollman 1989].

As the free energy is a thermodynamic state function the free energy difference between the initial and final states should be independent of the path along which the change is made, so long as it is reversible. It may be possible to proceed from the initial to the final state along more than one pathway. A change that involves high energy barriers will require much smaller increments to be made in the coupling parameter λ to ensure reversibility than a pathway that proceeds via a lower barrier.

Many free energy calculations involve changes in the molecular topologies of the species concerned, there are often different numbers of atoms in the initial and final states, and the atoms may be bonded in different ways. For example, suppose we wish to determine the free energy difference between acetaldehyde and its enol (Figure 11.13), in which a hydrogen atom migrates from the methyl carbon atom to the carbonyl oxygen. The system can be represented in the calculation using either a 'single' topology or a 'dual' topology. In the single-topology method, the molecular topology at all stages is the union of the initial and final states, using dummy atoms where necessary. A dummy atom does not interact with the other atoms in the system. Thus the hydrogen atom bonded to the oxygen in the enol form would be represented as a dummy atom when the simulation reached the endpoint corresponding to the aldehyde as shown in Figure 11.13.

The alternative to the single-topology representation is the dual-topology method. Here, both the molecular topologies are maintained during the entire simulation, such that both species 'exist' (in a topological sense) but do not interact with each other. The Hamiltonian that describes the interaction between these groups and the environment can be described in a number of ways, the simplest of which is the linear relationship:

$$\mathscr{H}(\lambda) = \lambda\mathscr{H}_Y + (1-\lambda)\mathscr{H}_X \tag{11.29}$$

Many free energy calculations involve the creation or annihilation of atoms. A potential problem with such simulations is that a singularity may occur in the function for which

an ensemble average is to be formed. One way to try to deal with this is to scale the initial Hamiltonian by a factor λ^n (rather than just λ) and the final Hamiltonian by $(1-\lambda)^n$ (rather than $(1-\lambda)$). It can be shown that for Monte Carlo simulations the singularity problem can be dealt with, provided n is at least 4 [Buetler *et al.* 1994]. However, a molecular dynamics simulation requires not only the energies to be calculated but also the first and the second derivatives. If λ^n scaling is used then either a steadily decreasing time step must be used as λ approaches zero, or these regions of the simulation must be omitted altogether and their contributions estimated by extrapolation. An alternative approach is to replace the traditional Lennard-Jones interaction with a soft-core potential of the following form [Buetler *et al.* 1994; Liu *et al.* 1996]:

$$v_{ij}^{\mathrm{LJ}} = 4\varepsilon_{ij}\left(\frac{\sigma_{ij}^{12}}{[\alpha_{\mathrm{LJ}}\sigma_{ij}^{6}+r_{ij}^{6}]^2} - \frac{\sigma_{ij}^{6}}{(\alpha_{\mathrm{LJ}}\sigma_{ij}^{6}+r_{ij}^{6})}\right) \tag{11.30}$$

where ε_{ij} and σ_{ij} have their usual Lennard-Jones meanings. The parameter α_{LJ} determines the 'softness' of the interaction, which has the effect of making the interaction approach a finite value as the interatomic distance r_{ij} goes to zero. With a suitable choice of the parameter α_{LJ} it is possible to ensure that the position of the minimum in the soft-core potential coincides with that of the unscaled energy curve (Figure 11.14) When used to perturb the system from X at $\lambda = 0$ to Y at $\lambda = 1$ then the soft-core Lennard-Jones interaction between the particle i that is being perturbed and some other particle j at a distance r_{ij} varies as:

$$\begin{aligned} v_{ij}^{\mathrm{LJ}}(\lambda) = {} & 4(1-\lambda)\varepsilon_{\mathrm{X}}\left(\frac{\sigma_{\mathrm{X}}^{12}}{[\alpha_{\mathrm{LJ}}\lambda^2\sigma_{\mathrm{X}}^{6}+r_{ij}^{6}]^2} - \frac{\sigma_{\mathrm{X}}^{6}}{[\alpha_{\mathrm{LJ}}\lambda^2\sigma_{\mathrm{X}}^{6}+r_{ij}^{6}]}\right) \\ & + 4\lambda\varepsilon_{\mathrm{Y}}\left(\frac{\sigma_{\mathrm{Y}}^{12}}{[\alpha_{\mathrm{LJ}}(1-\lambda)^2\sigma_{\mathrm{Y}}^{6}+r_{ij}^{6}]^2} - \frac{\sigma_{\mathrm{Y}}^{6}}{[\alpha_{\mathrm{LJ}}(1-\lambda)^2\sigma_{\mathrm{Y}}^{6}+r_{ij}^{6}]}\right) \end{aligned} \tag{11.31}$$

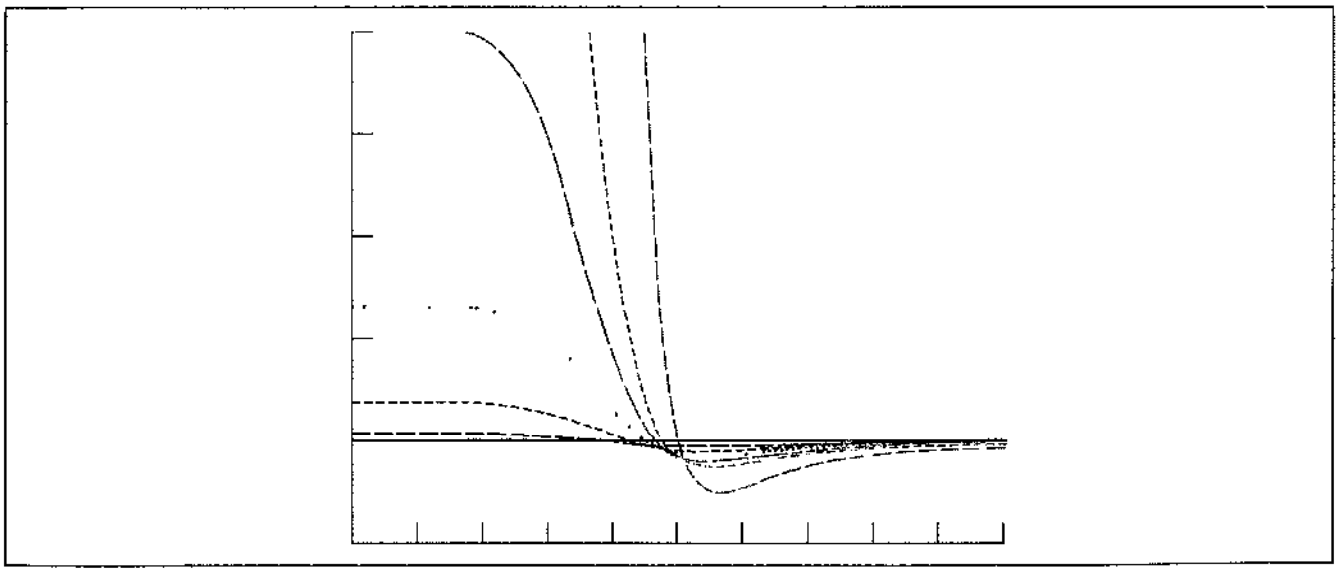


Fig 11 14. Comparison of scaled and unscaled Lennard-Jones potentials (Equation (11 31)) for the case where a particle disappears at $\lambda = 0$ As λ decreases the curves get progressively closer to the x axis

A similar soft-core expression can also be derived for the electrostatic interactions. In the situation where the particle disappears then only one of the terms remains (i.e. the second term if the particle disappears at $\lambda = 0$). The effect of using this type of soft-core potential can be seen in simulations of protein–ligand systems, where the ligand 'disappears' and is replaced by one or more solvent molecules. In a normal perturbation calculation, decreasing the effective radius of the ligand atoms will give rise to a collapse of the protein cavity and disruption to the surrounding protein structure. By contrast, placing the soft-core interaction sites at locations where the atoms are to be created or deleted maintains the protein cavity, because the solvent molecules are able to actually pass through the ligand as it is annihilated. This feature of soft-core potentials can also be useful for other kinds of free energy calculation where it is desired to try to simultaneously derive the relative free energies of binding of several ligands to a receptor and also in certain types of simulated annealing structure refinement.

11.7 Potentials of Mean Force

The free energy changes that we have considered so far correspond to chemical 'mutations'. We may also be interested to know how the free energy changes as a function of some inter- or intramolecular coordinate, such as the distance between two atoms, or the torsion angle of a bond within a molecule. The free energy surface along the chosen coordinate is known as a *potential of mean force* (PMF). When the system is in a solvent, the potential of mean force incorporates solvent effects as well as the intrinsic interaction between the two particles. Potentials of mean force were introduced in our discussion of Langevin dynamics (Section 7.8), where we noted that the ratio of *trans* to *gauche* conformers of 1,2-dichloroethane was significantly different in the liquid than in an isolated molecule. Unlike the mutations so common in free energy perturbation calculations, which are often along non-physical pathways, the potential of mean force is calculated for a physically achievable process. Consequently, the point of highest energy on the free energy profile that is obtained from a PMF calculation corresponds to the transition state for the process, from which it is possible to derive kinetic quantities such as rate constants.

Various methods have been proposed for calculating potentials of mean force. The simplest type of PMF is the free energy change as the separation (r) between two particles is changed. We might anticipate that we could calculate the potential of mean force from the radial distribution function using the following expression for the Helmholtz free energy:

$$A(r) = -k_B T \ln g(r) + \text{constant} \tag{11.32}$$

The constant is often chosen so that the most probable distribution corresponds to a free energy of zero.

Unfortunately, the potential of mean force may vary by several multiples of $k_B T$ over the relevant range of the parameter r. The logarithmic relationship between the potential of mean force and the radial distribution function means that a relatively small change in the free energy (i.e. a small multiple of $k_B T$) may correspond to $g(r)$ changing by an order of magnitude from its most likely value. Unfortunately, standard Monte Carlo or molecular

dynamics simulation methods do not adequately sample regions where the radial distribution function differs drastically from the most likely value, leading to inaccurate values for the potential of mean force. The traditional way to avoid this problem uses a technique called *umbrella sampling*.

11.7.1 Umbrella Sampling

Umbrella sampling attempts to overcome the sampling problem by modifying the potential function so that the unfavourable states are sampled sufficiently. The method can be used with both Monte Carlo and molecular dynamics simulations. The modification of the potential function can be written as a perturbation:

$$\mathscr{V}'(\mathbf{r}^N) = \mathscr{V}(\mathbf{r}^N) + W(\mathbf{r}^N) \tag{11.33}$$

where $W(\mathbf{r}^N)$ is a weighting function, which often takes a quadratic form:

$$W(\mathbf{r}^N) = k_W(\mathbf{r}^N - \mathbf{r}_0^N)^2 \tag{11.34}$$

For configurations that are far from the equilibrium state $\mathbf{r}_0^N$ the weighting function will be large and so a simulation using the modified energy function $\mathscr{V}'(\mathbf{r}^N)$ will be biased along some relevant 'reaction coordinate' away from the configuration $\mathbf{r}_0^N$. The resulting distribution will, of course, be non-Boltzmann. The corresponding Boltzmann averages can be extracted from the non-Boltzmann distribution using a method introduced by Torrie and Valleau [Torrie and Valleau 1977]. The result is:

$$\langle A \rangle = \frac{\langle A(\mathbf{r}^N) \exp[+W(\mathbf{r}^N)/k_B T] \rangle_W}{\langle \exp[+W(\mathbf{r}^N)/k_B T] \rangle_W} \tag{11.35}$$

The subscript W indicates that the average is based on the probability $P_W(\mathbf{r}^N)$, which in turn is determined by the modified energy function $\mathscr{V}'(\mathbf{r}^N)$. For example, to obtain the potential of mean force via the radial distribution function (Equation (11.32)) the distribution function with the forcing potential would be determined and then corrected to give the 'true' radial distribution function, from which the free energy can be calculated as a function of the separation It is usual to perform an umbrella sampling calculation in a series of stages, each of which is characterised by a particular value of the coordinate and an appropriate value of the forcing potential $W(\mathbf{r}^N)$. However, if the forcing potential is too large, the denominator in Equation (11.35) is dominated by contributions from only a few configurations with especially large values of $\exp[W(\mathbf{r}^N)]$ and the averages take too long to converge.

To illustrate the use of umbrella sampling, let us consider how the technique has been used to determine the potential of mean force for rotation of the central C–C bond of butane in aqueous solution. The barrier between the *trans* and *gauche* conformations of butane is approximately 3.5 kcal/mol, which is sufficiently high to give sampling problems in simulations. For example, in the molecular dynamics simulation of Ryckaert and Bellemans the mean time between *gauche–trans* transitions was about 10 ps [Ryckaert and Bellemans 1978]. Jorgensen, Gao and Ravimohan used umbrella sampling with Monte Carlo simulations to calculate the potential of mean force as the central bond in butane is rotated in a periodic box of water molecules, to determine the effect of the solvent on the relative

populations of the different conformations [Jorgensen *et al.* 1985]. The results predicted a shift in the expected populations of *trans* and *gauche* isomers from 68% *trans* in the gas phase to 54% in aqueous solution, a change of 14%. In addition, the barrier height was reduced in solution. Jorgensen and colleagues performed many calculations on similar systems using umbrella sampling and Monte Carlo simulations; he recommended that to reduce the barriers to a value between 1 kcal/mol and 3 kcal/mol was appropriate. In some cases, it is possible to use a barrier height of zero, though the barriers cannot be reduced too severely as this makes the forcing potential too large.

It is also possible to calculate potentials of mean force using the free energy perturbation method with a molecular dynamics or Monte Carlo simulation. As usual, the calculation is broken into a series of steps that are characterised by a coupling parameter λ. With molecular dynamics, holonomic constraint methods are used to fix the desired coordinates without affecting the dynamic motion of the system. This is the essence of the extension of the SHAKE procedure by Tobias and Brooks to cope with general coordinate changes [Tobias and Brooks 1988] (see Section 7.5). In a Monte Carlo simulation the required coordinates are simply fixed at the desired value(s). This contrasts with umbrella sampling, in which the coordinate(s) of interest would be able to vary over their range of values throughout the simulation, subjected to a potential that has been modified using the forcing function. At each step of the perturbation calculation, the difference in the energy between the configuration and the configuration that corresponds to $\lambda + \delta\lambda$ is determined and the free energy accumulated in the appropriate way.

To compare the perturbation and umbrella sampling methods for calculating potentials of mean force, Jorgensen and Buckner repeated the PMF calculation for butane in water using the perturbation method [Jorgensen and Buckner 1987]. The *gauche* population was calculated to increase by 12.3% using this method, in accordance with the previous umbrella sampling calculations. Jorgensen put forward several arguments in favour of the perturbation approach. A major concern with umbrella sampling is that a proper sampling of the phase space may not be achieved. In some cases, the presence of bottlenecks in phase space may be identified if separate simulations starting from different configurations give different results, but even this approach is not fail-safe as all simulations may encounter the same problem. Indeed, Jorgensen suggested that just such a bottleneck may have occurred in a previous simulation of pentane in water using umbrella sampling (which involved 5 million Monte Carlo steps). The only real problem with the perturbation method is the need to choose an appropriate value of $\delta\lambda$ so that there is adequate overlap between the configuration corresponding to λ and that corresponding to $\lambda + \delta\lambda$. Jorgensen and Buckner varied the central torsion angle in their simulation of butane using 15° increments.

11.7.2 Calculating the Potential of Mean Force for Flexible Molecules

To calculate a potential of mean force using free energy perturbation (or indeed umbrella sampling) it is necessary to determine the pathway for the transition of interest. This is trivial for simple problems such as the separation of two particles or the rotation of butane but can be quite complicated for more detailed changes such as conformational interconversions.

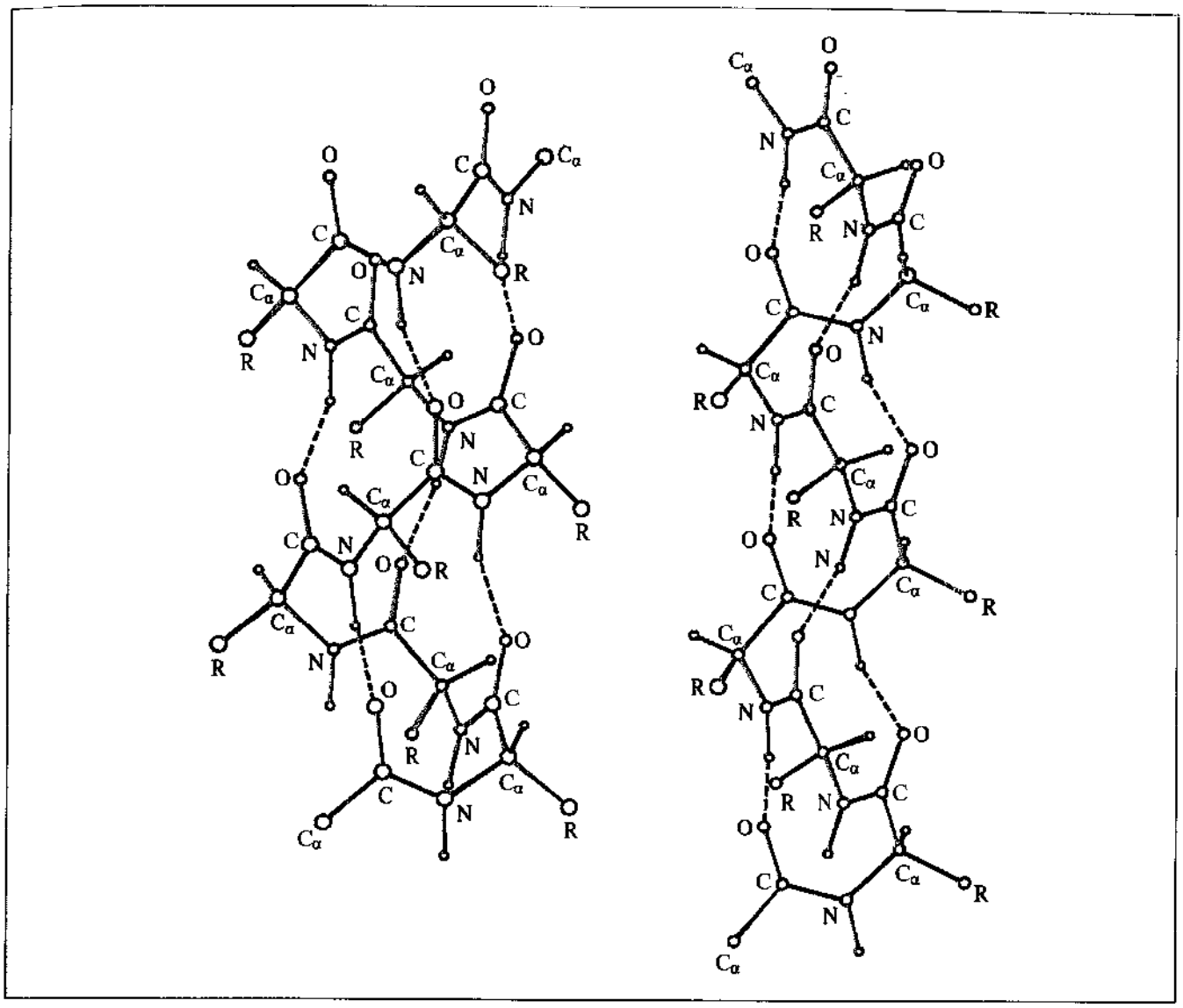


Fig 11 15. The α-helix (left) and the 3_{10}-helix

The reaction path methods discussed in Section 5.9.3 may be helpful in determining these pathways.

To illustrate the calculation of potentials of mean force for flexible systems we will consider helical conformations of polypeptide chains. We have already met the α-helix, which is commonly observed in protein structures (see Section 10.2). In this conformation, hydrogen bonds are formed between residues i and $i+4$. Polypeptide chains can also form a different type of helix, called a 3_{10}-helix. Here, the hydrogen bonds are formed between residues i and $i+3$. These two helices are compared in Figure 11.15. The backbone conformations of such helices do not differ significantly: the α-helix has backbone torsion angles ($\phi = -60°$, $\psi = -50°$) and the 3_{10}-helix has ($\phi = -50°$, $\psi = -28°$). The 3_{10}-helix is found to a small extent in protein structures, usually at the ends of α-helices. However, the 3_{10}-helix is much more common in peptides formed from α,α-dialkyl amino acids, which have two alkyl substituents at the α-carbon atom. The prototypical member of this class of amino acids is α-methylalanine (MeA; see Figure 11.16) Peptides containing this amino acid can

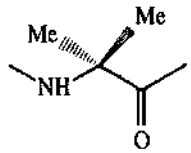


Fig. 11.16: Methylalanine

form both α-helices and 3_{10}-helices with the actual conformation present being rather sensitive to the conditions; such peptides are 3_{10}-helical in $CDCl_3$ and α-helical in $(CD_3)_2SO$.

To calculate the potential of mean force for interconverting the α-helix to the 3_{10}-helix requires an appropriate reaction coordinate to be determined. Here we describe the calculations of three groups who all used different approaches. Smythe, Huston and Marshall studied a decamer of α-methylalanine, Ch_3CO-MeA_{10}-NMe, using umbrella sampling [Smythe *et al.* 1993, 1995]. They used the self-penalty walk method described in Section 5.9.3 to determine the transition pathway and observed that the reaction coordinate correlated well with a smooth change in the end-to-end distance from the 3_{10}-helix (19 Å) to the α-helix (13 Å). Their umbrella sampling calculations were performed using molecular dynamics, with the end-to-end distance being subjected to a restraining potential. Simulations were performed in various solvents: in water, the free energy change for the α-helix $\rightarrow$ 3_{10}-helix transition was calculated to be 7.6 kcal/mol, with the value in dichloromethane being 5.8 kcal/mol and *in vacuo* 3.2 kcal/mol. Although a distinct energy barrier was found for the vacuum calculations, no transition barrier was found for either solution calculation.

Zhang and Hermans studied a 10-residue alanine peptide as well as a 10-residue α-methylalanine peptide, *in vacuo* and in water [Zhang and Hermans 1994]. In their calculations, the transition from one conformation to the other was performed using a restraining potential that forced the structure to exchange one set of hydrogen bonds to the set of hydrogen bonds appropriate to the other structure. This additional potential function could be used to drive the molecule back and forth between the two conformations by varying a coupling parameter, λ, between 0 and 1. Free energy profiles were determined for the α-helix to 3_{10}-helix transition using molecular dynamics and the slow growth method. The results showed that the alanine peptide had a clear preference for the α-helix both *in vacuo* and in water but that the free energy change for the MeA peptide was approximately zero in water and that the 3_{10}-helix was preferred *in vacuo*. It was proposed by Zhang and Hermans that the discrepancy between these results for the α-methylalanine peptide and those obtained by Smythe, Huston and Marshall was probably due to the different force field models employed; Smythe *et al.* used a united atom model, whereas Zhang and Hermans used an all-atom model.

Tirado-Reeves, Maxwell and Jorgensen used yet another approach for calculating the potential of mean force, this time for an undecaalanine peptide in water [Tirado-Reeves *et al.* 1993]. The free energy profile was calculated using the perturbation method and Monte Carlo simulations by gradually varying the ψ backbone torsion angles, keeping the ϕ torsion angles fixed at $-60°$. The free energy difference between the two conformations

was calculated to be 10.6 kcal/mol in favour of the α-helix, with a small activation barrier of 2.8 kcal/mol for the 3_{10}- to α-helical transition. *In vacuo*, a larger free energy difference was predicted (13.6 kcal/mol).

These three studies have been described at some length, in part to illustrate the different approaches available for calculating thermodynamic properties of complex systems but also to emphasise the fact that different methods can given quite different (and sometimes contradictory) results. Such comparative studies serve to highlight the fact that it is necessary to examine critically the methods and models used in a calculation. All three studies were in part prompted by experimental electron spin resonance results that suggested that a 16-residue alanine-based peptide adopted a 3_{10}-helical conformation in water [Miick *et al.* 1992]. These results were contradicted by all the simulations, and indeed prompted Smythe and Marshall to undertake similar experiments on their conformationally constrained peptides, experiments which showed that these peptides were α-helical, in agreement with the calculations.

11.8 Approximate/'Rapid' Free Energy Methods

Free energy calculations are notoriously time-consuming to perform. Whilst one might have anticipated that ever faster computers would have made significant inroads on this problem, in some respects the opposite has happened, as researchers are now able to more fully quantify the need for sufficient sampling of phase space and to attain better convergence. In addition, of course, there is a natural desire to investigate ever larger systems. A practical illustration of the dilemma facing the proponents of free energy methods as a predictive tool, at least in an industrial environment, is that, if the calculation takes longer to perform than a candidate molecule can be synthesised and tested, then there is little practical benefit from attempting the calculation. There has thus been continued interest in the development of alternative methods which, whilst still being based upon 'exact' statistical mechanics, are intended to provide answers with less computational effort than a full-blown free energy calculation. These methods tend to approach the problem from one of two perspectives. Some, such as the λ-dynamics method, enable a single simulation to provide information on a number of molecules. Others, such as the linear response method, aim to limit the amount of simulation that needs to be performed.

In a traditional free energy calculation a coupling parameter, λ, provides the link between the initial and final systems. In most free energy calculations λ varies uniformly from 0 to 1 (or from 1 to 0), one exception being the dynamically modified windows technique discussed in Section 11.6.1. By contrast, the λ-dynamics technique considers lambda to be another 'particle' in the simulation, with its own fictitious mass. As such, λ-dynamics is similar in some respects to those charge calculation schemes where the charges can vary as a dynamic variable (see Section 4.9.6). Specifically, lambda corresponds to the reaction coordinate along which the potential would be modified in an umbrella sampling calculation. The advantage of making this association is that the biasing potentials that are used in the umbrella sampling method can be used in the λ-dynamics technique to provide enhanced sampling in relevant regions of configurational space. Indeed, it is possible to

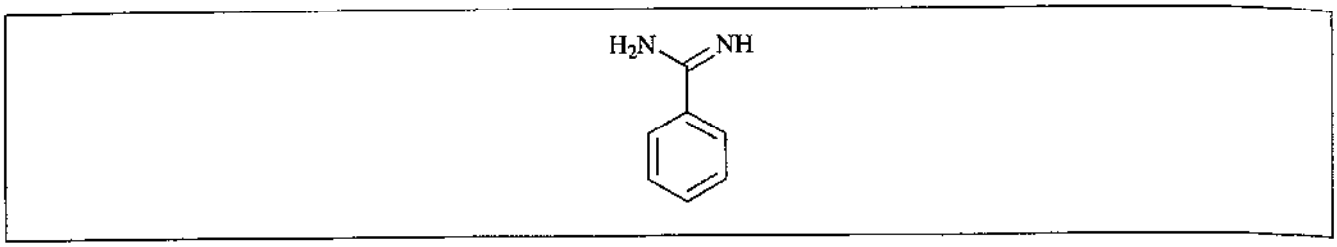


Fig 11.17 Benzamidine

use a set of coupling variables, λ_i, $i = 1, \ldots, n$. In the case where we have just one molecule being changed into another, then these different λ_i represent changes in different components of the interaction potential (for example, the Lennard-Jones and Coulombic interactions). If λ_1 and λ_2 refer to the Coulombic and the van der Waals interactions, respectively, then the potential function can be written:

$$\mathscr{V}(\mathbf{r}^N, \lambda_1, \lambda_2) = (1 - \lambda_1)\mathscr{V}_{\mathrm{A}}^{\mathrm{coul}} + \lambda_1 \mathscr{V}_{\mathrm{B}}^{\mathrm{coul}} + (1 - \lambda_2)\mathscr{V}_{\mathrm{A}}^{\mathrm{L-J}} + \lambda_2 \mathscr{V}_{\mathrm{B}}^{\mathrm{L-J}} + \mathscr{V}_{\mathrm{env}}(\mathbf{r}^N) \qquad (11.36)$$

Here, the term $\mathscr{V}_{\mathrm{env}}(\mathbf{r}^N)$ corresponds to all interactions involving those parts of the system that are not changing (i.e. the solvent and the unchanging part of the solute) The lambda variables move under the influence of a specific term which serves to limit their absolute extent (i.e. between 0 and 1) and which can be used to restrict their value to particular ranges during the simulation in order to provide enhanced sampling at particular points.

The basic λ-dynamics scheme can be used to perform a 'regular' type of free energy calculation in which one solute is perturbed into another such as the perturbation of methanol to ethane or to methane thiol [Kong and Brooks 1996]. However, it can also be used to investigate a number of perturbations simultaneously. As such, it provides a route to assess several free energies from a single simulation. One published example concerns the binding of benzamidine derivatives to the enzyme trypsin [Guo and Brooks 1998; Guo *et al.* 1998] Benzamidine is shown in Figure 11.17; this molecule binds relatively strongly to the enzyme because the positively charged amidine group interacts with a negatively charged aspartate residue in the protein. However, substitution at the para position can affect the strength of binding, with *p*-amino benzamidine binding slightly more strongly, *p*-methyl slightly more weakly and *p*-chloro more weakly still than the parent molecule. When λ-dynamics is applied to this problem, each of the L ligands ($L = 4$ in this case) is represented by a different value of λ_i. Initially, all values of λ_i are set to $1/L$ and their velocities set to zero. This means that each molecule is set on an equal footing at the beginning of the calculation. The system then evolves under the influence of the following hybrid potential:

$$\mathscr{V}(\mathbf{r}^N, \lambda_i) = \sum_{i=1}^{L} \lambda_i^2 (\mathscr{V}_i(\mathbf{r}^{\mathrm{int}}) - F_i) + \mathscr{V}_{\mathrm{env}}(\mathbf{r}^N) \qquad (11.37)$$

As in Equation (11.36), $\mathscr{V}_{\mathrm{env}}(\mathbf{r}^N)$ corresponds to those interactions concerning all atoms not directly involved in the perturbations, whereas $\mathscr{V}_i(\mathbf{r}^{\mathrm{int}})$ concerns those atoms associated with the group being perturbed in ligand i (for which the associated lambda parameter is λ_i). F_i is a reference free energy and can serve two purposes If F_i equals the solvation/desolvation free energy of the relevant ligand then the free energy value obtained from

the calculation corresponds to the free energy change for the full cycle. F_i can also be used as a biasing potential to control the sampling in particular regions of phase space. Finally, there is a constraint on the values of λ_i:

$$\sum_{i=1}^{L} \lambda_i^2 = 1 \tag{11.38}$$

As the simulation proceeds, the values of λ_i fluctuate, subject to the constraint in Equation (11.38). The free energy difference between two molecules i and j can be determined by identifying the probability that each molecule occupies the state $\lambda_i = 1$ or $\lambda_j = 1$, respectively Thus:

$$\Delta\Delta A_{ij} = -\frac{1}{k_B T} \ln \left[\frac{P(\lambda_i = 1, \lambda_{m \neq i} = 0)}{P(\lambda_j = 1, \lambda_{n \neq i} = 0)} \right] \tag{11.39}$$

These relative probabilities can be easily determined by simply counting the number of times during the simulation that the relevant value of lambda reaches unity. In the case of the para-substituted benzamidines it was possible after only a relatively short simulation (110 ps) to observe that the p-chloro and p-methyl derivatives were significantly weaker than the p-amino and the parent compound (Figure 11.18). In this particular case, all four

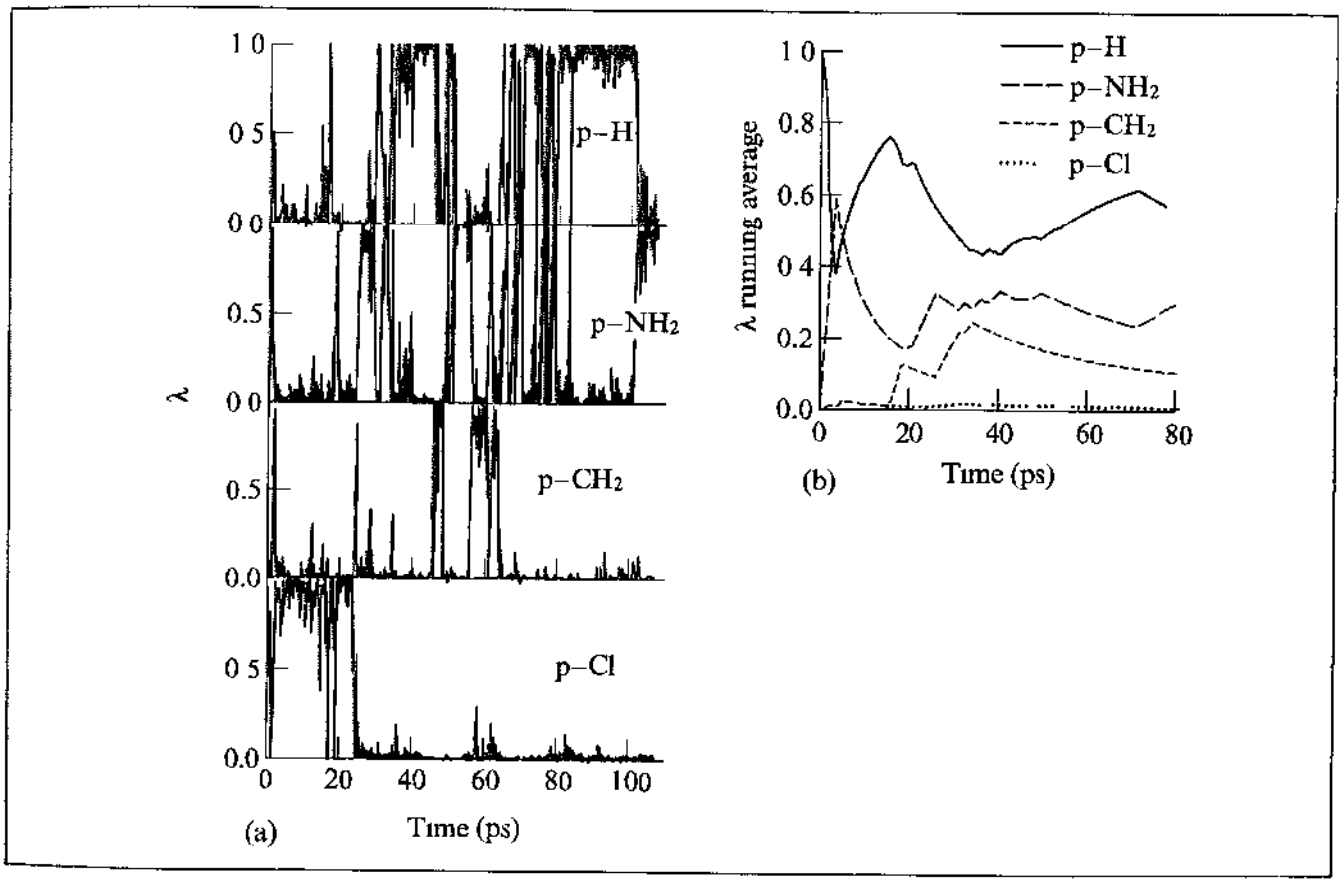


Fig 11 18 λ-dynamics simulation of benzamidine derivatives binding to trypsin. (a) The larger the value of λ the stronger the interaction with the protein at that instant (b) Running average of each value of λ over the course of the simulation (Figure redrawn from Guo Z. and C L Brooks III 1998. Rapid Screening of Binding Affinities Application of the λ-Dynamics Method to a Trypsin-Inhibitor System Journal of the American Chemical Society **120** 1920–1921)

inhibitors have rather similar binding affinities (within 1 kcal/mol), thus requiring a relatively long simulation to separate them. In general, a compound with a binding affinity more than 3 kcal/mol worse than the most favourable molecule should be screened out within a few tens of picoseconds, though longer simulation times would be required to provide a correct rank ordering.

Conceptually similar to the lambda-dynamics approach is the so-called 'chemical-Monte Carlo/molecular dynamics' method [Pitera and Kollman 1998; Eriksson *et al.* 1999], which also considers many molecules simultaneously. In this approach, molecular dynamics is used to sample the coordinate space with Monte Carlo moves sampling the various chemical states. To avoid possible problems associated with hybrid states the chemical sampling is restricted to jumps between the relevant end-states. At the end of the simulation the relative free energies of the various chemical states is given by the ratio of the populations. Both host–guest and protein–ligand systems have been successfully investigated with the method, which, like the other methods discussed in this section, is designed to rapidly identify which candidates look most promising for further investigation.

The linear response (LR) method was originally devised by Åqvist and co-workers [Åqvist *et al* 1994] for estimating the binding affinities of ligands binding to proteins. Also known as the linear interaction energy (LIE) approach, it is a semi-empirical method for estimating absolute binding free energies and requires just two simulations, one of the solvated ligand–protein system and one of the ligand alone in solution. In both cases, the interaction between the ligand and its environment is broken down into the electrostatic and van der Waals contributions. The free energy of binding is then given by the following expression.

$$\Delta G = \beta(\langle \mathscr{V}^{\text{el}}_{\text{ligand-protein}}\rangle - \langle \mathscr{V}^{\text{el}}_{\text{ligand-solvent}}\rangle) + \alpha(\langle \mathscr{V}^{\text{vdw}}_{\text{ligand-protein}}\rangle - \langle \mathscr{V}^{\text{vdw}}_{\text{ligand-solvent}}\rangle) \quad (11.40)$$

As usual, the angle brackets $\langle\,\rangle$ indicate ensemble averages. α and β are two parameters. To determine ΔG one thus needs to perform just two simulations, one of the ligand in the solvent and the other of the ligand bound to the protein. The interactions that are accumulated consist solely of the electrostatic and van der Waals interactions between the ligand and its environment. First we consider an expansion of the Zwanzig expression for the free energy difference between two states X and Y (Equation (11.6)). The result obtained (see Appendix 11.3) is:

$$\Delta A = \frac{1}{2}[\langle\Delta\mathscr{H}\rangle_0 + \langle\Delta\mathscr{H}\rangle_1] - \frac{1}{4k_BT}[\langle(\Delta\mathscr{H} - \langle\Delta\mathscr{H}\rangle_0)^2\rangle_0 - \langle(\Delta\mathscr{H} - \langle\Delta\mathscr{H}\rangle_1)^2\rangle_1] + \cdots$$

$$\text{where} \quad \Delta\mathscr{H} = \mathscr{H}_Y - \mathscr{H}_X \quad (11.41)$$

For the electrostatic component, the free energy varies in a harmonic fashion with respect to deviations from equilibrium with a constant force constant (Figure 11.19). This is a standard result from dielectric theory and means that the mean square fluctuations of the energy on the two surfaces (the second terms in Equation (11.41)) will cancel, leaving just the first term. This leads to a value of $\frac{1}{2}$ for the electrostatic component (i.e. $\beta = 0.5$). A simple test of this theory is to calculate the electrostatic contribution to solvation free energies. Here, state X corresponds to the situation where all of the solvent–solvent and intramolecular solute interactions are present but the interaction between the solute and solvent is only described

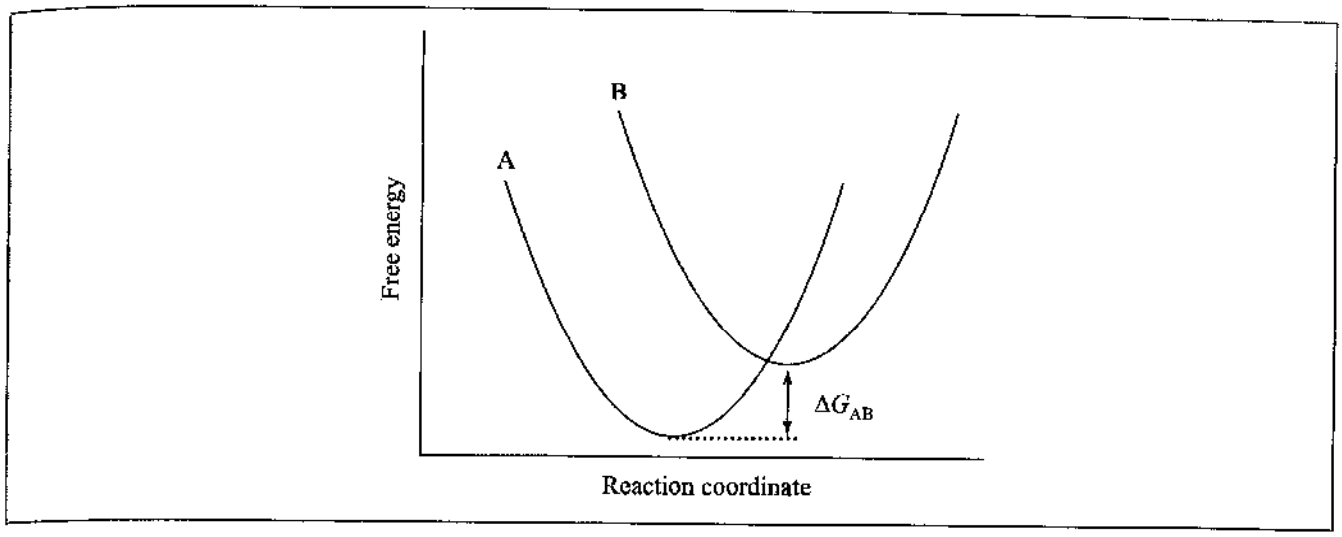


Fig 11.19: Representation of the harmonic variation of the electrostatic component of the free energy according to the linear response approximation

by a Lennard-Jones potential; the solute–solvent electrostatic interactions are missing. In state Y all interactions are included. The only difference between X and Y is thus the presence of the solute–solvent electrostatic terms, and so $\Delta\mathscr{H}$ $(= \mathscr{H}_Y - \mathscr{H}_X)$ in Equation (11.41) is equal to $\mathscr{H}^{el}$(ligand–solvent). Thus:

$$\Delta A^{el}_{sol} = \tfrac{1}{2}\langle \mathscr{H}^{el}_{ligand-solvent} \rangle \tag{11.42}$$

The validity of this result has been confirmed by for example comparing the free energy perturbation result for charging Na^+ and Ca^{2+} ions in water with the ensemble average value of $\mathscr{H}^{el}$(ion–solvent), giving factors of 0.49 and 0.52. Moreover, one can apply the same arguments to the case of a ligand in a protein environment, leading to the following expression for the electrostatic contribution to the free energy of binding (i.e. the first term in Equation (11.40)):

$$\Delta A^{el}_{binding} = \tfrac{1}{2}\left(\langle \mathscr{H}^{el}_{ligand-protein} \rangle - \langle \mathscr{H}^{el}_{ligand-solvent} \rangle\right) \tag{11.43}$$

For the van der Waals component no such analytical theory exists. Åqvist and co-workers assumed that a similar linear treatment would work for these interactions but with a different empirical factor, to be determined from calibration experiments. There was some indirect evidence that this approach would be reasonable. For example, the experimental free energies of solvation for various hydrocarbons (e.g. *n*-alkanes) depend in an approximately linear fashion on the length of the carbon chain. In addition, the mean van der Waals solute–solvent energies from molecular dynamics simulations did show a linear variation with chain length (the slope of the line varying according to the solvent).

What remains is to determine a value of the parameter α. In the original publication this was done using a series of ligands which bind to endothiapepsin, an enzyme for which various crystal structures are known. Some of these ligands are shown in Figure 11.20; as can be seen, they are quite substantial. Molecular dynamics simulations of four ligands were performed within the enzyme binding site and in water and accumulating the required average interaction energies. Assuming the factor $\frac{1}{2}$ for the electrostatic contribution and comparing with the experimental binding affinities gave $\alpha = 0.161$. When a fifth ligand was evaluated,

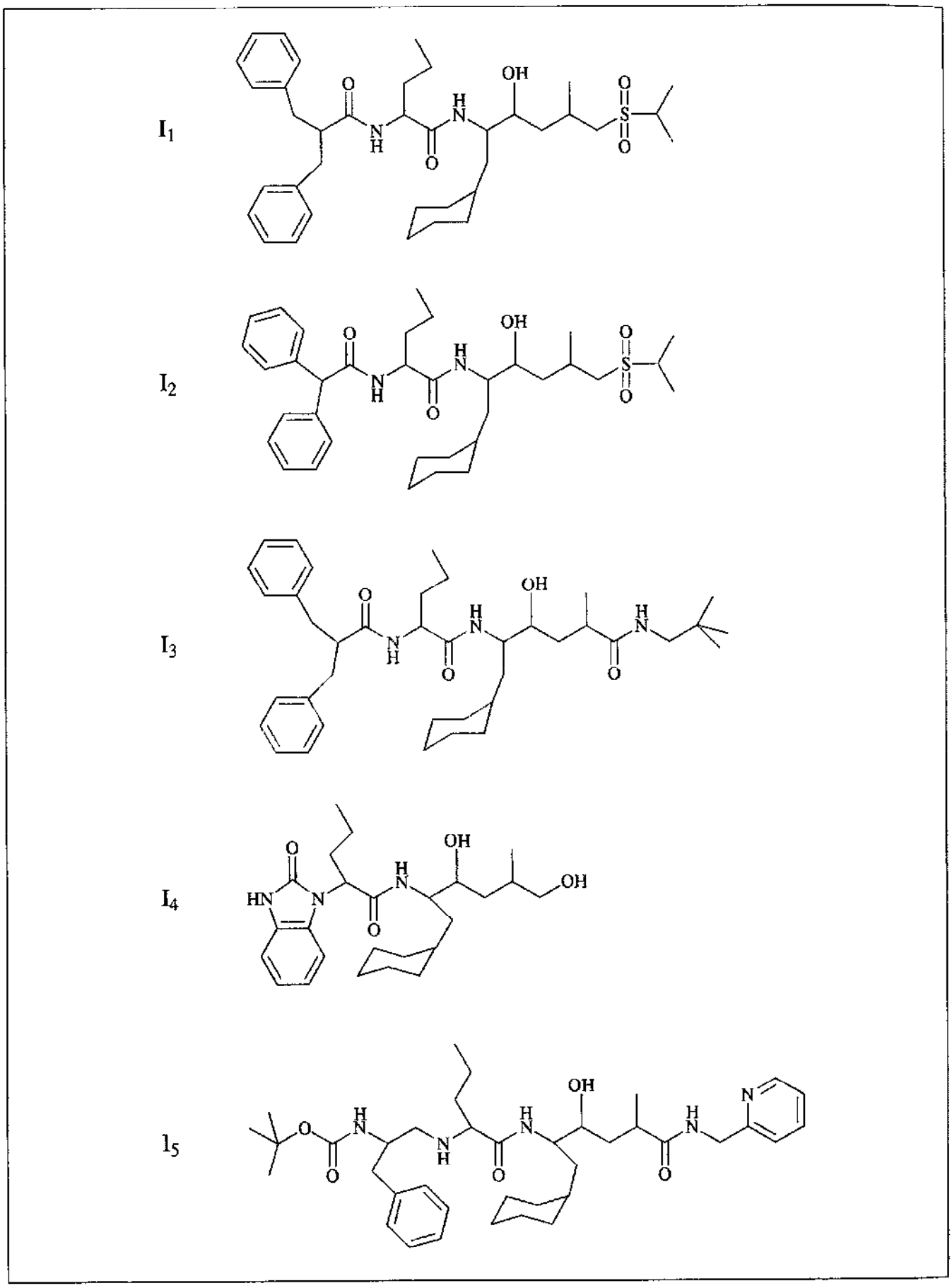


Fig 11 20: Structures of the endothiapepsin ligands used to calibrate the LIE approach.

not present in the calibration set, rather remarkably the predicted free energy of binding was within 0.2 kcal/mol of the experimental result.

Further studies for different ligands and different enzymes appeared to support the approach, and also the constants α and β [Hansson and Åqvist 1995, Hansson *et al.* 1998]. However, other groups found that different values of the van der Waals parameter, α, were required. One possible reason for this discrepancy could be due to the different protocols (for example, different force fields), but some groups found that different parameters were required for different systems, even when the same protocols were employed. One possible explanation for this is that α depends on the nature of the binding site. This is not an unreasonable conclusion, given the different distributions of polar and non-polar groups in different binding sites. Wang and co-workers investigated this variation in more detail and showed that there appeared to be a correlation between the value of α and the 'weighted non-polar desolvation ratio', which is a measure of the hydrophobicity of the binding site [Wang *et al.* 1999]. It was found that, whilst it is generally more accurate to calibrate α for each system if experimental binding data for similar ligands is available, choosing a value based on the weighted non-polar desolvation ratio could give better results for dissimilar compounds.

Other groups have applied the linear response method to problems other than protein-ligand binding. A good problem for any new free energy approach is to predict the free energies of hydration of small organic molecules. Accurate hydration data are available for a wide variety of systems, and the calculations can usually be run relatively quickly. One immediate problem with the two-parameter linear response method is that, as α and β are both positive, it is not possible for any solute to have a positive hydration free energy (both the electrostatic and van der Waals interactions between solutes and water give negative solute–solvent energies). To deal with this problem, Carlson and Jorgensen introduced an additional term which was related to the penalty for forming a solute cavity [Carlsen and Jorgensen 1995]. This third term was proportional to the solvent-accessible surface area:

$$\Delta G_{\text{hyd}} = \beta\langle\mathscr{V}^{\text{el}}\rangle + \alpha\langle\mathscr{V}^{\text{vdw}}\rangle + \gamma SASA \tag{11.44}$$

In their work on hydration, Carlson and Jorgensen attempted to fit the three coefficients α, β and γ, obtaining the best fit for $\alpha = 0.4$, $\beta = 0.45$ and $\gamma = 0.03$ kcal/(mol Å^2). In a subsequent study of the binding of a series of sulphonamide inhibitors to the enzyme thrombin, however, these parameter values were found to be ineffective and that new values were required to give an acceptable fit to the experimental data, with β now being much reduced in value (0.146) [Jones-Hertzog and Jorgensen 1997]. As more variables are considered for inclusion in an LIE-like relationship, it is important that a statistically correct strategy is employed to ensure that the 'optimal' equation is derived (the one with the most predictive power). The techniques to derive such equations are discussed in Section 12.12 on quantitative structure–activity relationships; one study where they were successfully employed considered the binding of a series of inhibitors to the enzyme neuraminidase [Wall *et al.* 1999].

Other attempts to predict free energies from a single simulation have explored the relationship between the coupling parameter, λ, and the free energy. Specifically, the free energy is

expressed as a Taylor series expansion in terms of λ around the point $\lambda = 0$. This expansion is [Smith and van Gunsteren 1994a; Liu *et al.* 1996] (showing just the first two terms explicitly):

$$A(\lambda) = A(\lambda) - A(0) = A'_{\lambda=0}\lambda + \frac{1}{2!}A''_{\lambda=0}\lambda^2 + \frac{1}{3!}A'''_{\lambda=0}\lambda^3 + \cdots$$

$$= \left\langle \frac{\partial \mathscr{H}}{\partial \lambda} \right\rangle_\lambda \lambda + \frac{1}{k_B T}\left\langle \left(\frac{\partial \mathscr{H}}{\partial \lambda} - \left\langle \frac{\partial \mathscr{H}}{\partial \lambda} \right\rangle_0 \right)^2 \right\rangle_0 + \cdots \qquad (11.45)$$

Truncating this series after the first derivative and integrating provides the basis for the thermodynamic integration approach. Moreover, if the Taylor series expansion is continued until it converges then Equation (11.45) is equivalent to the thermodynamic perturbation formula, so providing a link between the two approaches. In practice, it is always necessary to truncate the series; the problem then is whether it is appropriate to assume that the discarded higher-order terms are zero. A good way to test this approach is to consider model systems where the free energy change is known to be zero One such system involves a simple diatomic molecule in a box of water. Each atom of the diatomic molecule is assigned a charge, equal in magnitude (0.25) but of opposite signs. The state to which this system is 'perturbed' corresponds to simply switching the charges (i.e. the start and final states are equivalent) For this system, a standard free energy perturbation calculation can give an answer very close to zero The series expansion was not able to reproduce this result well, even from a 1 ns simulation However, if one considered an alternative problem involving a change from $\lambda = 0$ to $\lambda = 0.5$ (which corresponds to decharging the system) the series expansion did give a very good result. Nevertheless, all these methods were found to fail for calculations involving the creation or deletion of atoms (a problem we discussed above) In the same paper, Liu and colleagues suggested an interesting method that could not only overcome this problem but also enable many free energy values for a series of related ligands to be obtained from a single simulation At those positions where atoms are created or deleted, soft-core interaction sites are used of the form in Equation (11.30). A single long simulation of this (non-physical) reference state is performed. The soft-core potential has a functional form such that solvent molecules can sometimes penetrate 'within' the usual van der Waals radius. This extends the configurational space accessible to the system. Estimates of free energy differences can be obtained by running through the trajectory, substituting the soft-core sites for the appropriate 'real' atoms and calculating the energy for incorporation into the free-energy perturbation formula. In a 'proof-of-concept' illustration, the free energies of hydration for a series of small molecules were calculated from a single simulation consisting of a soft-core cavity in water [Schäfer *et al.* 1999]. These calculations suggested that the efficiency gains over conventional free energy calculations could reach 2–3 orders of magnitude but that the method did require some further development for certain types of system.

11.9 Continuum Representations of the Solvent

Most chemical processes take place in a solvent and so it is clearly important to consider how the solvent affects the behaviour of a system In some cases, solvent molecules are directly

involved, as in ester hydrolysis reactions or in systems where solvent molecules are so tightly bound that they are effectively an integral part of the solute. Such solvent molecules should be modelled explicitly. In other systems, the solvent does not directly interact with the solute but it provides an environment that strongly affects the behaviour of the solute. For example, the highly anisotropic environment in a liquid crystal or lipid bilayer strongly influences the conformations of dissolved solutes. Here, it may not be necessary to explicitly model the solvent molecules, though special treatments such as mean field theories (see Section 7.10) may be required. In the third case, the solvent merely acts as a 'bulk medium' but can still significantly affect solute behaviour, with the dielectric properties of the solvent often being particularly important. In this case, it would clearly be useful not to have to explicitly include every single solvent molecule in the system, to enable us to concentrate on the behaviour of the solute(s). The solvent acts as a perturbation on the gas-phase behaviour of the system. This is the purpose of the 'continuum' solvent models [Smith and Pettitt 1994]. A considerable variety of such models have been proposed, for use with both quantum mechanics and empirical models [Cramer and Truhlar 1992] Our discussion will be restricted to a few of the more widely used methods.

11.9.1 Thermodynamic Background

The solvation free energy (ΔG_{sol}) is the free energy change to transfer a molecule from vacuum to solvent. The solvation free energy can be considered to have three components:

$$\Delta G_{sol} = \Delta G_{elec} + \Delta G_{vdw} + \Delta G_{cav} \tag{11.46}$$

where ΔG_{elec} is the electrostatic component. This contribution is particularly important for polar and charged solutes due to the polarisation of the solvent, which we will model as a uniform medium of constant dielectric ε. ΔG_{vdw} is the van der Waals interaction between the solute and solvent; this may in turn be divided into a repulsive term, ΔG_{rep}, and an attractive dispersion term, ΔG_{disp}. ΔG_{cav} is the free energy required to form the solute cavity within the solvent. This component is positive and comprises the entropic penalty associated with the reorganisation of the solvent molecules around the solute together with the work done against the solvent pressure in creating the cavity. In addition to the above three components, an explicit hydrogen-bonding term, ΔG_{hb}, may be added for those systems where there is localised hydrogen bonding between the solute and solvent. Initially, we will discuss the electrostatic contribution to the free energy of solvation. We will then consider the van der Waals and cavity contributions.

11.10 The Electrostatic Contribution to the Free Energy of Solvation: The Born and Onsager Models

Two important contributions to the study of solvation effects were made by Born (in 1920) and Onsager (in 1936). Born derived the electrostatic component of the free energy of solvation for placing a charge within a spherical solvent cavity [Born 1920], and Onsager extended this to a dipole in a spherical cavity (Figure 11.21) [Onsager 1936]. In the Born

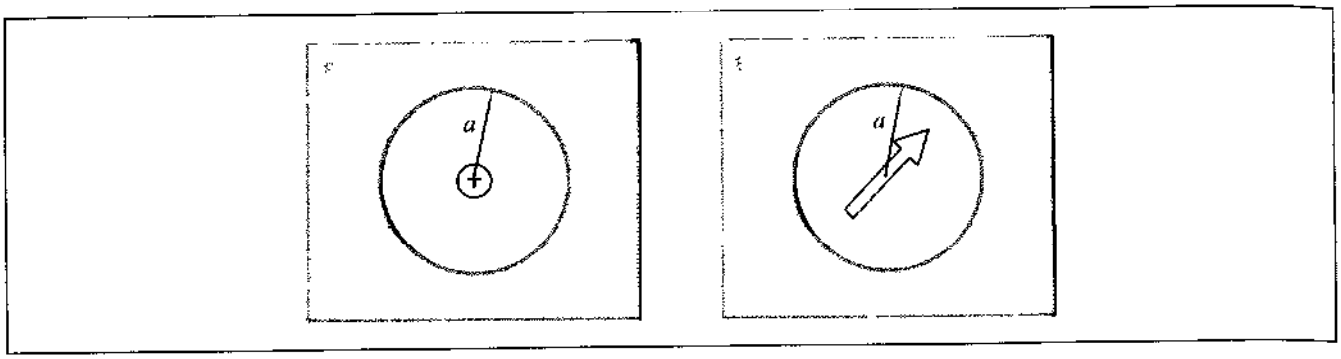


Fig 11 21: The Born and Onsager models

model, ΔG_{elec} of an ion is equal to the work done to transfer the ion from vacuum to the medium. This in turn is equal to the difference in the electrostatic work to charge the ion in the two environments. The work to charge an ion in a medium of dielectric constant ε equals $q^2/2\varepsilon a$, where q is the charge on the ion and a is the radius of the cavity The electrostatic contribution to the solvation free energy is thus the difference in the work done in charging the ion in the dielectric and *in vacuo*:

$$\Delta G_{elec} = -\frac{q^2}{2a}\left(1 - \frac{1}{\varepsilon}\right) \tag{11.47}$$

Note that in this equation, as throughout our discussion, we have used reduced electrostatic units, in which the factor $4\pi\varepsilon_0$ is ignored. This is common practice in the literature. The Born model is very simple yet can be quite successful. It is necessary to choose a set of cavity radii. Traditionally, ionic radii from crystal structures are used However, for the alkali halides it is found that adding 0 1 Å to the radii of anions and 0 85 Å to the radii of cations gives much better agreement with experimental data Justification for this adjustment was provided by Rashin and Honig, who examined electron density distributions in crystals and concluded that the ionic radii are reasonably good indicators of cavity size for anions but that for cations it is more appropriate to use covalent radii [Rashin and Honig 1985]. They subsequently suggested that the optimal agreement with experiment could be obtained by increasing these radii by an empirical factor of 7%.

11.10.1 Calculating the Electrostatic Contribution via Quantum Mechanics

The Born model is obviously only appropriate to species with a formal charge. Onsager's dipole model is relevant to many more molecules (in fact, the Onsager model is a special case of the result derived by Kirkwood [Kirkwood 1934], who considered an arbitrary distribution of charges within a spherical cavity). The solute dipole within the cavity induces a dipole in the surrounding medium, which in turn induces an electric field within the cavity (the *reaction field*). The reaction field then interacts with the solute dipole, so providing additional stabilisation of the system. The magnitude of the reaction field was determined by Onsager to be:

$$\phi_{RF} = \frac{2(\varepsilon - 1)}{(2\varepsilon + 1)a^3}\mu \tag{11.48}$$

where μ is the dipole moment of the solute; a and ε are the radius of the cavity and the dielectric constant of the medium, as before. The energy of a dipole in an electric field ϕ_{RF} is $-\phi_{RF}\mu$, but for a polarisable dipole it is necessary to add an additional term which represents the work done assembling the charge distribution within the cavity. This additional term has magnitude $\phi_{RF}\mu/2$ and so the electrostatic contribution to the free energy of solvation in this model is:

$$\Delta G_{elec} = -\frac{\phi_{RF}\mu}{2} \tag{11.49}$$

If the species is charged then an appropriate Born term must also be added. The reaction field model can be incorporated into quantum mechanics, where it is commonly referred to as the *self-consistent reaction field* (SCRF) method, by considering the reaction field to be a perturbation of the Hamiltonian for an isolated molecule. The modified Hamiltonian of the system is then given by:

$$\mathscr{H}_{tot} = \mathscr{H}_0 + \mathscr{H}_{RF} \tag{11.50}$$

where $\mathscr{H}_0$ is the Hamiltonian of the isolated molecule and $\mathscr{H}_{RF}$ is the perturbation, given by [Tapia and Goscinski, 1975]:

$$\mathscr{H}_{RF} = -\hat{\mu}^T \frac{2(\varepsilon - 1)}{(2\varepsilon + 1)a^3} \langle \Psi | \hat{\mu} | \Psi \rangle \tag{11.51}$$

where $\hat{\mu}$ is the dipole moment operator written in matrix form and $\hat{\mu}^T$ is its transpose. The wavefunction Ψ for the modified Hamiltonian is determined and the electrostatic contribution to the solvation free energy is then given by:

$$\Delta G_{elec} = \langle \Psi | \mathscr{H}_{tot} | \Psi \rangle - \langle \Psi_0 | \mathscr{H}_0 | \Psi_0 \rangle + \frac{1}{2} \frac{2(\varepsilon - 1)}{(2\varepsilon + 1)a^3} \mu^2 \tag{11.52}$$

The third term in Equation (11.52) is the correction factor corresponding to the work done in creating the charge distribution of the solute within the cavity in the dielectric medium. Ψ_0 is the gas-phase wavefunction.

A drawback of the SCRF method is its use of a spherical cavity; molecules are rarely exactly spherical in shape. However, a spherical representation can be a reasonable first approximation to the shape of many molecules. It is also possible to use an ellipsoidal cavity; this may be a more appropriate shape for some molecules. For both the spherical and ellipsoidal cavities analytical expressions for the first and second derivatives of the energy can be derived, so enabling geometry optimisations to be performed efficiently. For these cavities it is necessary to define their size. In the case of a spherical cavity a value for the radius can be calculated from the molecular volume:

$$a^3 = 3V_m/4\pi N_A \tag{11.53}$$

The molecular volume V_m can in turn be obtained by dividing the molecular weight by the density or from refractivity measurements; N_A is Avogadro's number. The cavity radius can also be estimated from the largest interatomic distance within the molecule. A third approach is to calculate the 'volume' of the molecule from a suitable electron density contour. The radii obtained by these procedures are often adjusted by adding an empirical constant to give the 'true' cavity radius. This extra value accounts for the fact that solvent

molecules cannot approach right up to the molecule. An additional extension to the simple SCRF procedure is the use of a multipolar expansion to represent the solute [Rinaldi *et al.* 1983]. This overcomes a drawback of the basic model in which a molecule with a zero dipole would have zero solvation energy.

A yet more realistic cavity shape is that obtained from the van der Waals radii of the atoms of the solute. This is the approach taken in the *polarisable continuum* method (PCM) [Miertus *et al.* 1981], which has been implemented in a variety of *ab initio* and semi-empirical quantum mechanical programs. Due to the non-analytical nature of the cavity shapes in the PCM approach, it is necessary to calculate ΔG_{elec} numerically. The cavity surface is divided into a large number of small surface elements, and there is a point charge associated with each surface element. This system of point charges represents the polarisation of the solvent, and the magnitude of each surface charge is proportional to the electric field gradient at that point. The total electrostatic potential at each surface element equals the sum of the potential due to the solute and the potential due to the other surface charges:

$$\phi(\mathbf{r}) = \phi_p(\mathbf{r}) + \phi_\sigma(\mathbf{r}) \tag{11 54}$$

where $\phi_p(\mathbf{r})$ is the potential due to the solute and $\phi_\sigma(\mathbf{r})$ is the potential due to the surface charges. The PCM algorithm is as follows. First, the cavity surface is determined from the van der Waals radii of the atoms. That fraction of each atom's van der Waals sphere which contributes to the cavity is then divided into a number of small surface elements of calculable surface area. The simplest way to to this is to define a local polar coordinate frame at the centre of each atom's van der Waals sphere and to use fixed increments of $\Delta\theta$ and $\Delta\phi$ to give rectangular surface elements (Figure 11.22). The surface can also be divided using tessellation methods [Paschual-Ahuir *et al.* 1987] An initial value of the point charge for each surface element is then calculated from the electric field gradient due to the solute alone:

$$q_i = -\left[\frac{\varepsilon - 1}{4\pi\varepsilon}\right] E_i \Delta S \tag{11 55}$$

where ε is the dielectric constant of the medium, E_i is the electric field gradient and ΔS is the area of the surface element. The contribution $\phi_\sigma(\mathbf{r})$ due to the other point charges can then be calculated using Coulomb's law. These charges are modified iteratively until they are

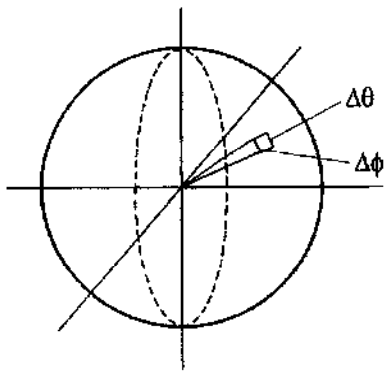


Fig 11 22 Small surface elements can be created on the van der Waals surface of an atom using constant increments of the polar angles, θ and ϕ

self-consistent. The potential $\phi_\sigma(\mathbf{r})$ from the final part of the charge is then added to the solute Hamiltonian ($\mathscr{H} = \mathscr{H}_0 + \phi_\sigma(\mathbf{r})$) and the SCF calculation initiated. After each SCF calculation new values of the surface charges are calculated from the current wavefunction to give a new value of $\phi_\sigma(\mathbf{r})$ which is used in the next iteration until the solute wavefunction and the surface charges are self-consistent.

To calculate ΔG_{elec}, we must take account of the work done in creating the charge distribution within the cavity in the dielectric medium. This is equal to one-half of the electrostatic interaction energy between the solute charge distribution and the polarised dielectric, and so:

$$\Delta G_{\text{elec}} = \int \Psi \mathscr{H} \Psi \, d\tau - \int \Psi_0 \mathscr{H}_0 \Psi_0 \, d\tau - \tfrac{1}{2} \int \phi(\mathbf{r}) \rho(\mathbf{r}) \, d\mathbf{r} \tag{11.56}$$

where $\rho(\mathbf{r})$ is the charge distribution of the surface elements.

There are two slight complications with the PCM approach. The first of these arises as a consequence of representing a continuous charge distribution over the cavity surface as a set of single point charges. When calculating the electrostatic potential due to the charges on the surface elements one must exclude the charge for the current surface element. To include it would cause the charges to diverge rather than converge. The contribution of the charge on that surface element is therefore determined separately using the Gauss theorem. The second complication arises because the wavefunction of the solute extends beyond the cavity. Thus the sum of the charges on the surface is not equal and opposite to the charge of the solute This problem can be easily overcome by scaling the charge distribution on the surface so that it is equal and opposite to the charge of the solute.

COSMO is an interesting variant on the PCM method (COSMO stands for 'conductor-like screening model') [Klamt and Schüürmann 1993; Klamt 1995, Klamt *et al* 1998]. The cavity is considered to be embedded in a conductor with an infinite dielectric constant. The advantage of this is that screening effects in an infinitely strong dielectric (i.e a conductor) are much easier to handle. A small correction to the results for this conductor can provide the appropriate value for water, which of course has a high dielectric constant. On the surface of a conductor the potential due to the solute and due to the surface charges is set to zero, which gives rise to a convenient boundary condition when determining the surface charges. For an alternative dielectric these charges are scaled by a factor:

$$q' = q \frac{\varepsilon_r - 1}{\varepsilon_r + 0.5} \tag{11 57}$$

The SCRF and PCM models have been used to investigate the effect of solvent upon energetics and equilibria. For example, Wong, Wiberg and Frisch used the SCRF method to investigate the effect of different solvents upon the tautomeric equilibria of 2-pyridone (Figure 11.23) [Wong *et al.* 1992]. Geometry optimisations were performed for the various tautomeric species at high levels of theory, and vibrational frequencies were calculated. Results were reported for the gas phase, for a non-polar solvent (cyclohexane, $\varepsilon = 2.0$) and for an aprotic polar solvent (acetonitrile, $\varepsilon = 35.9$). The calculated free energy changes in the gas phase, cyclohexane and acetonitrile were −0 64 kcal/mol, 0.36 kcal/mol and 2.32 kcal/mol, respectively, which compared favourably with the experimental values of −0 81 kcal/mol, 0.33 kcal/mol and 2.96 kcal/mol. The dielectric medium was found to have a much more pronounced effect

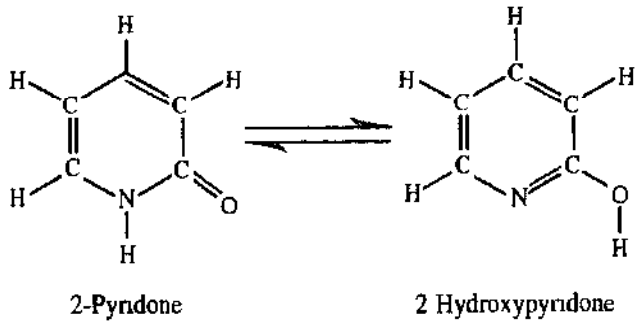


Fig 11 23: Tautomers of 2-pyridone

on the structure, charge distribution and vibrational frequencies of the keto form than of the enol form. This was ascribed to the more polar nature of the keto tautomer.

11.10.2 Continuum Models for Molecular Mechanics

In many cases, solvent effects can be incorporated into a force field model using one of the theories that we have just examined. It is possible to study larger systems with the empirical models, in which case it is necessary to take account of the dielectric properties of the solute as well as those of the solvent. Before reading this section it may be useful to revise the definitions of molecular surface and accessible surface given in Section 1.5, as these are widely referenced.

The *boundary element method* of Rashin is similar in spirit to the polarisable continuum model, but the surface of the cavity is taken to be the molecular surface of the solute [Rashin and Namboodiri 1987; Rashin 1990]. This cavity surface is divided into small boundary elements. The solute is modelled as a set of atoms with point polarisabilities. The electric field induces a dipole proportional to its polarisability. The electric field at an atom has contributions from dipoles on other atoms in the molecule, from polarisation charges on the boundary, and (where appropriate) from the charges of electrolytes in the solution. The charge density is assumed to be constant within each boundary element but is not reduced to a single point as in the PCM model. A set of linear equations can be set up to describe the electrostatic interactions within the system. The solutions to these equations give the boundary element charge distribution and the induced dipoles, from which thermodynamic quantities can be determined.

The *generalised Born equation* has been widely used to represent the electrostatic contribution to the free energy of solvation [Constanciel and Contreras 1984]. The model comprises a system of particles with radii a_i and charges q_i. The total electrostatic free energy of such a system is given by the sum of the Coulomb energy and the Born free energy of solvation in a medium of relative permittivity ε:

$$G_{\text{elec}} = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{q_i q_j}{\varepsilon r_{ij}} - \frac{1}{2}\left(1 - \frac{1}{\varepsilon}\right) \sum_{i=1}^{N} \frac{q_i^2}{a_i} \qquad (11\ 58)$$

The first term in Equation (11.58) can be written as the sum of a Coulomb interaction *in vacuo* and a second term in $(1 - 1/\varepsilon)$:

$$\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{q_iq_j}{\varepsilon r_{ij}} = \sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{q_iq_j}{r_{ij}} - \left(1-\frac{1}{\varepsilon}\right)\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{q_iq_j}{r_{ij}} \tag{11.59}$$

In the generalised Born approach the total electrostatic energy is written as a sum of three terms, the first of which is the Coulomb interaction between the charges *in vacuo*.

$$G_{\text{elec}} = \sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{q_iq_j}{r_{ij}} - \left(1-\frac{1}{\varepsilon}\right)\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{q_iq_j}{r_{ij}} - \frac{1}{2}\left(1-\frac{1}{\varepsilon}\right)\sum_{i=1}^{N}\frac{q_i^2}{a_i} \tag{11.60}$$

where ΔG_{elec} equals the difference between G_{elec} and the Coulomb energy *in vacuo*. This is the generalised Born (GB) equation:

$$\Delta G_{\text{elec}} = -\left(1-\frac{1}{\varepsilon}\right)\sum_{i=1}^{N}\sum_{j=i+1}^{N}\frac{q_iq_j}{r_{ij}} - \frac{1}{2}\left(1-\frac{1}{\varepsilon}\right)\sum_{i=1}^{N}\frac{q_i^2}{a_i} \tag{11.61}$$

The generalised Born equation has been incorporated into both molecular mechanics calculations (by Still and co-workers [Still *et al.* 1990; Qiu *et al.* 1997]) and semi-empirical quantum mechanics calculations (by Cramer and Truhlar, in an ongoing series of models called SM1, SM2, SM3, etc. [Cramer and Truhlar 1992; Chambers *et al.* 1996]). In these treatments, the two terms in Equation (11.61) are combined into a single expression of the following form:

$$\Delta G_{\text{elec}} = -\frac{1}{2}\left(1-\frac{1}{\varepsilon}\right)\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{q_iq_j}{f(r_{ij}, a_{ij})} \tag{11.62}$$

where $f(r_{ij}, a_{ij})$ depends upon the interparticle distances r_{ij} and the Born radii a_i. A variety of forms are possible for the function f; that proposed by Still and colleagues was:

$$f(r_{ij}, a_{ij}) = \sqrt{(r_{ij}^2 + a_{ij}^2 e^{-D})} \quad \text{where} \quad a_{ij} = \sqrt{(a_ia_j)} \quad \text{and} \quad D = r_{ij}^2/(2a_{ij})^2 \tag{11.63}$$

This form of the function f can be justified for the following reasons. When $i = j$, the equation returns the Born expression; for two charges close together (i.e. a dipole, in which r_{ij} is small compared to a_i and a_j) the expression is close to the Onsager result; and for two charges separated by a significant distance ($r_{ij} \gg a_i, a_j$) the result is very close to the sum of the Coulomb and Born expressions. A further advantage of this functional form is that the expression can be differentiated analytically, thereby enabling the solvation term to be included in gradient-based optimisation methods and molecular dynamics simulations

A rather complex procedure is used to determine the Born radii a_i, values of which are calculated for each atom in the molecule that carries a charge or a partial charge The Born radius of an atom (more correctly considered to be an 'effective' Born radius) corresponds to the radius that would return the electrostatic energy of the system according to the Born equation if all other atoms in the molecule were uncharged (i.e. if the other atoms only acted to define the dielectric boundary between the solute and the solvent). In Still's force field implementation, atomic radii from the OPLS force field are assigned to each

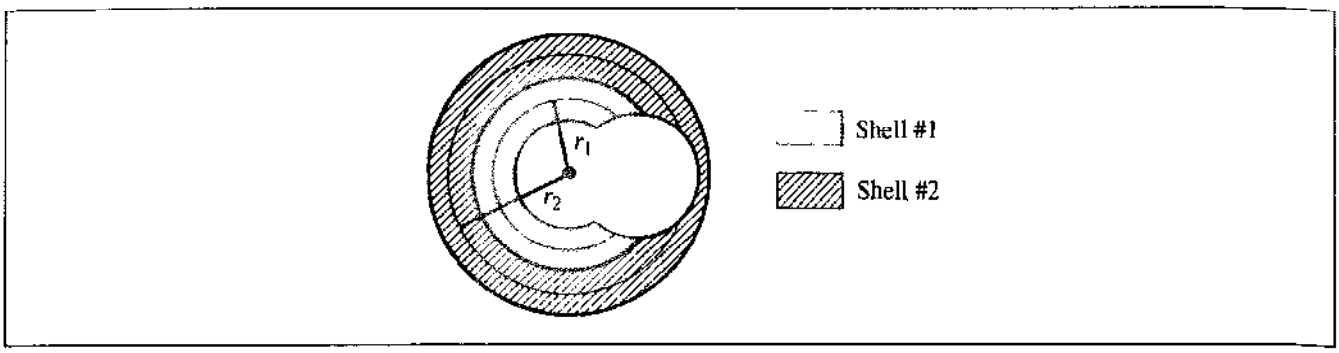


Fig 11.24. Calculation of the effective Born radius in the generalised Born model Shells are constructed until they contain the entire molecule For each shell the amount of exposed surface area is determined for the middle of the shell (Figure adapted from Still W C, A Tempczyrk, R C Hawley and T Hendrickson 1990 Semianalytical Treatment of Solvation for Molecular Mechanics and Dynamics Journal of the American Chemical Society *112.6127–6129*)

atom and amended by an empirically determined offset of −0.09 Å to define the dielectric boundary. In the quantum mechanical approach of Cramer and Truhlar, the radius of each atom is a function of the charge on the atom. The dielectric boundary is then taken to be the union of the relevant radii.

The electrostatic energy of an atom i is calculated numerically by constructing a series of spherical shells until the outer shell (shell M) entirely contains the entire van der Waals surface of the molecule, as shown in Figure 11 24. The Born electrostatic energies of the dielectric in these shells are determined using the following equation:

$$\Delta G_{\text{elec}} = -\frac{1}{2}\left(1-\frac{1}{\varepsilon}\right)q_i^2\left\{\sum_{k=1}^{M}\frac{A_k}{4\pi r_k^2}\left[\left(\frac{1}{r_k-0\,5T_k}\right)-\left(\frac{1}{r_k+0\,5T_k}\right)\right]+\frac{1}{r_{M+1}-0.5T_{M+1}}\right\} \tag{11 64}$$

The radius of the kth shell (r_k) is measured at the middle of the shell. A_k is the amount of surface area of a sphere of radius r_k that is not contained within the van der Waals surface of the molecule. The thickness of each shell, T_k, increases with distance from the atom as follows:

$$T_{k+1} = (1+F)T_k \tag{11.65}$$

where F (the expansion factor) and T_1 (the radius of the first shell) are parameters, chosen by Still to be 0.5 and 0.1 Å, respectively. The shells are constructed from where the dielectric boundary commences (i.e. in Still's case the shells start 0.9 Å inside the van der Waals radii). The final term in Equation (11.65) is the contribution due to the dielectric that lies beyond the van der Waals surface of the molecule. The effective Born radius is then given by equating Equation (11.65) with the Born equation for the atom, and so:

$$\frac{1}{a_i} = \sum_{k=1}^{M}\frac{A_k}{4\pi r_k^2}\left[\left(\frac{1}{r_k-0\,5T_k}\right)-\left(\frac{1}{r_k+0.5T_k}\right)\right]+\frac{1}{r_{M+1}-0.5T_{M+1}} \tag{11.66}$$

The effective Born radii do not change very much and so are recalculated whenever the non-bonded list is updated The Still formulation of the generalised Born equation requires the surface areas A_k of the spherical shells that are exposed to solvent to be calculated For

this, a fast numerical method devised by Wodak and Janin [Wodak and Janin 1980] is used in which the accessible surface area is given by:

$$A_i = S_i \prod_j (1.0 - b_{ij}/S_i) \tag{11.67}$$

where S_i is the total accessible surface area of an atom i with radius r_i as defined with a solvent probe of radius r_s. b_{ij} is the amount of surface area removed due to overlap with an atom j which is a distance d_{ij} from atom i:

$$S_i = 4\pi(r_i + r_s)^2 \tag{11.68}$$

$$b_{ij} = \pi(r_i + r_s)(r_j + r_i - 2r_s - d_{ij})[1.0 + (r_i - r_j)/d_{ij}] \tag{11.69}$$

The Wodak–Janin method is only approximate for more than two spheres. Exact values of A_i can be calculated, but only with a significant computational effort A comparison of results obtained with the approximate and exact methods showed that, for molecules significantly larger than the probe, the approximation was valid (Wodak and Janin's expression was intended to be used to study solvent effects in proteins). Still showed that it was also possible to reduce the b_{ij} term by an empirical constant and obtain accurate results for smaller systems.

The generalised Born model has been incorporated into a number of quantum mechanical and molecular mechanical programs. Still and his group have made extensive use of the model in conformational searching and for calculating relative free energies of binding with free energy perturbation methods. For example, the relative free energies of binding of D- and L-enantiomeric α-amino acid-derived substrates to a podand ionophore (1; Figure 11.25) were calculated to be in good agreement with experiment using a mixed Monte Carlo/dynamics method and the generalised Born model for chloroform [Burger *et al.* 1994]. Similar calculations were then used to predict which of a variety of substituted ionophores (varying the group X in Figure 11.25) would be expected to show the greatest selectivity for the guest (Y = NHMe). The derivative **2** was predicted to show the greatest enantioselectivity. Unfortunately, this particular compound was too insoluble to measure binding affinities, but a related compound, **3**, did show the desired selectivity. Moreover, when the calculations were repeated on **3** the predicted difference in binding affinity was within 0.3 kcal/mol of the experimental result. Such studies clearly illustrate the potential applicability of such calculations, but it should be noted that this system was carefully chosen to minimise any errors associated with the force field parameters (through the use of enantiomeric guests) and sampling (as the host is locked into a single binding conformation). Even so, to achieve accurate results it was usually necessary to perform simulations of the order of 10 ns; at the time such simulations could only be realistically achieved using a continuum model of the solvent.

11.10.3 The Langevin Dipole Model

The Langevin dipole method of Warshel and Levitt [Warshel and Levitt 1976] is intermediate between a continuum and an explicit solvation model. A three-dimensional

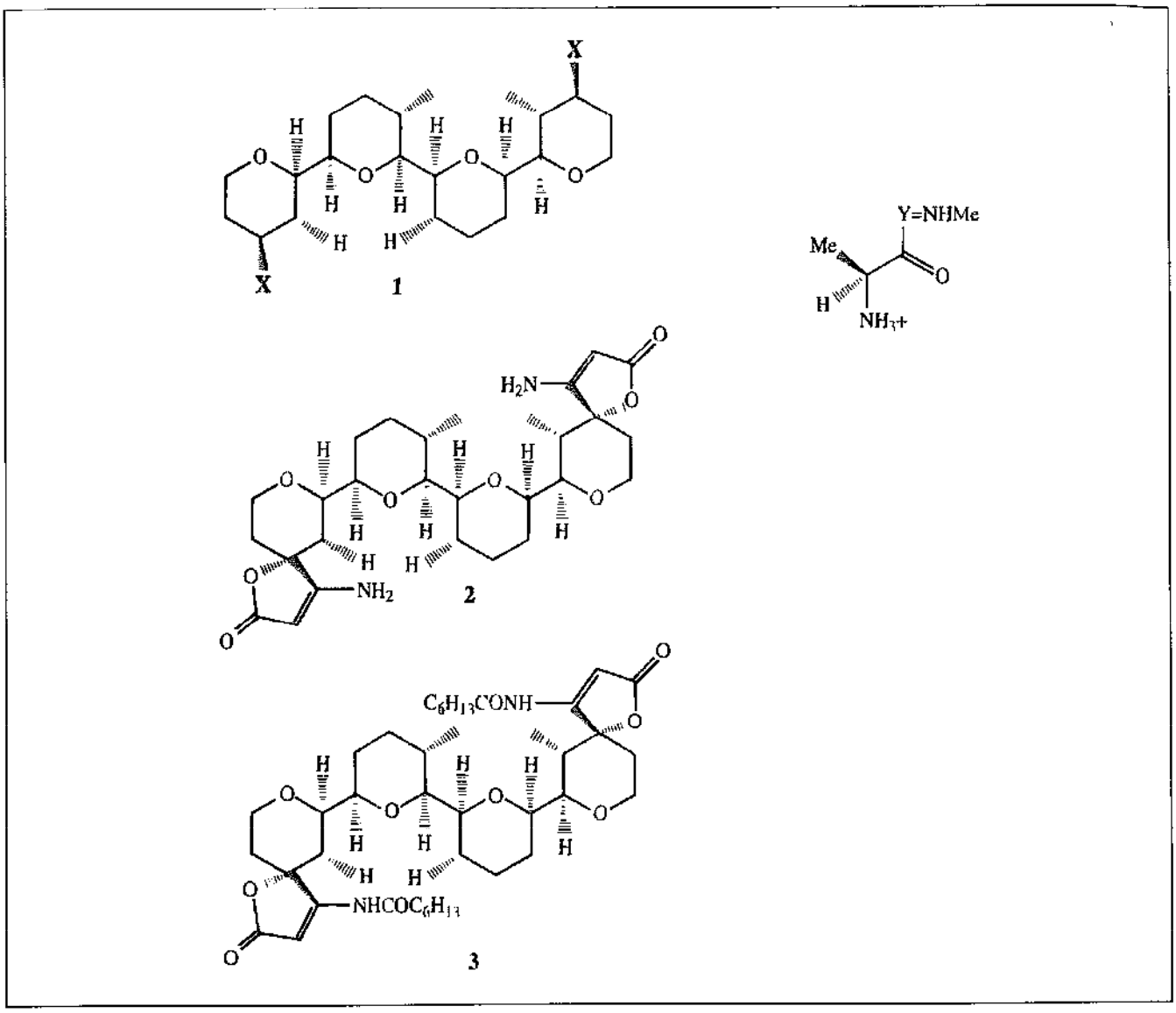


Fig 11.25 Ionophores that selectively bind amino acids [Burger et al *1994].*

grid of rotatable point dipoles is established in the region beyond the boundary (which can be of arbitrary shape; Figure 11.26). For macromolecules the boundary corresponds to the solvent accessible surface. These dipoles represent the molecular dipoles of the solvent molecules in the outer region, and the separation between them is chosen accordingly. The electric field $\mathbf{E}_i$ at each dipole has a contribution from the solute and from other solvent dipoles. The size and direction of each dipole is determined using the Langevin equation:

$$\boldsymbol{\mu}_i = \mu_0 \frac{\mathbf{E}_i}{|\mathbf{E}_i|}\left[\frac{\exp[C\mu_0|\mathbf{E}_i|/k_\mathrm{B}T] + \exp[-C\mu_0|\mathbf{E}_i|/k_\mathrm{B}T]}{\exp[C\mu_0|\mathbf{E}_i|/k_\mathrm{B}T] - \exp[-C\mu_0|\mathbf{E}_i|/k_\mathrm{B}T]} - \frac{1}{C\mu_0|\mathbf{E}_i|/k_\mathrm{B}T}\right] \tag{11.70}$$

where μ_0 is the size of the dipole moment of a solvent molecule and C is a parameter that represents the degree to which the dipoles resist reorientation; its value may be obtained from a separate simulation using explicit solvent Converged values of the dipoles are usually obtained within a few iterations The free energy of the Langevin dipoles is then

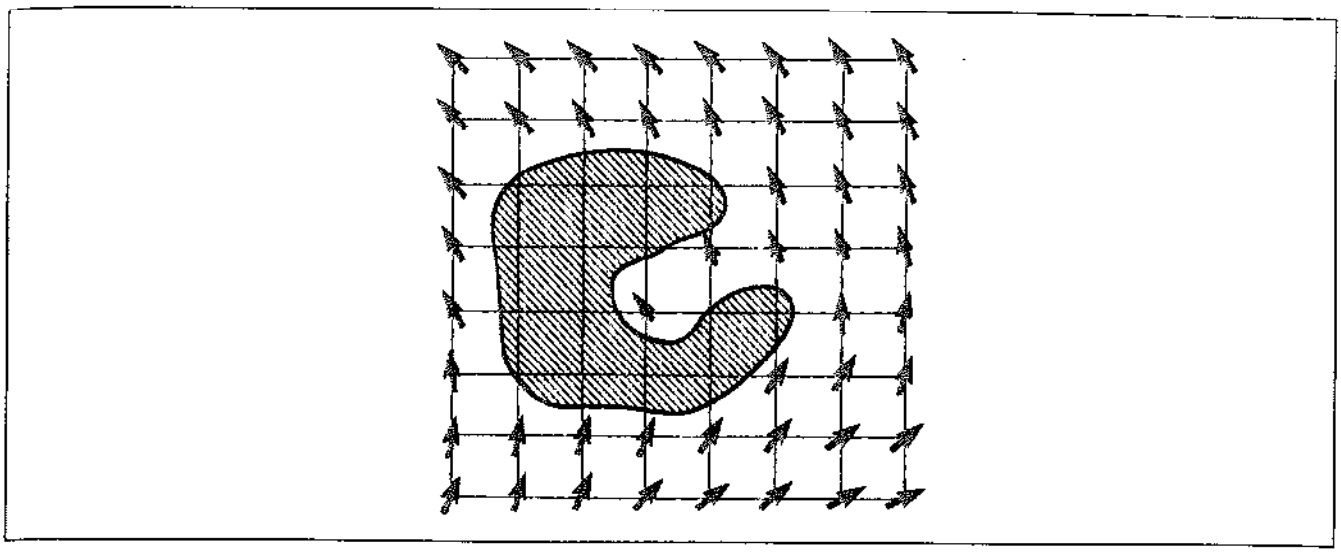


Fig 11 26: The Langevin dipole model

given by:

$$\Delta G_{\text{sol}} = -\tfrac{1}{2}\sum_i \boldsymbol{\mu}_i \cdot \mathbf{E}_i^0 \tag{11.71}$$

$\mathbf{E}_i^0$ is the field due to the solute charges alone. The Langevin dipole method has been widely used by Warshel in his studies of enzyme reactions (see Section 11.13.3).

11.10.4 Methods Based upon the Poisson–Boltzmann Equation

The final class of methods that we shall consider for calculating the electrostatic component of the solvation free energy are based upon the Poisson or the Poisson–Boltzmann equations. These methods have been particularly useful for investigating the electrostatic properties of biological macromolecules such as proteins and DNA. The solute is treated as a body of constant low dielectric (usually between 2 and 4), and the solvent is modelled as a continuum of high dielectric. The Poisson equation relates the variation in the potential ϕ within a medium of uniform dielectric constant ε to the charge density ρ:

$$\nabla^2\phi(\mathbf{r}) = -\frac{\rho(\mathbf{r})}{\varepsilon_0\varepsilon} \tag{11.72}$$

In reduced electrostatic units, the factor $4\pi\varepsilon_0$ is eliminated and the Poisson equation becomes:

$$\nabla^2\phi(\mathbf{r}) = -\frac{4\pi\rho(\mathbf{r})}{\varepsilon} \tag{11 73}$$

The charge density is simply the distribution of charge throughout the system and has SI units of $\mathrm{C\,m^{-3}}$. The Poisson equation is thus a second-order differential equation (∇^2 is the usual abbreviation for $(\partial^2/\partial x^2) + (\partial^2/\partial y^2) + (\partial^2/\partial z^2)$). For a set of point charges in a constant dielectric the Poisson equation reduces to Coulomb's law. However, if the dielectric

is not constant but varies with position, then Coulomb's law is not applicable and the Poisson equation is:

$$\nabla \cdot \varepsilon(\mathbf{r})\nabla\phi(\mathbf{r}) = -4\pi\rho(\mathbf{r}) \tag{11.74}$$

The Poisson equation must be modified when mobile ions are present, to account for their redistribution in the solution in response to the electric potential. The ions are prevented from congregating at the locations of extreme electrostatic potential due to repulsive interactions with other ions and their natural thermal motion. The ion distribution is described by a Boltzmann distribution of the following form:

$$n(\mathbf{r}) = \mathcal{N}\exp(-\mathcal{V}(\mathbf{r})/k_{\mathrm{B}}T) \tag{11.75}$$

where $n(\mathbf{r})$ is the number density of ions at a particular location r, $\mathcal{N}$ is the bulk number density and $\mathcal{V}(\mathbf{r})$ is the energy change to bring the ion from infinity to the position $\mathbf{r}$. When these effects are incorporated into the Poisson equation the result is the *Poisson–Boltzmann equation*:

$$\nabla \cdot \varepsilon(\mathbf{r})\nabla\phi(\mathbf{r}) - \kappa' \sinh[\phi(\mathbf{r})] = -4\pi\rho(\mathbf{r}) \tag{11.76}$$

κ' is related to the Debye–Hückel inverse length, κ, by:

$$\kappa^2 = \frac{\kappa'^2}{\varepsilon} = \frac{8\pi N_{\mathrm{A}} e^2 I}{1000\varepsilon k_{\mathrm{B}} T} \tag{11.77}$$

where e is the electronic charge, I is the ionic strength of the solution and N_{A} is Avogadro's number. This is a non-linear differential equation that can be written is an alternative form by expanding the hyperbolic sine function as a Taylor series:

$$\nabla \cdot \varepsilon(\mathbf{r})\nabla\phi(\mathrm{r}) - \kappa'\phi(\mathbf{r})\left[1 + \frac{\phi(\mathbf{r})^2}{6} + \frac{\phi(\mathbf{r})^4}{120} + \cdots\right] = -4\pi\rho(\mathbf{r}) \tag{11.78}$$

The linearised Poisson–Boltzmann equation is obtained by taking only the first term in the expansion, giving:

$$\nabla \cdot \varepsilon(\mathbf{r})\nabla\phi(\mathbf{r}) - \kappa'\phi(\mathbf{r}) = -4\pi\rho(\mathbf{r}) \tag{11.79}$$

How can Equation (11.79) be solved? Before computers were available only simple shapes could be considered. For example, proteins were modelled as spheres or ellipses (Tanford–Kirkwood theory); DNA as a uniformly charged cylinder; and membranes as planes (Gouy–Chapman theory). With computers, numerical approaches can be used to solve the Poisson–Boltzmann equation. A variety of numerical methods can be employed, including finite element and boundary element methods, but we will restrict our discussion to the finite difference method first introduced for proteins by Warwicker and Watson [Warwicker and Watson 1982]. Several groups have implemented this method; here we concentrate on the work of Honig's group, whose DelPhi program has been widely used.

A cubic lattice is superimposed onto the solute(s) and the surrounding solvent. Values of the electrostatic potential, charge density, dielectric constant and ionic strength are assigned to each grid point. The atomic charges do not usually coincide with a grid point and so the

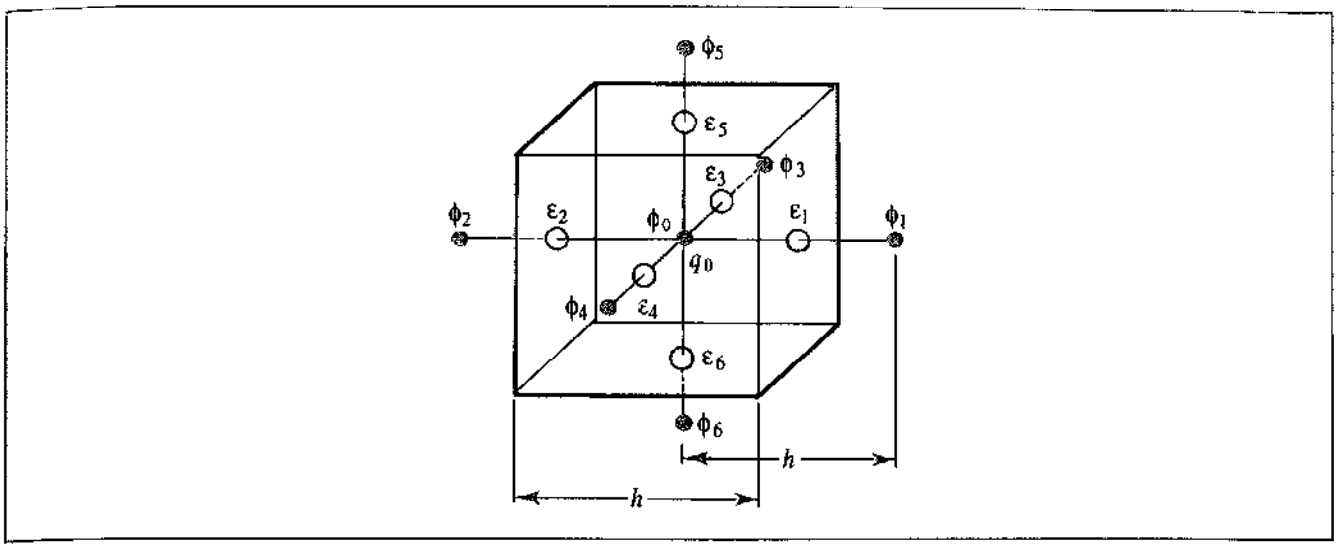


Fig 11 27: The cube used in the finite difference method for solving the Poisson–Boltzmann equation (Figure adapted from Klapper I, R Hagstrom, R Fine, K Sharp and B Honig 1986 Focusing of Electric Fields in the Active Site of Cu–Zn Superoxide Dismutase: Effects of Ionic Strength and Amino-Acid Substitution Proteins Structure, Function and Genetics *1·47–59)*

charge is allocated to the eight surrounding grid points in such a way that the closer the charge to the grid point, the greater the proportion of its total charge that is allocated. The derivatives in the Poisson–Boltzmann equation are then determined by a finite difference formula. Consider the cube of side h surrounding the grid point shown in Figure 11.27. A charge q_0 is associated with the grid point; this is equivalent to a uniform charge density of q_0/h^3 within the cube (i.e. $\rho_0 = q_0/h^3$). The potential at the grid point is given by:

$$\phi_0 = \frac{\sum \varepsilon_i \phi_i + 4\pi \dfrac{q_0}{h}}{\sum \varepsilon_i + k_0'^2 f(\phi_0)} \tag{11.80}$$

The summations are over the potentials ϕ_i at the six adjoining grid points and the dielectric constants ε_i which are associated with the midpoints of the lines between the grid points. The function $f(\phi_0)$ in the denominator has the value 1 for the linear Poisson–Boltzmann equation, and is equivalent to the series expansion $(1 + \phi_0^2/6 + \phi_0^4/120 + \ldots)$ for the non-linear case. κ'^2 is obtained from the ionic strength at the grid point. The crucial feature is that the potential at each grid point influences the potential at the neighbouring grid points, and so by iteratively repeating the calculation converged values will eventually be obtained.

To perform a Poisson–Boltzmann calculation it is necessary to allocate a value for the dielectric constant to each grid point, which requires us to decide which grid points lie within the solute(s) and which are in the solvent. The boundary between the solute and solvent is defined as either the molecular surface or the accessible surface. All grid points outside this surface are assigned a high dielectric constant (80 for water) and an ionic strength value. Grid points within the surface are assigned the dielectric constant of the

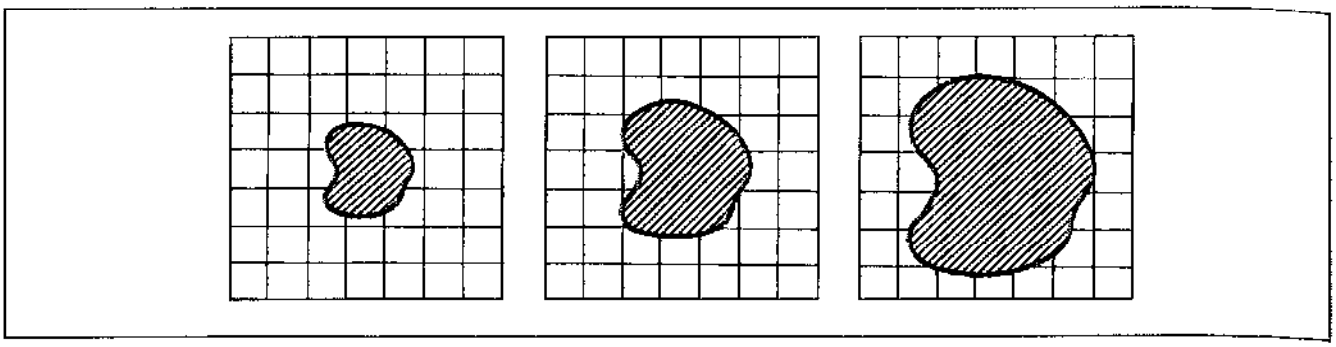


Fig 11.28: Focusing can improve the accuracy of finite difference Poisson–Boltzmann calculations

macromolecule, which is usually considered to lie between 2 and 4. This value of the dielectric constant is justified by the following arguments. The dielectric constant of a material is due to several factors, including its inherent polarisability and its ability to reorient internal dipoles within a changing electric field. A molecule that is fixed in conformation will not be able to change the orientations of its dipolar groups and so the only contribution to the dielectric constant will be due to polarisation effects. Polarisation effects alone lead to a dielectric constant of about 2 for organic liquids. If the conformation of the molecule can change then the dipolar effects should be taken into account, leading to an increased dielectric of 4. The atomic charges and van der Waals radii are often taken directly from an existing force field, though parameter sets designed specifically for use with the Poisson-Boltzmann method have been developed [Sitkoff *et al.* 1994].

The correct choice of grid size can be crucial to the success of a finite difference Poisson-Boltzmann calculation The finer the lattice, the more accurate the results, though more computer time will be required. A grid size of 65^3 has been widely used. A technique called *focusing* can help alleviate some of these problems. In this method, a series of calculations are performed with the system occupying an ever greater fraction of the total grid box at each step. The boundary points in each new grid are internal points from its predecessor, as shown in Figure 11.28. Focusing enables better estimates of the potential values at the boundary to be obtained. The results can also depend upon the orientation of the solute(s) within the grid. The error associated with this can be reduced by performing a series of calculations on randomly translated and rotated copies of the system and then averaging the results.

11.10.5 Applications of Finite Difference Poisson–Boltzmann Calculations

A wide variety of problems have been studied using the finite difference Poisson–Boltzmann (FDPB) method. In addition to the numerical values that the method can provide, significant insights can often be gleaned by graphical examination of the electrostatic potential around the molecule [Honig and Nicholls 1995]. It is often found that the electrostatic potential around a protein calculated using the FDPB method differs significantly from that obtained with a uniform dielectric model. The location of the charged and polar groups in the protein and the shape of the molecule (which determines the shape of the boundary between the regions of high and low dielectric) significantly influence the shape of the potential This

can be seen in Figure 11.29 (colour plate section), which shows the electrostatic potential around the enzyme trypsin. The activity of this enzyme is regulated *in vivo* by trypsin inhibitor, which is a smaller protein that binds strongly to trypsin. However, both trypsin and trypsin inhibitor have net positive charges. How then do the two molecules associate? If the electrostatic potential around trypsin is calculated assuming a uniform dielectric constant of 80 then, as expected, the potential is positive everywhere. However, when the effects of the dielectric boundary are included then a region of negative electrostatic potential appears in the region where the inhibitor binds. A second example is provided by the enzyme Cu-Zn superoxide dismutase, which catalyses the conversion of O_2^- radicals to O_2 and H_2O_2. The rate constant for the reaction is high, being only about one order of magnitude smaller than the expected collision rate of the substrate with the entire enzyme. However, the active site constitutes a very small proportion of the surface, and uniform collisions of the substrate over the protein surface would not explain the observed kinetics. It has therefore been suggested that the substrate is 'steered' into the active site by the electric field of the protein. Figure 11.30 (colour plate section) shows the electrostatic potential around the enzyme (which is a dimer) with the active sites at the top left and bottom right. As can be seen, a concentrated region of positive electrostatic potential extends from the active site into solution [Klapper *et al.* 1986]. The cleft-like nature of the protein around the active site enhances the positive electrostatic potential by focusing electric field lines out into the solvent.

The finite difference Poisson–Boltzmann method can be used to calculate the electrostatic contribution to various processes such as solvation and the formation of intermolecular complexes. The electrostatic component of the solvation free energy equals the change in electrostatic energy for transfer from vacuum to the solvent where the electrostatic energy of a charge q_i in a potential ϕ_i equals $q_i\phi_i$. The solvation free energy is determined by performing two separate calculations using the same grids and the same solute dielectric but exterior dielectrics of 80 (when the solvent is water) and 1 (for the vacuum). Then ΔG_{elec} is given by:

$$\Delta G_{elec} = \tfrac{1}{2}\sum_i q_i(\phi_i^{80} - \phi_i^{1}) \qquad (11.81)$$

The summation in Equation (11.81) is over all charges in the solute.

The change is free energy for the association of two molecules (assumed to have the same internal dielectric constant, ε_m) can be calculated using the finite difference Poisson–Boltzmann method. This problem is usefully discussed by dividing the free energy of association into a series of steps, as shown in Figure 11.31 [Gilson and Honig 1988]. First, the free energy associated with the transfer of the two isolated species from the solvent (dielectric constant ε_s) to a medium of dielectric ε_m is calculated in the same manner as for the solvation free energy (but here the transfer is from the solvent to a medium of dielectric ε_m, not to a vacuum). The free energy to bring the two molecules together is calculated using Coulomb's law in a medium of dielectric ε_m. Finally, the energy to transfer the complex from the medium of dielectric ε_m to the solvent is determined. The same procedure can be applied to other processes, such as the calculation of the free energy difference between two conformations in solution.

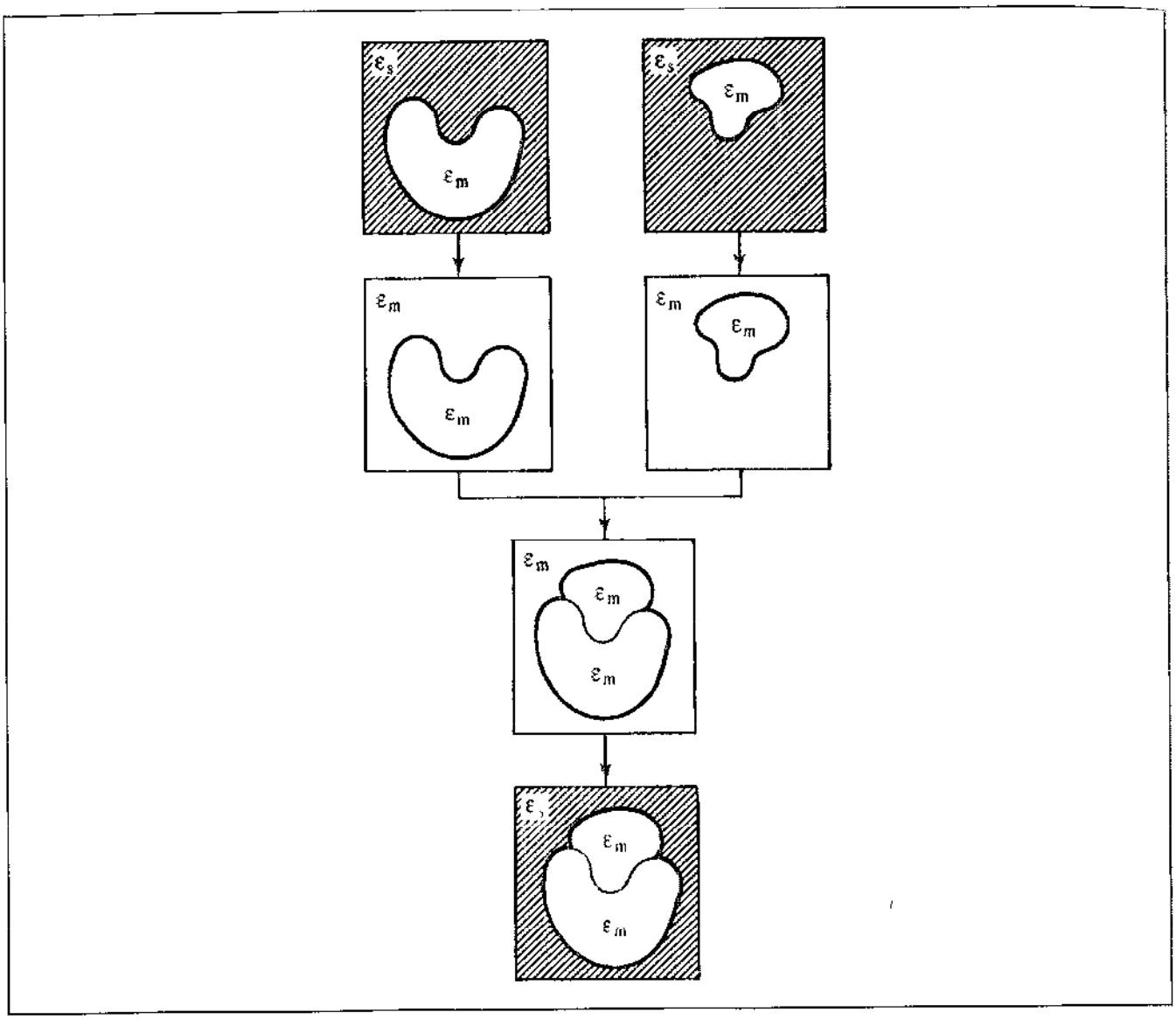


Fig. 11.31. Calculation of the electrostatic free energy of association of two molecules (Figure adapted from Gilson M K and B Honig 1988. Calculation of the Total Electrostatic Energy of a Macromolecular System: Solvation Energies, Binding Energies and Conformational Analysis Proteins Structure, Function and Genetics *4·7–18.)*

11.11 Non-electrostatic Contributions to the Solvation Free Energy

So far, we have only considered the electrostatic contribution to the free energy of solvation. Important though this is, there are some additional factors that contribute to the overall free energy of solvation, as shown in Equation (11.46). These extra contributions can be especially significant for solutes that are neither charged nor highly polar. The cavity and van der Waals terms are often combined and represented using an equation of the following form:

$$\Delta G_{\text{cav}} + \Delta G_{\text{vdw}} = \gamma A + b \tag{11.82}$$

where A is the total solvent accessible area and γ and b are constants This linear dependence upon the area A can be explained as follows. The cavity term equals the work to create the cavity against the solvent pressure and the entropy penalty associated with the reorganisation of solvent molecules around the solute. The solvent molecules most affected by this reorganisation are those in the first solvation shell. The number of solvent molecules in the first solvation shell is approximately proportional to the accessible surface area of the solute The solute–solvent van der Waals interaction energy would also be expected to be dependent primarily upon the number of solvent molecules in the first solvent shell, as van der Waals interactions fall off rapidly with distance. Hence both the cavity and van der Waals terms should be approximately proportional to the solvent accessible surface area The parameters γ and b in Equation (11.82) are usually taken from experimentally determined free energies for the transfer of alkanes from vacuum to water. The parameter b is commonly set to zero, making the cavity plus van der Waals terms directly proportional to the solvent accessible surface area. Still's generalised Born/surface area model (GB/SA) uses the generalised Born approach for the electrostatic contribution together with a cavity and van der Waals surface area term in which the surface area is calculated using a variant of the Wodak and Janin algorithm, the constant γ having the value 7.2 cal/(mol Å^2) [Hasel *et al.* 1988]. As we have already stated, analytical first and second derivatives of the surface area with respect to the atomic coordinates can be rapidly determined using the Wodak–Janin method, so enabling the GB/SA method to be incorporated into energy-minimisation and molecular dynamics calculations.

The cavity and van der Waals contributions may also be modelled as separate terms In some implementations an estimate of the cavity term may be obtained using scaled particle theory [Pierotti 1965; Claverie *et al* 1978], which uses an equation of the form:

$$\Delta G_{\text{cav}} = K_0 + K_1 a_{12} + K_2 a_{12}^2 \tag{11 83}$$

The constants K depend upon the volume of the solvent molecule (assumed to be spherical in shape) and the number density of the solvent. a_{12} is the average of the diameters of a solvent molecule and a spherical solute molecule. This equation may be applied to solutes of a more general shape by calculating the contribution of each atom and then scaling this by the fraction of that atom's surface that is actually exposed to the solvent. The dispersion contribution to the solvation free energy can be modelled as a continuous distribution function that is integrated over the cavity surface [Floris and Tomasi 1989].

11.12 Very Simple Solvation Models

Some particularly simple solvation models include all contributions to the solvation free energy (including the electrostatic contribution) in an equation of the following form

$$\Delta G_{\text{sol}} = \sum_i a_i S_i \tag{11 84}$$

where S_i is the exposed solvent accessible surface area of atom i, and the summation is over all atoms in the solute. a_i is a parameter that depends upon the nature of atom i Despite the

obvious assumptions inherent in such an approach, it does have the advantage of providing an extremely rapid way to calculate a solvation contribution. Eisenberg and McLachlan developed such a model to study proteins, with the parameters a_i being derived by considering just five classes of atom (carbon, neutral oxygen and nitrogen, charged oxygen, charged nitrogen, and sulphur) [Eisenberg and McLachlan 1986]. The values themselves were obtained by fitting to experimentally determined free energies of transfer. Eisenberg and McLachlan applied their solvation model to a variety of problems, such as the recognition of misfolded protein structures and ligand binding.

11.13 Modelling Chemical Reactions

It is obviously important to be able to model chemical reactions, as these lie at the heart of chemistry and biochemistry. Most reactions of interest do not take place in the gas phase but in some medium, be it in a solvent, in an enzyme or on the surface of a catalyst. The environment can have a significant impact upon the reaction by speeding it up or slowing it down or even changing the reaction pathway. Good agreement can sometimes be obtained for calculations performed on isolated systems (i.e. in the gas phase), but to model the system properly the environment must be taken into account.

The preferred technique for modelling chemical reactions is usually considered to be quantum mechanics. Unfortunately, if one wishes to represent the whole system explicitly, the large number of atoms that must be considered means that *ab initio* quantum mechanics is rarely practical. Here we will consider three methods that have been used to study chemical reactions involving large systems. One strategy is to use a purely empirical approach. An alternative is to divide the system into two and treat the 'reaction region' using quantum mechanics, with the rest of the system being modelled using molecular mechanics. Third, we shall consider techniques such as the Car–Parrinello method and density functional theory, which, when allied to extremely powerful computers, can enable the entire reacting system to be simulated using quantum mechanics.

11.13.1 Empirical Approaches to Simulating Reactions

Despite the often-held belief that reactions can only be studied using quantum mechanics, this is by no means the case. Many research groups have developed force field models for studying reactions, which can provide very satisfactory results. Such force fields are used to estimate the activation energies of possible transition states to explain and to predict the stereo- and regioselectivity of the reaction. The force field model is usually derived by extending an existing force field to enable the structures and relative energies of transition structures to be determined.

Here we will illustrate the method using a single example. The aldol reaction between an enol boronate and an aldehyde can lead to four possible stereoisomers (Figure 11.32). Many of these reactions proceed with a high degree of diastereoselectivity (i.e. *syn* : *anti*) and/or enantioselectivity (*syn*-I : *syn*-II and *anti*-I : *anti*-II). Bernardi, Capelli, Gennari,

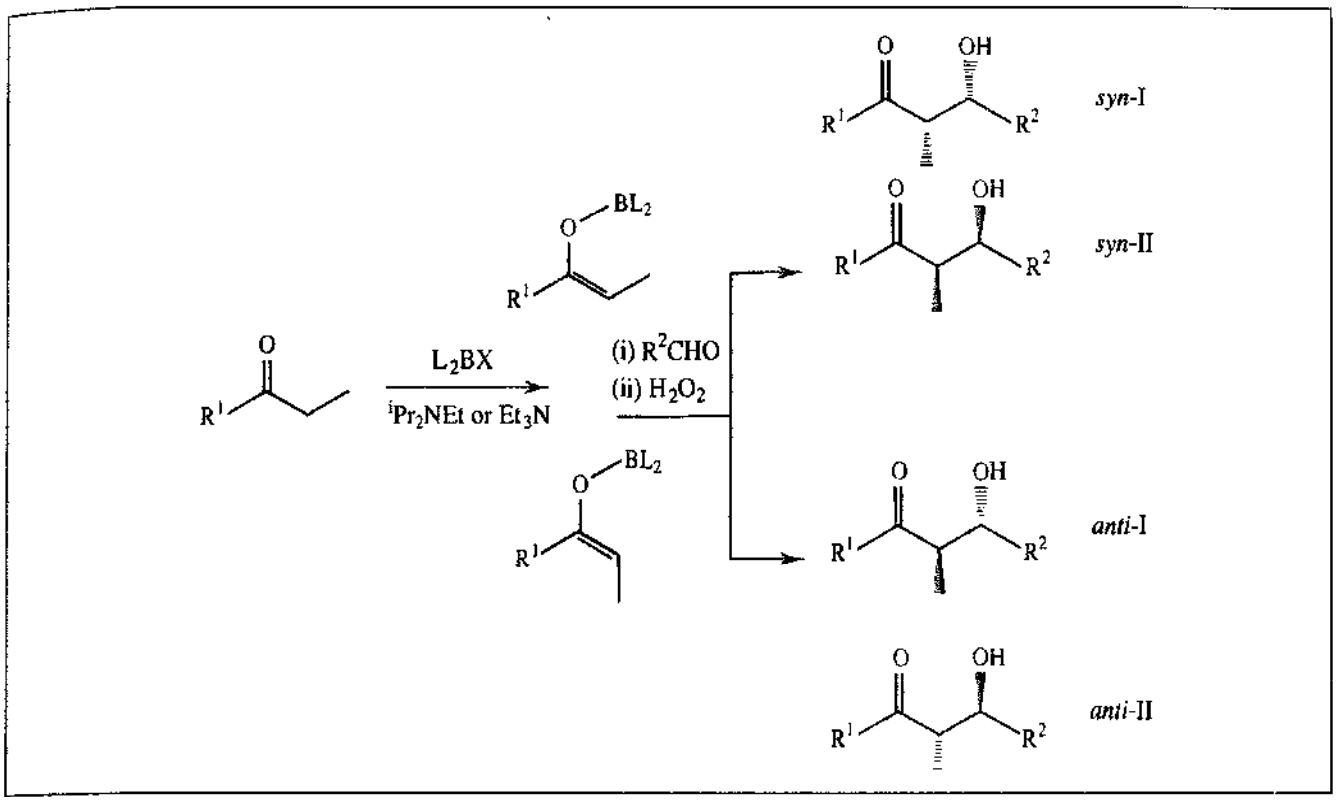


Fig. 11.32 The aldol reaction between an enol boronate and an aldehyde leads to four possible stereoisomers.

Goodman and Paterson studied this reaction using a force field based on MM2 [Bernardi *et al.* 1990]. The force field was parametrised to reproduce the geometries and relative energies of the chair and twist-boat transition structures with unsubstituted reactants, previously determined using *ab initio* methods (see Figure 11.33). It was assumed that the stereoselectivity was determined by the relative energies of the various possible transition structures (i.e. the reaction is assumed to be kinetically controlled).

The force field was then used to predict the results for the addition of the *E* and *Z* isomers of the enol boronate of butanone (R^1 = Me) to ethanol (R^2 = Me). The relevant transition structures are shown in Figure 11.34. A Boltzmann distribution, calculated at the temperature of the reaction (−78°C), predicted that the *Z* isomer would show almost complete *syn* selectivity (*syn* : *anti* = 99 : 1) and that the *E* isomer would be selective for the *anti* product (*anti* : *syn* = 86 : 14). These results were in good agreement with the experimental

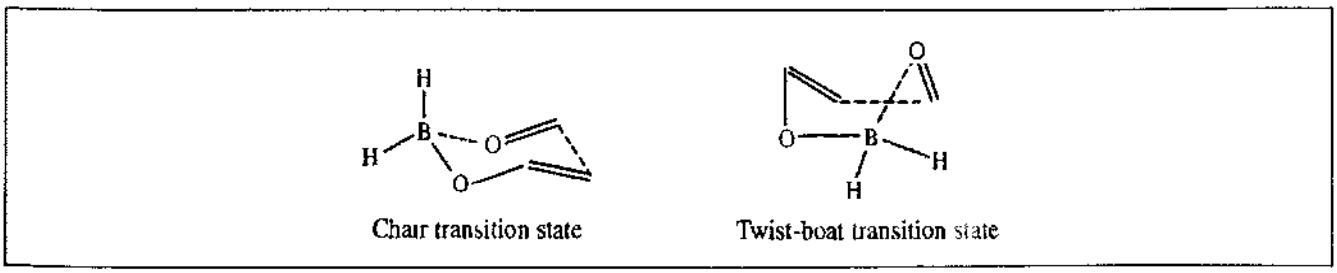


Fig. 11.33 Transition structures for the enol boronoate/aldehyde reaction

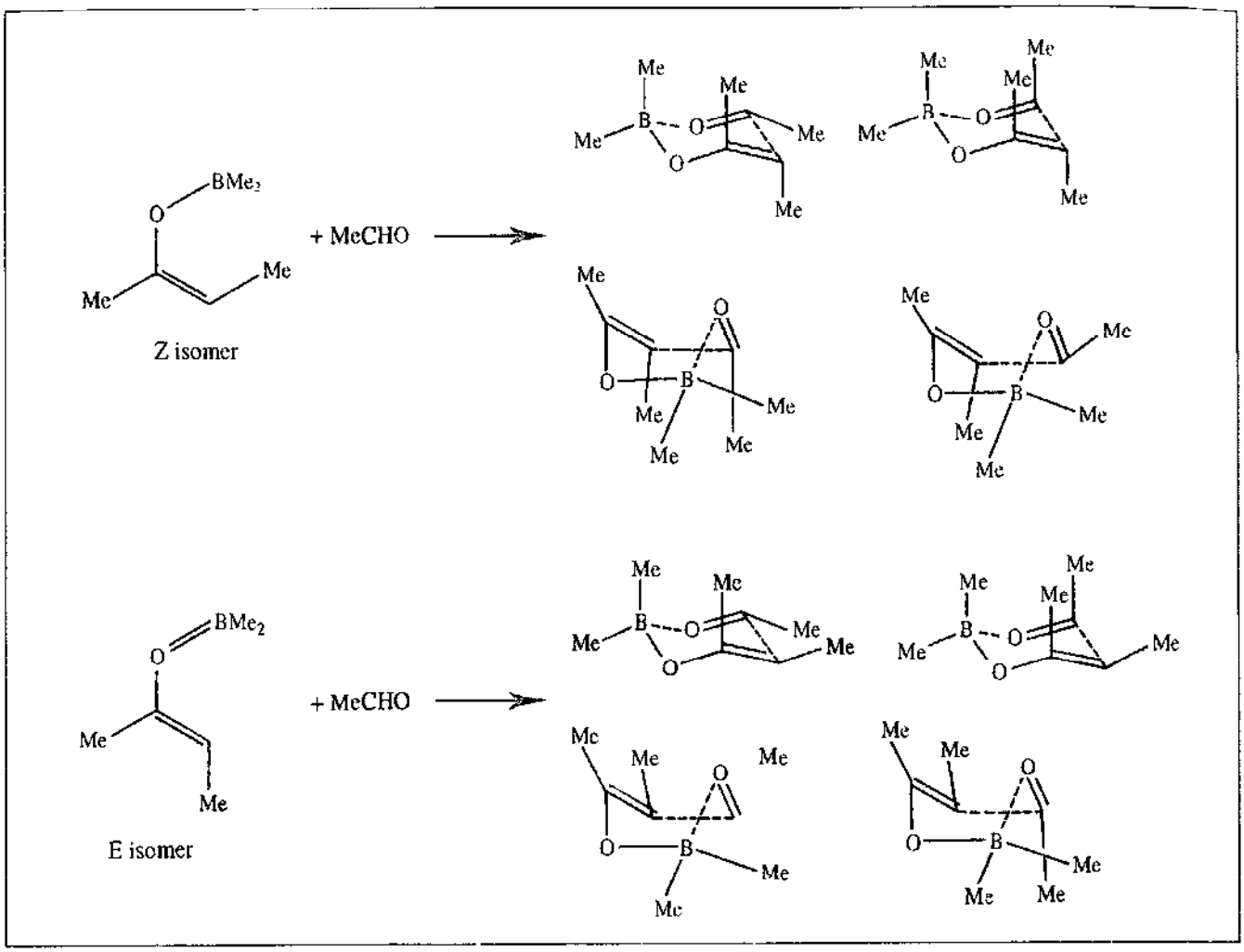


Fig 11 34 Transition states for aldol reaction between butanone and ethanol

observations. The major product in each case was obtained from a chair-like transition structure, but the reduced fraction of the *anti* product for the *E*-isomer was due to a significant contribution from the boat pathway, which leads to the *syn* product.

11.13.2 The Potential of Mean Force of a Reaction

A complete description of a chemical reaction needs to take account of solvent effects. The most realistic way to achieve this is by including explicit solvent molecules. A classic example of how to tackle this problem is Jorgensen's study of the nucleophilic attack of the chloride anion on methyl chloride [Chandrasekhar *et al.* 1985; Chandrasekhar and Jorgensen 1985]. This reaction proceeds via the S_N2 reaction, in which the chloride anion approaches along the carbon–chlorine bond of methyl chloride to give a five-coordinate transition state, which then collapses to give the products. We considered some aspects of the energy surface for this system in Section 5.9, though there we were interested only in the energy change for the gas-phase reaction. The aim of Jorgensen's calculation was to obtain a potential of mean force for the reaction (i.e the change in the free energy as a function of the reaction coordinate) in a variety of solvents.

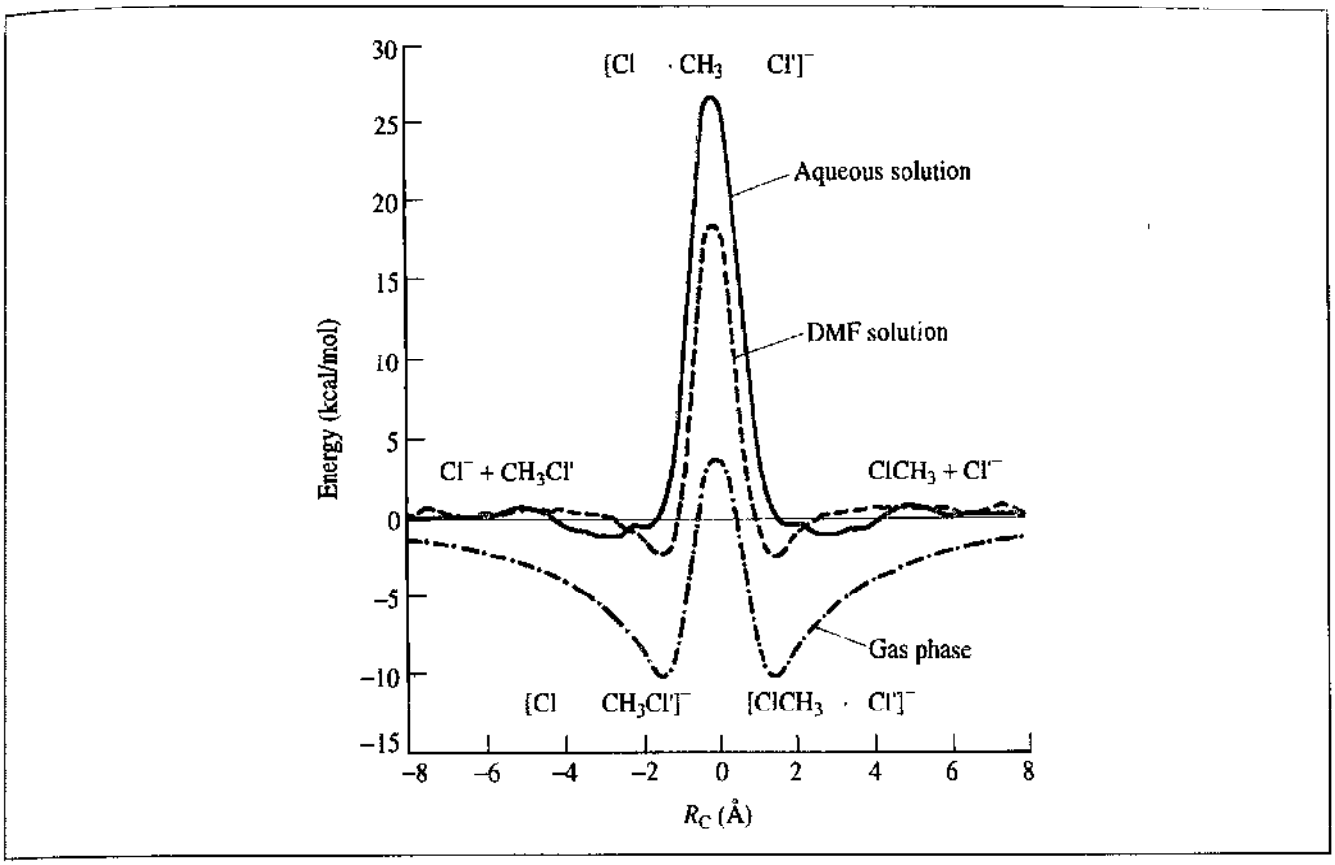


Fig 11 35. Potential of mean force for the Cl^-+MeCl reaction in various solvents. (Figure redrawn from Chandrasekhar J and W L Jorgensen 1985 Energy Profile for a Nonconcerted S_N2 Reaction in Solution Journal of the American Chemical Society **107** *2974–2975.)*

The first step was to determine the quantum mechanical reaction pathway; a series of geometries along the path were determined using the path-following method of Gonzalez and Schlegel [Gonzalez and Schlegel 1988]. The solute–solvent interactions were modelled using Lennard-Jones and electrostatic terms in which the parameters smoothly varied with the reaction coordinate. To perform the Monte Carlo simulations, umbrella sampling was employed to constrain the geometry of the solute to a series of windows along the pathway, and thus calculate the potential of mean force. Preferential sampling methods were used, so that solvent molecules near the solute were sampled more often than solvent molecules further away.

The results are summarised in Figure 11.35, which shows how the potential of mean force varies for the reaction in the gas phase, in water and in dimethyl formamide (DMF) The results exhibit a number of interesting features. In the gas phase an ion-dipole complex forms, giving a minimum in the free energy profile. There is then an activation barrier of approximately 13.9 kcal/mol to reach the pentagonal transition state. In aqueous solution, no ion-dipole minimum is observed. This is because any favourable contribution due to the formation of the ion-dipole is compensated for by the energy lost in the desolvation of the chloride ion. There is then a large activation free energy barrier of approximately 26.3 kcal/mol from the ion-dipole pair to the transition state. This barrier is much larger than in the gas phase because of the poorer solvation of the transition state relative to the

ion–dipole complex. In DMF (a solvent with smaller anion-solvating ability), the ion–dipole complex is at a minimum in the free energy as less energy is required to desolvate the chloride anion in this solvent.

11.13.3 Combined Quantum Mechanical/Molecular Mechanical Approaches

One approach to the simulation of chemical reactions in solution is to use a combination of quantum mechanics and molecular mechanics. The 'reacting' parts of the system are treated quantum mechanically, with the remainder being modelled using the force field. The total energy E_{TOT} for the system can be written:

$$E_{TOT} = E_{QM} + E_{MM} + E_{QM/MM} \tag{11.85}$$

where E_{QM} is the energy of those parts of the system treated exclusively with quantum mechanics, and E_{MM} is the energy of the purely molecular mechanical parts of the system. $E_{QM/MM}$ is the energy of interaction between the quantum mechanical and molecular mechanical parts of the system. This is described by a Hamiltonian $\mathscr{H}_{QM/MM}$. In some cases, $E_{QM/MM}$ is due entirely to non-bonded interactions between the quantum mechanical and molecular mechanical atoms. An example where this could arise would be if all of the atoms in the reacting species were treated quantum mechanically, with molecular mechanics being used exclusively for the solvent. For example, Cl^- and MeCl could be treated using quantum mechanics and solvent with molecular mechanics. In this case, the Hamiltonian $\mathscr{H}_{QM/MM}$ can be written:

$$\mathscr{H}_{QM/MM} = -\sum_i \sum_M \frac{q_M}{r_{iM}} + \sum_\alpha \sum_M \frac{Z_\alpha q_M}{R_{\alpha M}} + \sum_\alpha \sum_M \left(\frac{A_{\alpha,M}}{R^{12}_{\alpha M}} - \frac{C_{\alpha,M}}{R^6_{\alpha,M}} \right) \tag{11.86}$$

The subscript i in Equation (11.86) refers to a quantum mechanical electron and the subscript α to a quantum mechanical nucleus. The subscript M indicates a molecular mechanical nucleus and q_M is its partial atomic charge. There are thus electrostatic interactions between the electrons of the quantum mechanical region and the molecular mechanical nuclei, electrostatic interactions between quantum mechanical and molecular mechanical nuclei, and van der Waals interactions between the quantum mechanical and molecular mechanical atoms. The second and third terms in Equation (11.86) do not involve electronic coordinates and so can be calculated in a straightforward way (i.e. they are constant for a given nuclear configuration). The first term must be incorporated into the quantum mechanical calculation via one-electron integrals added to the one-electron matrix, H^{core}. These one-electron integrals have the form:

$$\int \phi_\mu(1) \frac{1}{r_{1,M}} \phi_\nu(1)\, d\nu(1) \tag{11.87}$$

In some cases, the quantum mechanical and molecular mechanical regions are in the same molecule and so there are bonds between atoms from each region. The energy $E_{QM/MM}$ must now contain terms that describe this interaction. This can be done by adding a molecular mechanical-like energy which contains bond-stretching, angle-bending

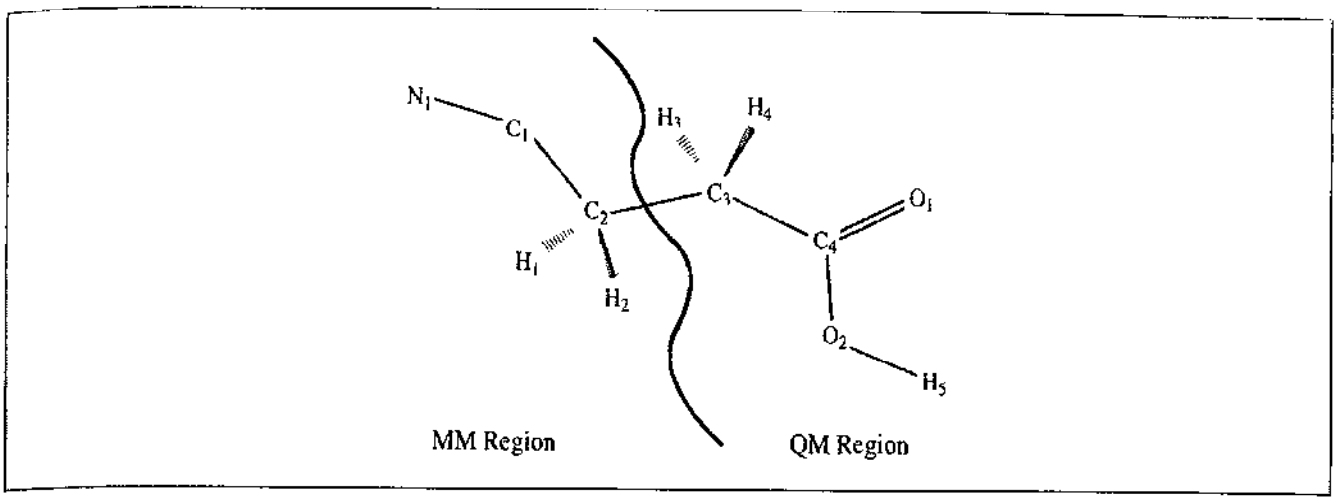


Fig 11 36 The division of a molecule into quantum mechanical and molecular mechanical regions, with the molecular mechanical contributions as indicated.

and torsional terms for atoms from both the quantum mechanical and molecular mechanical sets. This is illustrated in Figure 11.36, which shows which terms would be included in $E_{QM/MM}$.

Various combined quantum mechanical/molecular mechanical implementations have been described [Warshel and Levitt 1976; Singh and Kollman 1986; Field *et al.* 1990; Maseras and Morokuma 1995]. These implementations differ in the quantum mechanical theory that is used (semi-empirical, *ab initio*, valence bond or density functional theory), the molecular mechanical model, and the way in which the solvent is represented (either explicitly or using a simplified model). Another important difference is the way in which the junction between the QM/MM regions is handled. In particular, one must avoid half-filled orbitals for the quantum mechanical region, which would arise if the connection bonds were simply truncated. Two general approaches to this problem have been developed. In one approach, a hybrid sp^2 orbital containing one electron is established along the QM–MM [Warshel and Levitt 1976]. The alternative method simply includes 'link' atoms (typically hydrogen atoms), which ensure that valency is maintained. Interactions between these link atoms and the molecular mechanical region is reduced in magnitude or completely neglected. Comparisons of the two approaches on simple model systems suggest that neither is systematically better than the other provided care is taken in the formulation [Reuter *et al.* 2000].

Combined quantum mechanical/molecular mechanical methods are not, of course, restricted to studies of reactions but can also be used to study association processes and conformational transitions. Most implementations use a two-zone model as described above, but Morokuma and colleagues have described a multilayered approach called ONIOM [Svensson *et al.* 1996]. ONIOM is a particularly apt name given that a typical calculation is constructed from a series of layers For example, a three-layer ONIOM calculation on the Diels–Alder reaction involved an inner core treated with the B3LYP density functional approach, the intermediate layer with a Hartree–Fock level of theory and the outer layer with MM3. A particular feature of ONIOM and its related methods is that they provide rigorous gradients and second derivatives, so enabling properties such as vibrational frequencies to be calculated [Dapprich *et al.* 1999]

The objective of many of the research groups involved in the development of combined quantum mechanical/molecular mechanical models has been the simulation of enzyme reactions. Warshel has reported studies in which the reaction centre in treated using a valence bond model [Warshel 1991; Åqvist and Warshel 1993]. The first part of his strategy is a calibration of the valence bond model for the reference reaction in solution. This model is then used to simulate the enzyme reaction using molecular dynamics and free energy perturbation methods, with solvent effects being treated using the Langevin dipole model. Warshel has extensively studied a wide range of enzyme systems. One example is his study of the enzyme carbonic anhydrase, which is a zinc-containing enzyme that catalyses the reversible hydration of carbon dioxide according to the following mechanism:

$$\text{E-H}_2\text{O} \rightleftharpoons \text{H}^+\text{-E-OH}^- \rightleftharpoons \text{E-OH}^- \tag{11.88}$$

$$\text{E-OH}^- + \text{CO}_2 \rightleftharpoons \text{E-HCO}_3 \rightleftharpoons \text{E-H}_2\text{O} + \text{HCO}_3 \tag{11.89}$$

where E represents the enzyme. In the first step, a bound water molecule is proteolised and the protein is transferred to solution. This is the rate-determining step of the reaction. In the second step, CO_2 is converted to HCO_3^-. Åqvist, Fothergill and Warshel have examined both steps; here we concentrate on their results for the nucleophilic attack on the carbon dioxide (Equation (11.89)) [Åqvist *et al.* 1993]. Simulations of the hydration reaction of CO_2 in water were performed to find valence bond parameters which reproduced the experimentally observed value. This valence bond model was then used to simulate the same reaction in the enzyme The resulting free energy profiles for the reference reaction and the enzyme reaction are shown in Figure 11.37. These results suggest that the enzyme markedly lowers the activation barrier for the reaction and that the reaction is less exothermic in the enzyme than in water ($\Delta G^{\ddagger} = 6.3$ kcal/mol versus 11.9 kcal/mol; $\Delta G^0 = -4.8$ kcal/mol versus -10.5 kcal/mol). The experimental values were estimated to be $\Delta G^{\ddagger} = 7.1$ kcal/mol and $\Delta G^0 = -4.1$ kcal/mol. The enzyme thus speeds up the reaction by a factor of about 10^3 compared with aqueous solution. The transition-state geometry obtained from the simulation was found to be similar to geometries obtained using gas-phase *ab initio* calculations.

11.13.4 *Ab Initio* Molecular Dynamics and the Car–Parrinello Method

The 'ideal' way to simulate reactions (and indeed many other processes where we might wish to derive properties dependent upon the electronic distribution) would of course be to use a fully quantum mechanical approach.

In principle, it would be relatively straightforward to use a quantum mechanical model to determine the forces required by molecular dynamics or the energies for a Monte Carlo simulation algorithm. Hartree–Fock calculations are normally solved using iterative matrix diagonalisation techniques, as discussed in Chapter 2. Density functional calculations can also be tackled using such methods. However, for systems with many atoms and/or basis functions such calculations can be very time-consuming, and it may also be difficult to achieve convergence. Even with pseudopotentials, the number of plane-wave basis functions that can be required for density functional calculations may be very large, and as the number of occupied orbitals is often considerable, it can be a major task to solve the Kohn–Sham equations and determine the energy of a given configuration of atoms. In

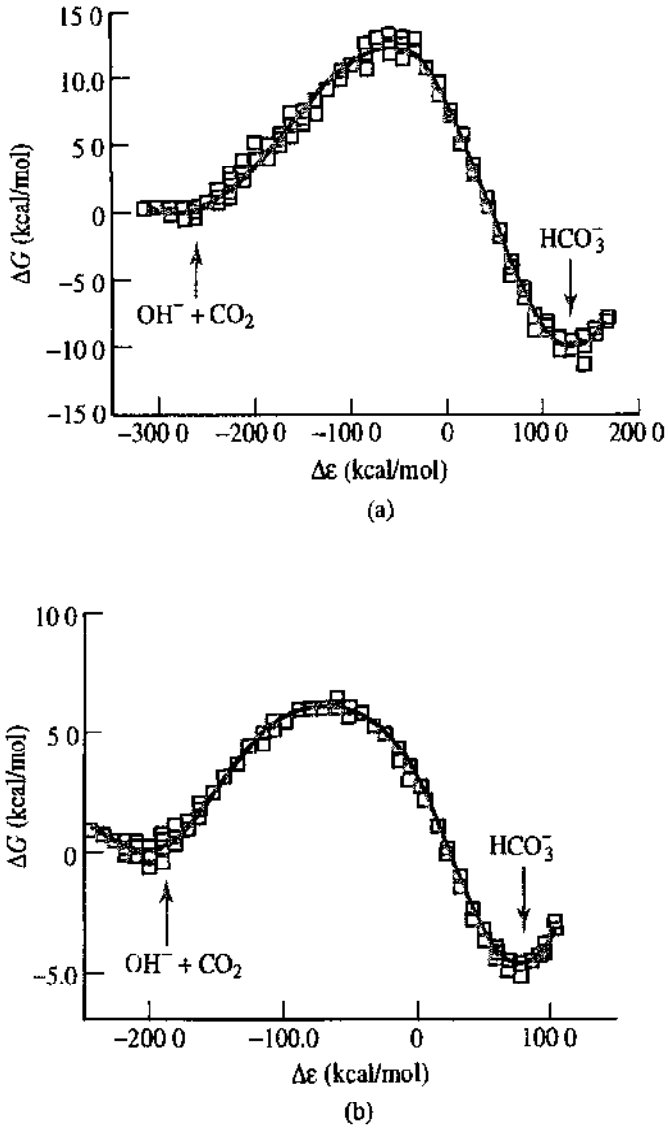


Fig. 11 37 Free energy profile for the nucleophilic attack of water on CO_2 (a) in aqueous solution and (b) in the enzyme carbonic anhydrase (Graphs redrawn from Åqvist J, M Fothergill and A Warshel 1993 Computer Simulation of the CO_2/HCO_3^- Interconversion Step in Human Carbonic Anhydrase I Journal of the American Chemical Society *115 631–635)*

1985, Car and Parrinello described a method that brought together a number of key concepts that we have considered in earlier chapters [Car and Parrinello 1985; Remler and Madden 1990]. They were primarily concerned with the problem of performing *ab initio* simulations involving both the electronic and the nuclear motions ('total energy' simulations or '*ab initio* molecular dynamics'). However, their scheme can be used to perform energy minimisation or simply to determine the basis set coefficients for a fixed atomic configuration.

A key feature of the Car–Parrinello proposal was the use of molecular dynamics and simulated annealing to search for the values of the basis set coefficients that minimise the electronic energy. In this sense, their approach provides an alternative to the traditional matrix diagonalisation methods. In the Car–Parrinello scheme, 'equations of motion' for

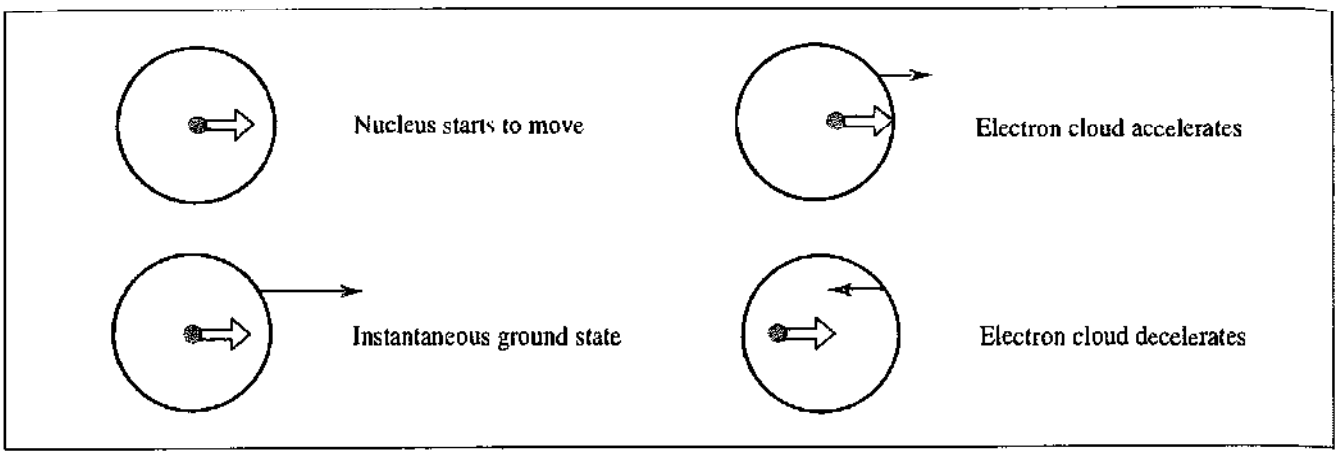


Fig. 11.38: Lag effects in ab initio *molecular dynamics (Figure redrawn from Payne M C, M P Teter, D C Allan, R A Arias and D J Joannopoulos 1992 Iterative Minimisation Techniques for* Ab Initio *Total-Energy Calculations Molecular Dynamics and Conjugate Gradients.* Reviews of Modern Physics *64 1045–1097)*

the coefficients are set up, and then molecular dynamics is used to move the system through the space of the basis set coefficients. Starting from a random set of coefficients (which correspond to a high energy) the system moves downhill on the energy surface, accumulating 'kinetic energy'. Simulated annealing was proposed as a mechanism for preventing the system becoming trapped in a local minimum. The SHAKE algorithm (see Section 7.5) is used to impose constraints on the system to ensure that the orbitals remain orthonormal.

To perform *ab initio* molecular dynamics, Car and Parrinello suggested that the electronic and nuclear dynamics could be performed simultaneously. Somewhat surprisingly, it is found that in the Car–Parrinello scheme it is not necessary for the electronic configuration to be at a minimum in coefficient space for each molecular dynamics time step, even though this gives errors in the forces on the nuclei. It can be shown that the errors in the nuclear forces are cancelled by the associated errors in the electronic motion. One possible explanation for this rather strange (and fortuitous!) result is to consider the motion of an atom with a single occupied molecular orbital. If the nucleus starts to move with a constant velocity, then the orbital will initially lag behind the nucleus. The orbital starts to accelerate until it eventually overtakes the nucleus. Having overtaken the nucleus, the orbital starts to slow down, until the nucleus overtakes the orbital, and so on, as illustrated in Figure 11.38.* An important practical feature of this molecular dynamics approach is that the fictitious masses assigned to the coefficients must be chosen so that the frequencies of electron motion are higher than those of the nuclei to avoid energy exchange. This can have the practical consequence of requiring a smaller time step, which adds to the computational cost.

An alternative to the Car–Parrinello method is the following scheme, which separates the electronic and nuclear motions:

1. Calculate the forces on the nuclei.
2. Move the nuclei according to the molecular dynamics integration scheme.

* It should be pointed out that not all workers in the field are convinced of this particular explanation, appealing though it is

3. Optimise the electronic configuration for the new nuclear configuration.
4. Go to step 1.

This algorithm alternates between the electronic structure problem and the nuclear motion. It turns out that to generate an accurate nuclear trajectory using this decoupled algorithm the electrons must be fully relaxed to the ground state at each iteration, in contrast to the Car-Parrinello approach, where some error is tolerated. This need for very accurate basis set coefficients means that the minimum in the space of the coefficients must be located very accurately, which can be computationally very expensive. However, conjugate gradients minimisation is found to be an effective way to find this minimum, especially if information from previous steps is incorporated [Payne *et al.* 1992]. This reduces the number of minimisation steps required to locate accurately the best set of basis set coefficients.

11.13.5 Examples of *Ab Initio* Molecular Dynamics Simulations

As we have already observed, liquid water is one of the most challenging systems to model due to its almost unique properties such as hydrogen bonding and high dielectric constant. It is therefore not surprising that it was one of the first systems to be considered for a true Car-Parrinello *ab initio* molecular dynamics simulation [Laasonen *et al.* 1993; Sprik *et al.* 1996]. As the calculations were performed using density functional methods the two studies were thus not only designed to actually perform the simulation but also to investigate a variety of DFT models. Specifically, a variety of the gradient-corrected functionals discussed in Section 3.7.3 were considered, together with two different pseudopotential schemes. These latter differed in the number of plane waves needed; in the first study, a so-called supersoft pseudopotential was employed, which required fewer plane waves than the more conventional pseudopotential used in the second study. This was largely driven by the available computational resources; there were some shortcomings with the supersoft pseudopotential, but it did enable the simulation to be run on what would now be considered a very modest computer. It is worth recalling from Section 3.8.6 that the combination of plane waves and density functional theory provides a very natural and appealing way to tackle periodic systems. However, there we were concerned with naturally periodic systems, whereas for *ab initio* molecular dynamics it is the use of periodic boundary conditions which gives rise to the periodicity.

The simulations were restricted to a relatively small number of molecules (32) under periodic boundary conditions. Only rather short simulations were possible (of the order of 5 ps), but it was still possible to determine many of the standard structural properties such as the radial distribution function together with properties such as the vibrational spectra, which provide information on hydrogen bonding within the system. More recent simulations using a larger number of molecules and for a longer time focused on the molecular charge distribution and polarisation effects [Silvestrelli and Parrinello 1999]. A broad distribution was found for the dipole moment around an average value of 3.0 D. This may have important implications for empirical potentials, which are often parametrised to reproduce a lower value around 2.6 D. In addition, the anisotropy of the electronic charge distribution in the water molecule was found to be reduced in the liquid

Building upon the earlier simulations, a subsequent study investigated systems containing hydronium and hydroxyl ions in water [Tuckerman *et al.* 1995a, b]. Protons show exceptionally high mobilities that are far in excess of the values expected from a straightforward diffusion process. A model that accounts for this (the *Grotthuss mechanism*) involves the proton jumping from the oxygen atom of one water molecule to another. This simple picture works very well for proton conduction in ice, but the situation is more complicated for the liquid species. The simulation of a single hydronium ion (H_3O^+) in water showed that for about 60% of the time the proton is associated with a single water molecule, with the three protons making hydrogen bonds to three neighbouring molecules, giving an $H_9O_4^+$ complex. For the remaining 40% of the time the proton could not be assigned to a unique oxygen atom but was shared between the oxygen atoms of two water molecules, to give an $H_5O_2^+$ structure. Close examination of these two structures indicated that they were, in fact, part of the same fluctuating complex. Much less experimental information is available about the OH^- ion, which the simulation suggests is coordinated to four water molecules, each pointing one OH bond towards it. This $H_9O_5^-$ species remains intact for about 2–3 ps before one of the hydrogen bonds breaks, giving a transient tetrahedral H_7O_4 complex.

Liquid hydrogen fluoride is another fluid of interest due to its strong hydrogen-bonding potential Experimental data suggest the existence of chain-like structures, each containing between six and eight HF molecules held together by hydrogen bonds. In the liquid these chains adopt a zig-zag conformation and are significantly entangled. In addition, there is the possibility of branched structures forming, but the relative importance of these is a matter of debate The structure of the liquid is very sensitive to the nature of the potential model. The *ab initio* molecular dynamics simulations used a density functional approach, and it was necessary to use a gradient-corrected functional in order to describe the system correctly. The simulation contained 54 molecules, with the production phase lasting 0.8 ps [Rothlisberger and Parrinello 1997]. Although the data from the simulation were rather noisy due to the short simulation time, a number of features were apparent. For example, a small degree of branching was observed, with a difference between the likelihood of branching at the hydrogen (1%) and fluorine atoms (6%).

Ab initio molecular dynamics has been applied to many 'materials science' problems. One interesting early application was the *ab initio* molecular dynamics simulation of the reaction between a chlorine molecule and a silicon surface [Stich *et al.* 1994]. This reaction is particularly important in silicon chip manufacture, where the dissociative chemisorption of chlorine (and other halogens) is widely used for processes such as dry etching and surface cleaning. A series of simulations was performed, in each of which a chlorine molecule was 'fired' towards the silicon surface. The subsequent motion and reaction was then determined using the *ab initio* molecular dynamics approach based upon conjugate gradients minimisation The motions of the nuclei were determined using the Verlet algorithm with a time step of approximately 0.5 fs, and each simulation was performed for a total time of between 200 and 400 fs.

The silicon surface contains chains of atoms that are formally bonded to just three other atoms. These atoms compensate for the lack of a full valence complement of bonds by π

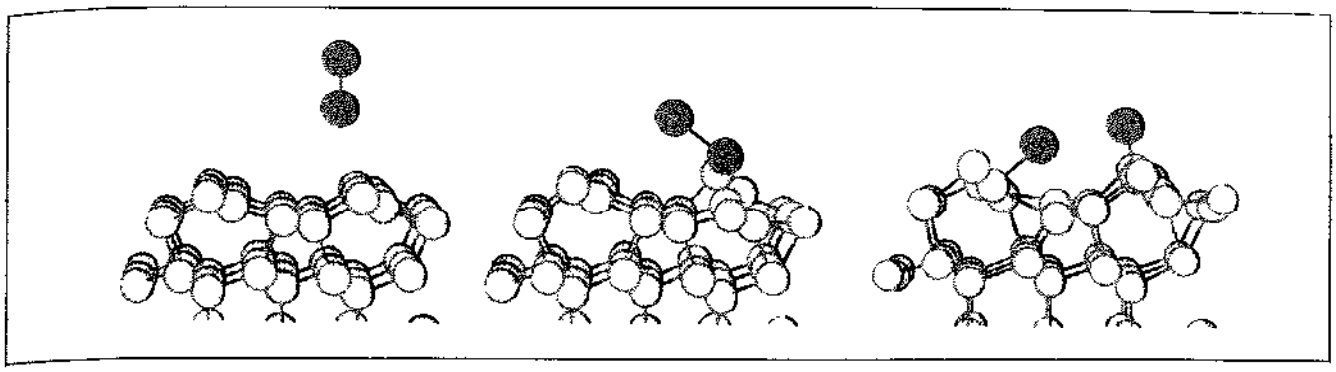


Fig 11.39 Structural changes observed during the reaction of a chlorine molecule with a silicon surface. (Figure redrawn from Stich I, A De Vita, M C Payne, M J Gillan and L J Clarke 1994 Surface Dissociation from First Principles: Dynamics and Chemistry Physical Review **B49**:8076–8085)

bonding along the chains (Figure 11.39). These π-bonded chains represent regions of high electron density, with the valleys between the chains being of relatively low density. Despite this difference in electron density, the chlorine molecule dissociated when it was directed towards either of these two regions. Bonds form between the chlorine atoms and π-bonded silicon atoms, which in turn causes a change in the local hydridisation from sp^2 to sp^3. This then leads to a large local deformation, which lifts the silicon atoms involved above the π-bonded chains. Many other processes of significant commercial interest are now within the scope of the *ab initio* simulation technique, a more recent example being the Ziegler–Natta catalysed polymerisation of ethylene [Boero *et al.* 1999].

A somewhat more unusual illustration of the use of *ab initio* molecular dynamics was of the effect of a knot on the breaking strength of a polymer strand [Saitta *et al.* 1999]. A simple linear alkane formed the basis for the work; the initial calculations involved stretching *n*-decane until one of the bonds broke, to give two radicals. An analogous calculation involving a polyethylene chain with a trefoil knot was then performed. In this case the chain broke at the entrance to the knot (Figure 11.40). Of some interest is the fact that the presence of the knot significantly weakens the strand, as measured by the strain energy per C–C bond in the chain (12.7 kcal/mol for the knotted strand and 16.2 kcal/mol for the linear unknotted case).

Our small selection of examples has tended to concentrate on those which involve the making or breaking of bonds, but this is not of course a requirement for using *ab initio* molecular dynamics. Systems of particular interest are those for which it is difficult to generate empirical force-field models. One such example is the study by de Wijs and colleagues of the viscosity of liquid iron under the conditions believed to exist at the Earth's core [de Wijs *et al.* 1998]. The Earth's magnetic field is believed to arise from the convection of this liquid, an understanding of which is clearly dependent upon knowledge of the viscosity of the medium Estimates of this viscosity vary over many orders of magnitude; for obvious reasons, it is unlikely that this uncertainty will be resolved by experimental measurements. Two regions were of particular interest: the boundary between the solid inner core and the molten outer core, and the boundary between the core and the mantle. The temperatures of these two regions are somewhat uncertain; for the inner core boundary a temperature of 6000 K was assumed and for the core–mantle boundary two temperatures (4300 K and

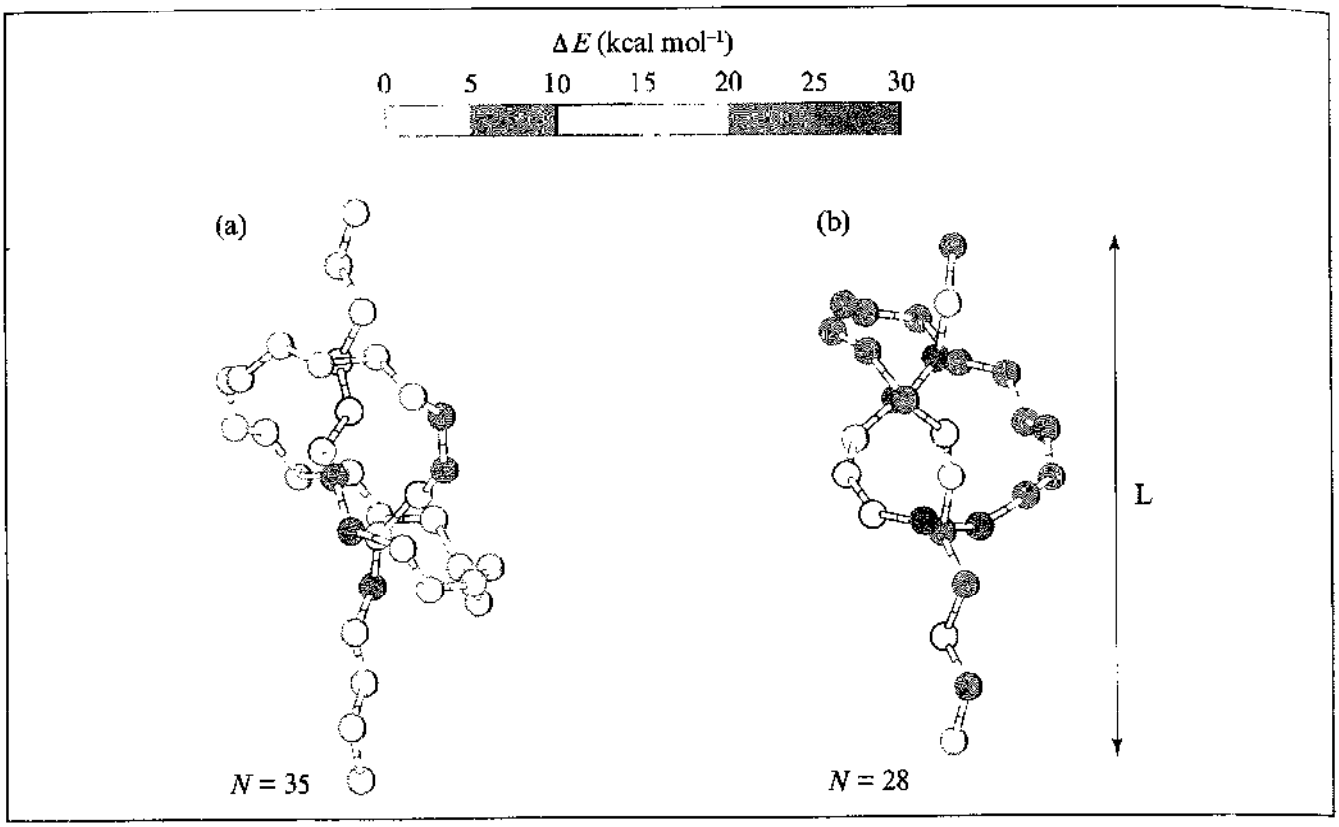


Fig 11 40 Distribution of strain energy is two knotted polymer chains containing 35 (left) and 28 (right) carbon atoms The strain energy is localised and most of the bonds immediately outside the entrance point to the knot (Figure redrawn from Saitta A M, P D Sooper, E Wasserman and M L Klein 1999 Influence of a knot on the strenght of a polymer strand Nature **399** *46–48)*

3500 K) were investigated. From an equation of state for iron the densities at these temperatures could be predicted to enable the simulations to be performed. A periodic system containing 64 atoms was used and the simulation run for 2 ps after equilibration. The calculated pressure agreed within 10% with the 'experimental' values (330 GPa at the inner core boundary and 135 GPa at the core–mantle boundary). Additional parameters could also be calculated, including the viscosity, the values for which were at the low end of previous suggestions.

11.14 Modelling Solid-state Defects

Materials that contain defects and impurities can exhibit some of the most scientifically interesting and economically important phenomena known. The nature of disorder in solids is a vast subject and so our discussion will necessarily be limited. The smallest degree of disorder that can be introduced into a perfect crystal is a *point defect*. Three common types of point defect are vacancies, interstitials and substitutionals. *Vacancies* form when an atom is missing from its expected lattice site. A common example is the Schottky defect, which is typically formed when one cation and one anion are removed from the bulk and placed on the surface. Schottky defects are common in the alkali halides *Interstitials* are due to the presence of an atom in a location that is usually unoccupied. A

Frenkel defect arises when an ion (usually the smaller cation) is removed from its regular site and is placed in an interstitial position. Frenkel defects are common when there is a significant difference in size between the cation and the anion (such as AgBr). *Substitutionals* occur when a foreign ion occupies a regular lattice site. The presence of additional atoms may be due either to accidental impurities or to deliberate doping. These impurities can occupy interstitial sites or they may substitute for an existing atom. The substitution of one atom for another is often difficult due to the tightly packed nature of most solids, but it can be achieved if the atomic sizes are approximately equal. An *aliovalent* substituent is one which has a different valence state from the host. An example of this is the introduction of magnesium (as Mg^{2+} ions) into NaCl (the term *heterovalent* is also used). An additional consequence of this is the formation of cation vacancies, which neutralise the extra charge of the impurity. Defects must always be present in any crystalline solid above absolute zero, purely on entropic grounds (though the concentration can still be very small) These are known as intrinsic defects. Extrinsic defects, by contrast, arise from the accidental or deliberate incorporation of impurities.

Two point defects may aggregate to give a defect pair (such as when the two vacancies that constitute a Schottky defect come from neighbouring sites). Clusters of defects can also form. These defect clusters may ultimately give rise to a new periodic structure or to an extended defect such as a dislocation. Increasing disorder may alternatively give rise to a random, amorphous solid. As the properties of a material may be dramatically altered by the presence of defects it is obviously of great interest to be able to understand these relationships and ultimately predict them. However, we will restrict our discussion to small concentrations of defects.

The most direct effect of defects on the properties of a material usually derive from the altered ionic conductivity and diffusion properties. So-called superionic conductors are materials which have an ionic conductivity comparable to that of molten salts. This high conductivity is due to the presence of defects, which can be introduced thermally or via the presence of impurities. Diffusion affects important processes such as corrosion and catalysis. The specific heat capacity is also affected; near the melting temperature the heat capacity of a defective material is higher than for the equivalent ideal crystal. This reflects the fact that the creation of defects is enthalpically unfavourable but is more than compensated for by the increase in entropy, so leading to an overall decrease in the free energy.

Energy minimisation, molecular dynamics and Monte Carlo simulations have all been used to study the nature of defects and their influence on the material's properties. Special treatments are required for defects, because they can lead to very long-range perturbations. This is particularly the case when the defect has a net positive or negative charge. The calculation of defect energies using energy minimisation is commonly performed using a two-region strategy, based upon a paper published by Mott and Littleton [Mott and Littleton 1938]. The ions in the inner region are fully and explicitly affected by the presence of the defect, in contrast to the ions in the second region (which extends to infinity). Labelling the inner region as 1 and the outer region as region 2 leads to the following expression for the total energy of the system:

$$E = E_1(\mathbf{x}) + E_{12}(\mathbf{x}, \mathbf{y}) + E_2(\mathbf{y}) \tag{11.90}$$

where E_1 is the energy of region 1 (dependent on the coordinates $\mathbf{x}$ of the ions within region 1), E_2 is the energy of region 2 (dependent on the *displacements* $\mathbf{y}$ of the ions in region 2) and E_{12} is the energy of interaction between the two regions. It is assumed that E_2 is a quadratic function of the displacements, which means that it can be written as follows:

$$E_2(\mathbf{y}) = \tfrac{1}{2}\mathbf{y}^{\mathrm{T}}\mathbf{A}\mathbf{y} \tag{11.91}$$

where $\mathbf{A}$ is a force constant matrix. The harmonic-well assumption for the ions in region 2 is appropriate provided the perturbations are small. In practice, it also requires the bulk lattice to be optimised before the defect calculation is performed. At equilibrium, the derivative of the energy with respect to these coordinates $\mathbf{y}$ is zero, from which we can derive:

$$(\partial E/\partial \mathbf{y})_x = (\partial E_{12}(\mathbf{x},\mathbf{y})/\partial y)_x + \mathbf{A}\cdot\mathbf{y} = 0 \tag{11.92}$$

This can be used to eliminate the energy E_2 from Equation (11.90), giving the following expression for the total energy:

$$E = E_1(\mathbf{x}) + E_{12}(\mathbf{x},\mathbf{y}) - \tfrac{1}{2}(\partial E_{12}(\mathbf{x},\mathbf{y})/\partial \mathbf{y})_x \cdot \mathbf{y} \tag{11.93}$$

In order to determine the energy it would thus seem that it is necessary merely to minimise E with respect to the positions $\mathbf{x}$ and the displacements $\mathbf{y}$. However, a complication arises due to the fact that the displacements in the outer region are themselves a function of the inner-region coordinates. The solution to this problem is to require that the forces on the ions in region 1 are zero, rather than that the energy should be at a minimum (for simple problems the two are synonymous, but in practice there may still be some non-zero forces present when the energy minimum is considered to have been located). An additional requirement is that the ions in region 2 need to be at equilibrium.

Implementation of the two-region method requires calculation of the interaction between the ions in region 1 and region 2. For short-range potentials (e.g. the van der Waals contribution) it is only the inner part of region 2 that contributes significantly to the energy, E, and the forces on ions in region 1. Thus in current practical implementations of the method the outer region is subdivided into two regions, 2a and 2b (Figure 11.41). In region 1, an atomistic representation is used with full relaxation of the ions. Region 2a also contains explicit ions, whereas in region 2b it is assumed that the only effect of the defect is to change the polarisation of the ions. An iterative approach is used to identify the configuration in which the forces on the ions in region 1 are zero and the ions in region 2a are at equilibrium. The displacements of the ions in region 2a are commonly determined using just the electrostatic force from the defect species alone and equals the force due to any interstitial species less the force due to any vacancies (based on

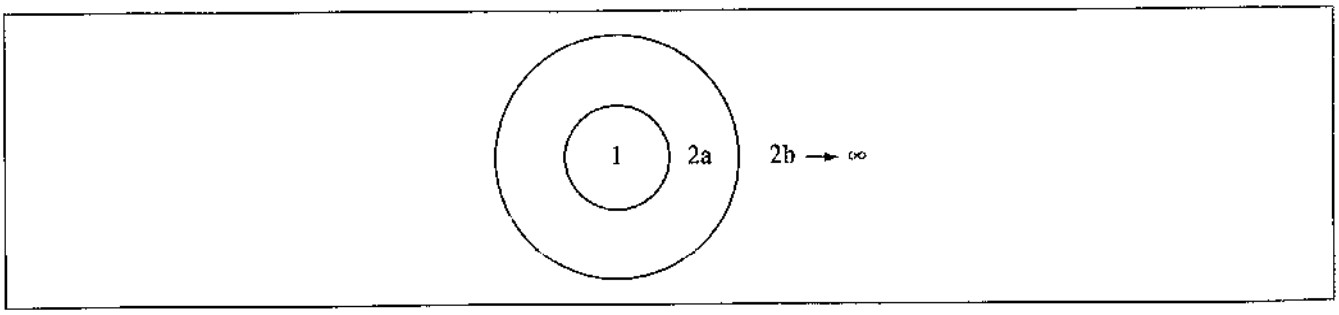


Fig 11.41. Two region scheme used in Mott–Littleton calculations

the position of the original vacancy site). A Newton–Raphson approach is employed, wherein the displacement $\mathbf{y}$ of an ion from its current position is given by:

$$\mathbf{y} = -\mathscr{V}' \cdot \mathscr{V}''^{-1} \tag{11.94}$$

From these calculated displacements, the contributions to the energy from the ions in region 2a can be determined. Finally, it is necessary to determine the contribution from the ions in region 2b. As we have mentioned, these are not included explicitly but are considered to polarise due to the electrostatic field from the total charge on the defect. This contribution is given by the following summation:

$$E_{2b} = -Q \sum_j \frac{q_j(\mathbf{y}_j \cdot \mathbf{R}_j)}{|\mathbf{R}_j|^3} \tag{11.95}$$

where Q is the total effective charge on the defect and q_j is the charge on ion j in region 2b, with $\mathbf{y}_j$ and $\mathbf{R}_j$ being its displacement and equilibrium position. The Mott–Littleton approximation provides a formula for the displacement, leading to the following expression for the energy (for an isotropic medium):

$$E_{2b} = -\frac{Q^2 V_m}{8\pi\varepsilon_0} \sum_j \frac{M_i q_j}{|\mathbf{R}_j|^4} \tag{11.96}$$

where V_m is the unit cell volume and M_i is the Mott–Littleton factor, which is related to the polarisabilities of the ions and the dielectric constant by:

$$M_i = \left(\frac{\alpha_i}{\sum_j \alpha_j}\right)\left(1 - \frac{1}{\varepsilon}\right) \tag{11.97}$$

The summation is over the different types of ion in the unit cell. The summation can be written as an analytical expression, depending upon the lattice structure (the original Mott–Littleton paper considered the alkali halides, which form simple cubic lattices) and evaluated in a manner similar to the Ewald summation; this typically involves a summation over the complete lattice from which the explicit sum for the inner region is subtracted.

The defect energy equals the difference in the total energies for the defective and the perfect lattice, corrected for the intrinsic energy of the defective species (interstitial and/or vacancy) at infinite separation. In a modern Mott–Littleton calculation the inner region may contain up to a few hundred atoms; ideally, a series of calculations is performed with increasing numbers of explicit ions until the defect energy converges. Incorporation of polarisability is usually important and is often handled using a shell model (Section 4.22.2). In addition to the energy of defect formation it is also possible to calculate the associated entropy change. This requires a consideration of the effect of the defect on the lattice phonon spectrum (Section 5.10). One technical point that is important to note is that these calculations are performed at constant volume and should be corrected prior to comparison with the constant pressure values typically obtained by experiment. This is particularly important for phenomena which occur at high temperatures, where the difference between the constant volume and constant pressure results can be significant.

Table 11.1 gives the results of Mott–Littleton calculations on some simple ionic systems [Mackrodt 1982]. By comparing the relative energies of the various types of defect, it is

	Theory (eV)	Experiment (eV)
LiF (Schottky)	2.37	2.34–2.68
NaCl (Schottky)	2.22	2.20–2.75
KBr (Schottky)	2 27	2.37–2 53
RbI (Schottky)	2.16	2 1
MgF_2 (anion Frenkel)	3.12	—
CaF_2 (anion Frenkel)	2.75	2.7
BaF_2 (anion Frenkel)	1.98	1.91
$CaCl_2$ (anion Frenkel)	4.7	—
MgO (Schottky)	7.5	5–7

Table 11.1 Defect energies for various materials Data from [Mackrodt 1982]

possible to predict which types of defect might be expected in a particular material For example, the energies to form Schottky defects in the alkali halides are 1–2 eV lower than to form Frenkel defects. By contrast, the dominant type of defect in the alkaline earth fluorides is the anion Frenkel defect.

An alternative to the Mott–Littleton method is to use a so-called supercell calculation, wherein the defect is located within a lattice that is subjected to periodic boundary conditions. The main difficulty with this approach is that, when a charged defect is present, the Ewald summation that is used to determine the Coulombic contribution diverges. This is dealt with by compensating for the net charge of the cell by a uniform background charge density. In addition, the energies of defect formation must be corrected for the interactions between defects in different cells. Supercells are often considered to be simpler for the calculation of defect entropies (and hence free energies) through the use of lattice statics and lattice dynamics. Full free energy minimisation can be performed on cells containing up to 1000 atoms under conditions of either constant volume or constant pressure. Calculations on such large systems are facilitated by the calculation of analytical derivatives of the vibrational frequencies with respect to all external and internal variables, an example being the study of defects in MgO [Taylor *et al.* 1997].

Defect calculations are traditionally performed using an empirical potential function, but there are some types of problem for which a quantum mechanical model is required, such as when the defect formation is accompanied by a transition to an excited electronic state. The obvious drawback to this is that the quantum mechanical method is computationally more expensive than the empirical potentials that are typically used in Mott–Littleton or supercell calculations. As a consequence, the number of atoms that can be treated quantum mechanically is often limited to the defect and its immediate neighbours. It is then necessary in some way to incorporate the effects of the surrounding region. In the embedded cluster approach this outer region provides a representation of the electrostatic potential due to the surrounding lattice, most easily simulated using point charges placed at the appropriate lattice sites. In more sophisticated approaches, the influence of the defect on the surrounding region can be taken into account in a manner similar to the Mott–Littleton approach [Grimes *et al.* 1989; Pisani 1999].

The most basic data that the Mott–Littleton and supercell methods provide are the energies and entropies of defect formation. Nevertheless, despite the fact that these techniques are essentially static approaches it can also be possible to deduce information on the 'dynamic' processes of diffusion and conductivity. These two processes are related by the Nernst–Einstein relationship:

$$\frac{\sigma}{D} = \frac{Nq^2}{fk_BT} \tag{11.98}$$

where σ is the electrical conductivity, D is the diffusion coefficient, N is the number of particles per unit volume and q is the charge on the mobile species. f is a correlation factor whose value depends upon the underlying migration mechanism. f may deviate from unity if the atomic movements affect the migration of charge and mass in different ways. For example, if charge transport is caused by a vacancy mechanism in which atoms jump into vacant sites then this is effectively a random process. However, after the atom has jumped into the vacancy it is possible for it to jump back to the original site. Mass transport is thus a correlated process. Three different defect migration mechanisms are shown in Figure 11.42. Of these, the vacancy mechanism dominates in most close-packed crystal structures. In the interstitial mechanism, the interstitial atom jumps from one site to another, an example being the diffusion of carbon in iron. The interstitialcy mechanism involves an interstitial atom displacing a lattice atom onto a new interstitial site. An example of this is the motion of silver ions in silver halides.

If the transport is due to discrete jumps of atoms then the diffusion coefficient D is related to the concentration of the jumping species (x), the jumping frequency (ν) and the distance over which the jump occurs (d):

$$D = \tfrac{1}{6}x\nu d^2 \tag{11.99}$$

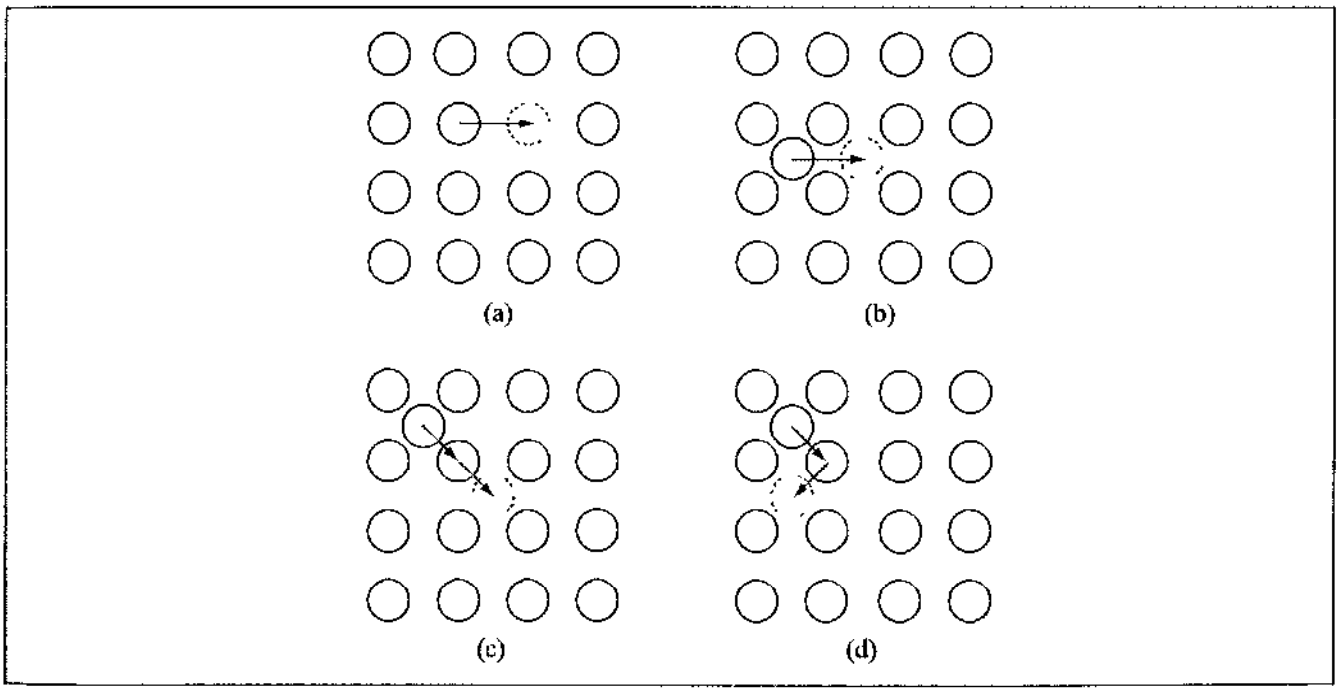


Fig. 11.42 Three different defect migration mechanisms (a) vacancy, (b) interstitial, (c) collinear interstitialcy and (d) non-collinear interstitialcy (Figure redrawn from Chadwick A V and J Corish 1997 Defects and Matter Transport in Solid Materials In NATO ASI Series C **498** (New Trends in Materials Chemistry), *pp 285–318)*

The jump frequency is related in an exponential fashion to the free energy of activation between the ground state and the saddle point:

$$\nu = \nu_0 \exp(-\Delta G_{act}/k_B T) \tag{11.100}$$

ΔG_{act} in turn is composed of the enthalpy and entropy of activation, quantities which can in principle be calculated using other methods, such as those discussed above. The concentration of the jumping species is also predicted to vary in an exponential manner. It is thus expected that transport coefficients will follow an Arrhenius-like behaviour, and plotting the logarithm of the diffusion coefficient or the conductivity against $1/T$ will give a straight line (in fact, in the case of conductivity it is usual to plot $\log(\sigma T)$ against $1/T$ in accordance with the Nernst–Einstein relationship). It is indeed quite common to observe a series of linear regions, each corresponding to different types of defect population. Some typical activation energies are 0.66 eV (cation vacancy migration in NaCl), 0.35 eV (anion vacancy migration in CaF_2) and 2.0 eV (cation vacancy migration in MgO).

Molecular dynamics and Monte Carlo simulations can also be used to investigate systems with defects. In many respects these simulation methods are complementary to static techniques for the study of diffusion and conductivity. As we have discussed, calculation of transport coefficients using static methods is based upon the random jump model. This model is most appropriate when there are relatively high energy barriers involved. These high energy barriers make such systems less appropriate for simulation methods due to sampling difficulties. Simulation methods are most applicable to systems with facile diffusion (i.e. low activation energy barriers to transport), where the random jump model is less valid. Of course, one advantage of molecular dynamics is that diffusion coefficients can be calculated directly The early molecular dynamics simulations concentrated on superionic materials such as $SrCl_2$, CaF_2 and Li_3N; the latter has a layered structure with much higher conductivity parallel to the layers, leading to very different mean squared displacements (Figure 6.10).

11.14.1 Defect Studies of the High-T_c Superconductor $YBa_2Cu_3O_{7-x}$

The discovery of materials which exhibit 'high'-temperature superconductivity (for which Bednorz and Mülller were awarded the Nobel Prize for physics in 1987) led to a frenzy of activity to discover similar materials. This activity was not restricted to purely experimental considerations, as various theories were proposed to explain the reasons for this abnormal behaviour, with particular emphasis on variants on the so-called BCS theory, which involves the formation of pairs of electrons (Cooper pairs). One of the most studied of these high-T_c superconductors is the Y–Ba–Cu–O system, which was the first material found to display a transition temperature at liquid nitrogen temperatures (~ 90 K). The formula of the pertinent material is best written as $YBa_2Cu_3O_{6+x}$, with the superconducting properties being very sensitive to the value of x (rather than arising from aliovalent substitution of the Y^{3+} ions, as occurs in some other materials). High-T_c behaviour does not generally occur when x is less than approximately 0.3. The two related 'parent' molecules are $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$, whose structures are shown in Figure 11.43. These materials contain two different types of copper atom and a variety of oxygen sites. In $YBa_2Cu_3O_6$,

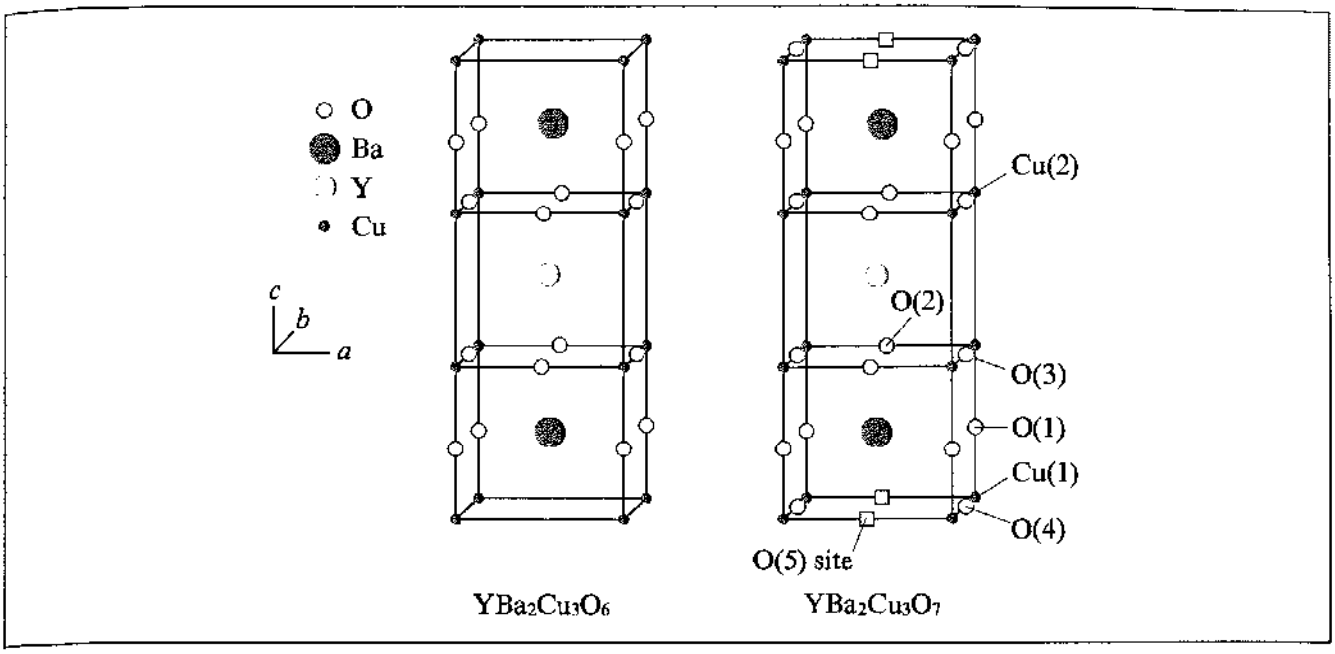


Fig. 11.43 The structures of $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$

the copper in the Cu(2) site is five-fold coordinate to oxygen, whereas in the Cu(1) site the copper is two-fold coordinated. The oxidation states of the copper in these two sites are Cu(II) and Cu(I), respectively. $YBa_2Cu_3O_7$ is derived from $YBa_2Cu_3O_6$ by introducing oxygen into the vacant O(4) sites; the copper in the Cu(1) sites becomes four-fold coordinate with the formation of the so-called CuO_2 basal plane.

These materials have been the subject of considerable computational investigations, involving both static and molecular dynamics methods. For example, static methods have been used to calculate the energies of formation of various vacancy and interstitial defects [Allan and Mackrodt 1994]. From these calculations, it was suggested that the lack of superconductivity for $x < 0.3$ was because the holes resulting from the excess oxygen are trapped in the basal plane adjacent to the O(4) position. In this case, we use the word 'hole' in the physicists' sense to denote a species (in this case Cu) from which an electron has been removed. Thus the addition of oxygen to $YBa_2Cu_3O_6$ causes Cu(I) to be oxidised to Cu(II), with this extra positive charge (i.e. the hole) being trapped by the negative charge of the now negatively charged oxygen ion. Superconductivity is associated with oxidation of the Cu(II) in the CuO_2 planes to Cu(III); this process does not occur until $x > 0.3$.

Many of the high-T_c superconductors also appear to be good oxygen ion conductors at high temperatures, and these effects have also been studied using computational methods. For example, oxygen diffusion in the $YBa_2Cu_3O_{6.9}$ system has been studied using molecular dynamics [Zhang and Catlow 1992]. This particular composition has been shown experimentally to have the fastest oxygen diffusion. The system was set up by removing three O(4) atoms from 32 $YBa_2Cu_3O_7$ units, giving three oxygen vacancies, and then assigning three of the remaining O(4) species a charge of −2 to maintain charge neutrality, for a total of 413 atoms. The simulations were performed at quite a high temperature (higher than most relevant experimental studies) in order to obtain reasonable statistics. It was found that the oxygen diffuses in three directions, but that the diffusion coefficients were larger in the *a*

and b directions than in the c direction (see Figure 11.43). The overall values were in very nice agreement with those extrapolated from experiment data, although some low-temperature experiments had suggested that the b direction diffusion was significantly larger than in the a direction, in contrast to the simulations. The mechanism of oxygen transport was also investigated by analysing the trajectories using molecular graphics. This showed that the oxygen vacancies migrated between the O(4), O(1) and O(5) sites but not to the O(2) and O(3) sites. It was suggested that at lower temperatures the oxygen vacancies mostly followed a O(4)–O(1)–O(4) jump mechanism with a small proportion of O(4)–O(5)–O(4) jumps. At higher temperatures, the vacant O(5) sites were more easily occupied and O(4)–O(5) jumps occurred with a frequency comparable to that of the O(4)–O(1) mechanism.

Appendix 11.1 Calculating Free Energy Differences Using Thermodynamic Integration

If the free energy, A, is a continuous function of λ then we can write:

$$\Delta A = \int_0^1 \frac{\partial A(\lambda)}{\partial \lambda}\, d\lambda \tag{11.101}$$

Now

$$A(\lambda) = -k_B T \ln Q(\lambda) \tag{11.102}$$

Thus

$$\Delta A = -k_B T \int_0^1 \left[\frac{\partial \ln Q(\lambda)}{\partial \lambda}\right] d\lambda = \int_0^1 \frac{-k_B T}{Q(\lambda)} \frac{\partial Q(\lambda)}{\partial \lambda}\, d\lambda \tag{11.103}$$

From the definition of Q (Section 6.1.1):

$$Q_{NVT} = \frac{1}{N!}\frac{1}{h^{3N}} \iint d\mathbf{p}^N\, d\mathbf{r}^N \exp\left[-\frac{\mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{k_B T}\right] \tag{11.104}$$

we can write the following for $\partial Q(\lambda)/\partial\lambda$:

$$\frac{\partial Q(\lambda)}{\partial \lambda} = \frac{1}{N!}\frac{1}{h^{3N}} \iint d\mathbf{p}^N\, d\mathbf{r}^N \frac{\partial}{\partial \lambda} \exp\left[-\frac{\mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{k_B T}\right] \tag{11.105}$$

Applying the chain rule:

$$\frac{\partial Q(\lambda)}{\partial \lambda} = -\frac{1}{N!}\frac{1}{h^{3N}}\frac{1}{k_B T} \iint d\mathbf{p}^N\, d\mathbf{r}^N \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \exp\left[-\frac{\mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{k_B T}\right] \tag{11.106}$$

Substituting back into the expression for $\partial A/\partial\lambda$ gives:

$$\begin{aligned}\frac{\partial A(\lambda)}{\partial \lambda} &= \frac{1}{N!}\frac{1}{h^{3N}}\frac{1}{Q(\lambda)} \iint d\mathbf{p}^N\, d\mathbf{r}^N \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \exp\left[-\frac{\mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{k_B T}\right] \\ &= \iint d\mathbf{p}^N\, d\mathbf{r}^N \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)}{\partial \lambda} \left\{\frac{\exp[-\mathscr{H}(\mathbf{p}^N, \mathbf{r}^N)/k_B T]}{Q(\lambda)}\right\} = \left\langle \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N, \lambda)}{\partial \lambda} \right\rangle_\lambda\end{aligned} \tag{11.107}$$

Thus

$$\Delta A = \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial \mathscr{H}(\mathbf{p}^N, \mathbf{r}^N, \lambda)}{\partial \lambda} \right\rangle_\lambda d\lambda \tag{11.108}$$

Appendix 11.2 Using the Slow Growth Method for Calculating Free Energy Differences

The slow growth expression can be derived from the thermodynamic perturbation expression (Equation (11.7)) if it is written as a Taylor series:

$$\Delta A = -k_B T \sum_{i=0}^{N_{\text{step}}-1} \ln\langle \exp(-[\mathscr{H}(\lambda_{i+1}) - \mathscr{H}(\lambda_i)]/k_B T)\rangle_{NVT} \tag{11.109}$$

$$\Delta A \approx -k_B T \sum_{i=0}^{N_{\text{step}}-1} \ln\langle 1 - [\mathscr{H}(\lambda_{i+1}) - \mathscr{H}(\lambda_i)]/k_B T + \cdots\rangle_{NVT} \tag{11.110}$$

$$\Delta A \approx -k_B T \sum_{i=0}^{N_{\text{step}}-1} \ln\left\{1 - \frac{1}{k_B T}\langle[\mathscr{H}(\lambda_{i+1}) - \mathscr{H}(\lambda_i)]\rangle_{NVT} + \cdots\right\} \tag{11.111}$$

$$\Delta A \approx \sum_{i=0}^{N_{\text{step}}-1} \langle[\mathscr{H}(\lambda_{i+1}) - \mathscr{H}(\lambda_i)]\rangle_{NVT} \tag{11.112}$$

Appendix 11.3 Expansion of Zwanzig Expression for the Free Energy Difference for the Linear Response Method

The starting point is the standard expression for the free energy difference, Equation (11.6):

$$\Delta A = -k_B T \ln\langle \exp[-(\mathscr{H}_Y - \mathscr{H}_X)/k_B T]\rangle_0 \tag{11.113}$$

We expand the exponential:

$$\begin{aligned}\Delta A &= -k_B T \ln\left\langle 1 - \frac{(\mathscr{H}_Y - \mathscr{H}_X)}{k_B T} + \frac{(\mathscr{H}_Y - \mathscr{H}_X)^2}{2(k_B T)^2} - \cdots \right\rangle_0 \\ &= -k_B T \ln\left[1 - \frac{\langle \mathscr{H}_Y - \mathscr{H}_X\rangle_0}{k_B T} + \frac{\langle(\mathscr{H}_Y - \mathscr{H}_X)^2\rangle_0}{2(k_B T)^2} - \cdots\right]\end{aligned} \tag{11.114}$$

Using the series expansion of $\ln(1+x)$ gives:

$$\begin{aligned}\Delta A = -k_B T\Bigg\{ &-\frac{\langle \mathscr{H}_Y - \mathscr{H}_X\rangle_0}{k_B T} + \frac{\langle(\mathscr{H}_Y - \mathscr{H}_X)^2\rangle_0}{2(k_B T)^2} \\ &-\frac{1}{2}\left[\left(\frac{\langle \mathscr{H}_Y - \mathscr{H}_X\rangle_0}{k_B T}\right)^2 - \frac{\langle \mathscr{H}_Y - \mathscr{H}_X\rangle_0\langle(\mathscr{H}_Y - \mathscr{H}_X)^2\rangle_0}{2(k_B T)^3} + \left(\frac{\langle(\mathscr{H}_Y - \mathscr{H}_X)^2\rangle_0}{2(k_B T)^2}\right)^2\right]\Bigg\}\end{aligned} \tag{11.115}$$

This can be rearranged to:

$$\Delta A = \langle \mathscr{H}_Y - \mathscr{H}_X \rangle_0 - \frac{1}{2k_B T} \langle [(\mathscr{H}_Y - \mathscr{H}_X) - \langle \mathscr{H}_Y - \mathscr{H}_X \rangle_0]^2 \rangle_0 + \cdots \tag{11.116}$$

A similar procedure applied to the result from averaging at Y gives:

$$\Delta A = \langle \mathscr{H}_Y - \mathscr{H}_X \rangle_1 + \frac{1}{2k_B T} \langle [(\mathscr{H}_Y - \mathscr{H}_X) - \langle \mathscr{H}_Y - \mathscr{H}_X \rangle_1]^2 \rangle_1 + \cdots \tag{11.117}$$

When Equations (11.116) and (11.117) are added together and we substitute $\Delta\mathscr{H}$ for $\mathscr{H}_Y - \mathscr{H}_X$ then we obtain:

$$\Delta A = \tfrac{1}{2}[\langle \Delta\mathscr{H} \rangle_0 + \langle \Delta\mathscr{H} \rangle_1] - \frac{1}{4k_B T}[\langle (\Delta\mathscr{H} - \langle \Delta\mathscr{H} \rangle_0)^2 \rangle_0 - \langle (\Delta\mathscr{H} - \langle \Delta\mathscr{H} \rangle_1)^2 \rangle_1] + \cdots \tag{11.118}$$

Further Reading

Allan N L and W C Mackrodt 1997. High-T_c Superconductors in Computer Modelling In Catlow C R A (Editor) *Inorganic Crystallography*, pp. 241–268.

Amara P and M J Field 1998. Combined Quantum Mechanical and Molecular Mechanical Potentials. In Schleyer, P v R, N L Allinger , T Clark, J Gasteiger, P A Kollman H F Schaefer III and P R Schreiner (Editors). *The Encyclopedia of Computational Chemistry*. Chichester, John Wiley & Sons.

Beveridge D L and F M DiCapua 1989. Free Energy via Molecular Simulation: A Primer. In van Gunsteren W F and P K Weiner (Editors) *Computer Simulation of Biomolecular Systems* Leiden, ESCOM, pp. 1–26

Catlow C R A 1994. An Introduction to Disorder in Solids. In NATO ASI Series C 418 (*Defects and Disorder in Crystalline and Amorphous Solids*), pp. 1–23

Catlow C R A 1994. Molecular Dynamics Studies of Defects in Solids In NATO ASI Series C 418 (*Defects and Disorder in Crystalline and Amorphous Solids*), pp. 357–373.

Catlow C R A, R G Bell and J D Gale 1994. Computer Modelling as a Technique in Materials Chemistry. *Journal of Materials Chemistry* 4:781–792

Catlow C R A and W C Mackrodt 1982. Theory of Simulation Methods for Lattice and Defect Energy Calculations in Crystals. In *Lecture Notes in Physics* **166** (Comput. Simul. Solids), pp 3–20.

Chadwick A V and J Corish 1997. Defects and Matter Transport in Solid Materials In NATO ASI Series C 498 (*New Trends in Materials Chemistry*), pp. 285–318.

Cramer C J and Truhlar D G 1995. Continuum Solvation Models· Classical and Quantum Mechanical Implementations In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 6. New York, VCH Publishers, pp. 1–72

Gale J 1999. *General Utility Lattice Program Manual*, Imperial College, London.

Gao J 1995. Methods and Applications of Combined Quantum Mechanical and Molecular Mechanical Potentials. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 7 New York, VCH Publishers, pp. 119–185.

Gillan M J 1989. *Ab Initio* Calculation of the Energy and Structure of Solids. *Journal of the Chemical Society Faraday Transactions 2* **85**:521–536.

Gillan M J 1997. The Virtual Matter Laboratory. *Contemporary Physics* **38**:115–130

Harding J H 1997 Defects, Surfaces and Interfaces In Catlow C R A (Editor) *Inorganic Crystallography*, pp. 185–199.

Jorgensen W L 1983. Theoretical Studies of Medium Effects on Conformational Equilibria *Journal of Physical Chemistry* **87**:5304–5314

King P M 1993. Free Energy via Molecular Simulation: A Primer. In van Gunsteren W F, P K Weiner and A J Wilkinson (Editors) *Computer Simulation of Biomolecular Systems* Volume 2 Leiden, ESCOM, pp 267–314.

Kollman P A 1993. Free Energy Calculations: Applications to Chemical and Biochemical Phenomena *Chemical Reviews* **93**:2395–2417

Lybrand T P 1990, Computer Simulation of Biomolecular Systems Using Molecular Dynamics and Free Energy Perturbation Methods. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 1. New York, VCH Publishers, pp 295–320.

Mark A E and van Gunsteren W F 1995 Free Energy Calculations in Drug Design: A Practical Guide In Dean P M, G Jolles and C G Newton (Editors) *New Perspectives in Drug Design* London, Academic Press, pp 185–200.

Mezei M and D L Beveridge 1986. Free Energy Simulations In Beveridge D L and W L Jorgensen (Editors) *Computer Simulation of Chemical and Biomolecular Systems. Annals of the New York Academy of Sciences* **482**:1–23.

Sandre E and A Pasturel 1997. An Introduction to *Ab-Initio* Molecular Dynamics Schemes *Molecular Simulation* **20**:63–77.

Straatsma T P 1996 Free Energy by Molecular Simulation. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 9 New York, VCH Publishers, pp 81–127.

van Gunsteren W F 1989. Methods for Calculation of Free Energies and Binding Constants. Successes and Problems. In van Gunsteren and P K Weiner (Editors) *Computer Simulation of Biomolecular Systems*. Leiden, ESCOM, pp. 27–59.

References

Allan N L and W C Mackrodt 1994. Oxygen Interstitial Defects in High-T_c Oxides. *Molecular Simulation* **12**:89–100.

Åqvist J, C Medina and J-E Samuelsson 1994 A New Method for Predicting Binding Affinity in Computer-aided Drug Design *Protein Engineering* **7**:385–391.

Åqvist J and A Warshel 1993. Simulation of Enzyme Reactions Using Valence Bond Force Fields and Other Hybrid Quantum/Classical Approaches. *Chemical Reviews* **93**:2523–2544

Åqvist J, M Fothergill and A Warshel 1993. Computer Simulation of the CO_2/HCO_3^- Interconversion Step in Human Carbonic Anhydrase I. *Journal of the American Chemical Society* **115**:631–635

Barrows S E, J W Storer, C J Cramer, A D French and D G Truhlar 1998 Factors Controlling Relative Stability of Anomers and Hydroxymethyl Conformers of Glucopyranose. *Journal of Computational Chemistry* **19**:1111–1129.

Bartlett P A and C K Marlowe 1987. Evaluation of Intrinsic Binding Energy from a Hydrogen-bonding Group in an Enzyme Inhibitor. *Science* **235** 569–571

Bash P A, U C Singh, F K Brown, R Langridge and P A Kollman 1987. Calculation of the Relative Change in Binding Free-Energy of a Protein–Inhibitor Complex. *Science* **235** 574–576.

Bernardi A, A M Capelli, A Comotti, C Gannari, J M Goodman and I Paterson 1990. Transition-State Modeling of the Aldol Reaction of Boron Enolates: A Force Field Approach. *Journal of Organic Chemistry* **55**:3576–3581

Boero M, M Parrinello and K Terakura 1999 Ziegler–Natta Heterogeneous Catalysis by First Principles Computer Experiments *Surface Science* **438**:1–8.

Boresch S, G Archontis and M Karplus 1994 Free Energy Simulations: The Meaning of the Individual Contributions from a Component Analysis. *Proteins: Structure, Function and Genetics* **20**.25–33

Boresch S and M Karplus 1995. The Meaning of Component Analysis· Decomposition of the Free Energy in Terms of Specific Interactions. *Journal of Molecular Biology* **254**:801–807.

Born M 1920 Volumen and Hydratationswärme der Ionen *Zeitschrift fur Physik* 1:45–48

Buetler T C, A E Mark, R C van Schaik, P R Gerber and W F van Gunsteren 1994. Avoiding Singularities and Numerical Instabilities in Free Energy Calculations Based on Molecular Simulations *Chemical Physics Letters* **222**:529–539

Burger M T, A Armstrong, F Guarnieri, D Q McDonald and W C Still 1994. Free Energy Calculations in Molecular Design: Predictions by Theory and Reality by Experiment with Enantioselective Podand Ionophores. *Journal of the American Chemical Society* **116**:3593–3594

Car R and M Parrinello 1985. Unified Approach for Molecular Dynamics and Density Functional Theory *Physical Review Letters* **55**:2471–2474.

Carlson H A and W L Jorgensen 1995 An Extended Linear Response Method for Determining Free Energies of Hydration *Journal of Physical Chemistry* **99**.10667–10673

Chambers C C, G D Hawkins, C J Cramer and D G Truhlar 1996. Model for Aqueous Solvation Based on Class IC Atomic Charges and First Solvation Shell Effects *Journal of Physical Chemistry* **100** 16385–16398

Chandrasekhar J and W L Jorgensen 1985. Energy Profile for a Nonconcerted S_N2 Reaction in Solution. *Journal of the American Chemical Society* **107**:2974–2975.

Chandrasekhar J, S F Smith and W L Jorgsensen 1985. Theoretical Examination of the S_N2 Reaction Involving Chloride Ion and Methyl Chloride in the Gas Phase and Aqueous Solution *Journal of the American Chemical Society* **107**·154–163.

Claverie P, J P Daudey, J Langlet, B Pullman, D Piazzola and M J Huron 1978 Studies of Solvent Effects I. Discrete, Continuum and Discrete-Continuum Models and Their Comparison for Some Simple Cases. NH_4^+, CH_3OH and substituted NH_4^+. *Journal of Physical Chemistry* **82**:405–418.

Constanciel R and R Contreras 1984. Self-Consistent Field Theory of Solvent Effects Representation by Continuum Models – Introduction of Desolvation Contribution. *Theoretica Chimica Acta* **65**.1–11.

Cramer C J and D G Truhlar 1992. AM1-SM2 and PM3-SM3 Parametrized SCF Solvation Models for Free Energies in Aqueous Solution *Journal of Computer-Aided Molecular Design* **6**:629–666

Dapprich S, I Komiromi, K S Byun, K Morokuma and M J Frisch 1999. A New ONIOM Implementation in Gaussian '98 Part I The Calculation of Energies, Gradients, Vibrational Frequencies and Electric Field Derivatives. *THEOCHEM* **461–462**:1–21.

de Wijs G A, G Kresse, L Vočadlo, D Dobson, D Alfè, M J Gillan and G D Price 1998. The Viscosity of Liquid Iron at the Physical Conditions of the Earth's Core *Nature* **392**:805–807

Eisenberg D and A D McLachlan 1986 Solvation Energy in Protein Folding and Binding. *Nature* **319**·199–203.

Elber R and M Karplus 1990. Enhanced Sampling in Molecular Dynamics: Use of the Time-Dependent Hartree Approximation for a Simulation of Carbon Monoxide Diffusion through Myoglobin *Journal of the American Chemical Society* **112**.9161–9175.

Eriksson M A L, J Pitera and P A Kollman 1999. Prediction of the Binding Free Energies of New TIBO-like HIV-1 Reverse Transcriptase Inhibitors Using a Combination of PROFEC, PB/SA, CMC/MD, and Free Energy Calculations. *Journal of Medicinal Chemistry* **42**·868–881

Essex J W, C A Reynolds and W G Richards 1989. Relative Partition Coefficients from Partition Functions A Theoretical Approach to Drug Transport *Journal of the Chemical Society Chemical Communications* 1152–1154

Field M J, P A Bash and M Karplus 1990. A Combined Quantum Mechanical and Molecular Mechanical Potential for Molecular Dynamics Simulations. *Journal of Computational Chemistry* **11**:700–733

Fleischman S H and C L Brooks III 1987 Thermodynamics of Aqueous Solvation – Solution Properties of Alcohols and Alkanes. *Journal of Chemical Physics* **87**.3029–3037.

Floris F and J Tomasi 1989 Evaluation of the Dispersion Contribution to the Solvation Energy – A Simple Computational Model in the Continuum Approximation *Journal of Computational Chemistry* **10**.616–627

Freitag S, J Le Trong, P S Stayton and R E Stenkamp 1997. Structural Studies of the Streptavidin Binding Loop *Protein Science* **6** 1157.

Gilson M K and B Honig 1988. Calculation of the Total Electrostatic Energy of a Macromolecular System: Solvation Energies, Binding Energies and Conformational Analysis *Proteins Structure, Function and Genetics* **4**:7–18.

Gonzalez C and H B Schlegel 1988. An Improved Algorithm for Reaction Path Following. *Journal of Chemical Physics* **90**:2154–2161.

Grimes R W, C R A Catlow and A M Stoneham 1989 Quantum-mechanical Cluster Calculations and the Mott–Littleton Methodology *Journal of the Chemical Society, Faraday Transactions* **85** 485–495.

Guo Z and C L Brooks III 1998. Rapid Screening of Binding Affinities: Application of the λ-Dynamics Method to a Trypsin–Inhibitor System *Journal of the American Chemical Society* **120**.1920–1921.

Guo Z, C L Brooks III and X Kong 1998 Efficient and Flexible Algorithm for Free Energy Calculations using the λ-Dynamics Approach. *Journal of Physical Chemistry* **B102**:2032–2036

Ha S, J Gao, B Tidor, J W Brady and M Karplus 1991. Solvent Effect on the Anomeric Equilibrium in D-Glucose: A Free Energy Simulation Analysis *Journal of the American Chemical Society* **113**:1553–1557.

Hansson T and J Åqvist 1995 Estimation of Binding Free Energies for HIV Proteinase Inhibitors by Molecular Dynamics Simulations *Protein Engineering* **8**.1137–1144

Hansson T, J Marelius and J Åqvist 1998. Ligand Binding Affinity Prediction by Linear Interaction Energy Methods. *Journal of Computer-Aided Molecular Design* **12**:27–35

Hasel W, T F Hendrickson and W C Still 1988 A Rapid Approximation to the Solvent Accessible Surface Areas of Atoms *Tetrahedron Computer Methodology* 1:103–116

Honig B and A Nicholls 1995. Classical Electrostatics in Biology and Chemistry. *Science* **268**:1144–1149.

Jones-Hertzog D K and W L Jorgensen 1997. Binding Affinities for Sulphonamide Inhibitors with Human Thrombin Using Monte Carlo Simulations with a Linear Response Method. *Journal of Medicinal Chemistry* **40**:1539–1549

Jorgensen W L, J M Briggs and M L Contreras 1990. Relative Partition Coefficients for Organic Solutes from Fluid Simulations *Journal of Physical Chemistry* **94**.1683–1986.

Jorgensen W L and J K Buckner 1987. Use of Statistical Perturbation Theory for Computing Solvent Effects on Molecular Conformation. Butane in Water *Journal of Physical Chemistry* **91**:6083–6085.

Jorgensen W L, J K Buckner, S Boudon and J Tirado-Reeves 1988. Efficient Computation of Absolute Free Energies of Binding by Computer Simulations – Applications to the Methane Dimer in Water. *Journal of Chemical Physics* **89**:3742–3746

Jorgensen W L, J Gao and C Ravimohan 1985. Monte Carlo Simulations of Alkanes in Water: Hydration Numbers and the Hydrophobic Effect. *Journal of Physical Chemistry* **89**:3470–3473

Kirkwood J G 1934 Theory of Solutions of Molecules Containing Widely Separated Charges with Special Application to Zwitterions. *Journal of Chemical Physics* **2**:351–361.

Klamt A 1995. Conductor-like Screening Model for Real Solvent: A New Approach to the Quantitative Calculation of Solvation Phenomena. *Journal of Physical Chemistry* **99** 2224–2235

Klamt A, V Jonas, T. Bürger and J C W Lohrenz 1998 Refinements and Parametrisation of COSMO-RS *Journal of Physical Chemistry* **102**:5074–5085.

Klamt A and G Schüürmann 1993. COSMO: A New Approach to Dielectric Screening in Solvents with Explicit Expressions for the Screening Energy and its Gradient *Journal of the Chemical Society, Perkin Transations* **2**:799–805

Klapper I, R Hagstrom, R Fine, K Sharp and B Honig 1986 Focusing of Electric Fields in the Active Site of CuZn Superoxide Dismutase: Effects of Ionic Strength and Amino-Acid Substitution *Proteins Structure, Function and Genetics* **1**:47–59

Kong X and C L Brooks III 1996 λ-Dynamics: A New Approach to Free Energy Calculations. *Journal of Chemical Physics* **105**.2414–2423.

Laasonen, M Sprik and M Parrinello 1993. 'Ab Initio' Liquid Water. *Journal of Chemical Physics* **99**.9080–9089.

Liu H, A E Mark and W F van Gunsteren 1996 Estimating the Relative Free Energy of Different Molecular States with Respect to a Single Reference State. *Journal of Physical Chemistry* **100**·9485–9494

Lybrand T P, J A McCammon and G Wipff 1986 Theoretical Calculation of Relative Binding Affinity in Host–Guest Systems. *Proceedings of the National Academy of Sciences USA* **83**:833–835.

Mackrodt W C 1982 Defect Calculations for Ionic Materials *Lecture Notes in Physics* **166** (Computer Simulation of Solids).175–194

Marquart M, J Walter, J Deisenhofer, W Bode and R Huber 1983. The Geometry of the Reactive Site and of the Peptide Groups in Trypsin, Trypsinogen and its Complexes with Inhibitors. *Acta Crystallographica* **B39**:480–490.

Maseras F and K Morokuma 1995 IMOMM· A New Integrated *Ab Initio* + Molecular Mechanics Geometry Optimisation Scheme of Equilibrium Structures and Transition States. *Journal of Computational Chemistry* **16**.1170–1179.

McRee D E, S M Redford, E D Getzoff, J R Lepock, R A Hallewell and J A Tainer 1990. Changes in Crystallographic Structure and Thermostability of a Cu, Zn Superoxide Dismutase Mutant Resulting from the Removal of Buried Cysteine. *Journal of Biological Chemistry* **265**·14234–14241

Merz K M Jr and P A Kollman 1989 Free Energy Perturbation Simulations of the Inhibition of Thermolysin: Prediction of the Free Energy of Binding of a New Inhibitor. *Journal of the American Chemical Society* **111**·5649–5658

Miertus S, E Scrocco and J Tomasi 1981 Electrostatic Interaction of a Solute with a Continuum – A Direct Utilization of *Ab Initio* Molecular Potentials for the Provision of Solvent Effects. *Chemical Physics* **55**.117–129.

Miick S M, G V Martinez, W R Fiori, A P Todd and G L Millhauser 1992. Short Alanine-based Peptides May Form 3(10)-Helices and not Alpha-helices in Aqueous Solution *Nature* **359**:653–655

Mitchell M J and J A McCammon 1991 Free Energy Difference Calculations by Thermodynamic Integration· Difficulties in Obtaining a Precise Value. *Journal of Computational Chemistry* **12**.271–275.

Miyamoto S and P A Kollman 1993a Absolute and Relative Binding Free Energy Calculations of the Interaction of Biotin and its Analogues with Streptavidin Using Molecular Dynamics/Free Energy Perturbation Approaches. *Proteins: Structure, Function and Genetics* **16**·226–245.

Miyamoto S and P A Kollman 1993b What Determines the Strength of Noncovalent Association of Ligands to Proteins in Aqueous Solution? *Proceedings of the National Academy of Sciences USA* **90**·8402–8406

Mott N F and M J Littleton 1938. Conduction in Polar Crystals. I Electrolytic Conduction in Solid Salts *Transactions of the Faraday Society* **34**: 485–499.

Onsager L 1936 Electric Moments of Molecules in Liquids *Journal of the American Chemical Society* **58**.1486–1493

Paschual-Ahuir J L, E Silla, J Tomasi and R Bonaccorsi 1987. Electrostatic Interaction of a Solute with a Continuum. Improved Description of the Cavity and of the Surface Cavity Bound Charge Distribution. *Journal of Computational Chemistry* **8** 778–787

Payne M C, M P Teter, D C Allan, R A Arias and D J Joannopoulos 1992 Iterative Minimisation Techniques for *Ab Initio* Total-Energy Calculations. Molecular Dynamics and Conjugate Gradients. *Reviews of Modern Physics* **64**.1045–1097

Pearlman D A and P A Kollman 1989. A New Method for Carrying Out Free-Energy Perturbation Calculations – Dynamically Modified Windows *Journal Of Chemical Physics* **90**:2460–2470.

Pierotti R 1965. Aqueous Solutions of Nonpolar Gases *Journal of Physical Chemistry* **69**:281–288

Pisani C 1999 Software for the Quantum-mechanical Simulation of the Properties of Crystalline Materials: State of the Art and Prospects. *THEOCHEM* **463**:125–137

Pitera J and P Kollman 1998 Designing an Optimum Guest for a Host Using Multimolecule Free Energy Calculations: Predicting the Best Ligand for Rebek's 'Tennis Ball'. *Journal of the American Chemical Society* **120**:7557–7567.

Postma J P M, H J C Berendsen and J R Haak 1982 Thermodynamics of Cavity Formation in Water *Faraday Symposium of the Chemical Society* 17:55–67.

Qiu D, P S Shenkin, F P Hollinger and W C Still 1997. The GB/SA Continuum Model for Solvation A Fast Analytical Method for the Calculation of Approximate Born Radii. *Journal of Physical Chemistry* **101** 3005–3014.

Rashin A A 1990. Hydration Phenomena, Classical Electrostatics, and the Boundary Element Method. *Journal of Physical Chemistry* **94**:1725–1733.

Rashin A A and B Honig 1985 Reevaluation of the Born Model of Ion Hydration. *Journal of Physical Chemistry* **89** 5588–5593.

Rashin A A and K Namboodiri 1987 A Simple Method for the Calculation of Hydration Enthalpies of Polar Molecules with Arbitrary Shapes. *Journal of Physical Chemistry* **91**.6003–6012.

Remler D K and P A Madden 1990 Molecular Dynamics without Effective Potentials via the Car–Parrinello Approach. *Molecular Physics* **70**.921–966

Reuter N, A Dejaegere, B Maigret and M Karplus 2000. Frontier Bonds in QM/MM Methods: A Comparison of Different Approaches *Journal of Physical Chemistry* **A104**:1720–1733.

Rinaldi D, M F Ruiz-Lopez and J L Rivail 1983. *Ab Initio* SCF Calculations on Electrostatically Solvated Molecules Using a Deformable Three Axes Ellipsoidal Cavity *Journal of Chemical Physics* **78**:834–838.

Röthlisberger and M Parrinello 1997. *Ab Initio* Molecular Dynamics Simulation of Liquid Hydrogen Fluoride *Journal of Chemical Physics* **106** 4658–4664.

Ryckaert J-P and A Bellemans 1978 *Molecular Dynamics of Liquid Alkanes, Faraday Discussions* **20**:95–106.

Saitta A M, P D Sooper, E Wasserman and M L Klein 1999 Influence of a Knot on the Strength of a Polymer Strand. *Nature* **399**:46–48

Schäfer H, W F van Gunsterten and A E Mark 1999 Estimating Relative Free Energies from a Single Ensemble Hydration Free Energies. *Journal of Computational Chemistry* **20**:1604–1617

Silvestrelli P L and M Parrinello 1999. Structural, Electronic and Bonding Properties of Liquid Water from First Principles *Journal of Chemical Physics* **111**:3572–3580.

Simmerling C and R Elber 1995. Computer Determination of Peptide Conformations in Water: Different Roads to Structure *Proceedings of the National Academy of Sciences USA* **92**:3190–3193

Simmerling C, T Fox and P A Kollman 1998. Use of Locally Enhanced Sampling in Free Energy Calculations: Testing and Application to the $\alpha \rightarrow \beta$ Anomerisation of Glucose. *Journal of the American Chemical Society* **120**:5771–5782.

Singh U C and P A Kollman 1986. A Combined *Ab Initio* Quantum Mechanical and Molecular Mechanical Method for Carrying out Simulations on Complex Molecular Systems: Applications to the $CH_3Cl + Cl^-$ Exchange Reaction and Gas Phase Protonation of Polyethers. *Journal of Computational Chemistry* **7**:718–730

Sitkoff D, K A Sharp and B Honig 1994 Accurate Calculation of Hydration Free Energies Using Macroscopic Solvent Models. *Journal of Physical Chemistry* **98** 1978–1988

Smith P E and B M Pettitt 1994. Modeling Solvent in Biomolecular Systems *Journal of Physical Chemistry* **98**.9700–9711.

Smith P E and W F van Gunsteren 1994a Predictions of Free Energy Differences from a Single Simulation of the Initial State. *Journal of Chemical Physics* **100**:577–585.

Smith P E and W F van Gunsteren 1994b. When Are Free Energy Components Meaningful? *Journal of Physical Chemistry* **98**:13735–13740

Smythe M L, S E Huston and G R Marshall 1993 Free Energy Profile of a 3_{10} to α-Helical Transition of an Oligopeptide in Various Solvents. *Journal of the American Chemical Society* **115**:11594–11595.

Smythe M L, S E Huston and G R Marshall 1995. The Molten Helix: Effects of Solvation on the α- to 3_{10}-Helical Transition. *Journal of the American Chemical Society* **117** 5445–5452.

Sprik M, J Hutter and M Parrinello 1996. *Ab Initio* Molecular Dynamics Simulation of Liquid Water. Comparison of Three Gradient-corrected Density Functionals *Journal of Chemical Physics* **105**:1142–1152.

Stich I, A De Vita, M C Payne, M J Gilland and L J Clarke 1994. Surface Dissociation from First Principles: Dynamics and Chemistry. *Physical Review* **B49**.8076–8085.

Still W C, A Tempczyrk, R C Hawley and T Hendrickson 1990. Semianalytical Treatment of Solvation for Molecular Mechanics and Dynamics. *Journal of the American Chemical Society* **112**:6127–6129

Svensson M, S Humbel, R D J Froese, T Matsubara, S Sieber and K Morokuma 1996. ONIOM. A Multilayered Integrated MO+MM Method for Geometry Optimisations and Single Point Energy Predictions. A Test for Diels–Alder Reactions and $Pt(P(t\text{-}Bu)_3)_2 + H_2$ Oxidative Addition. *Journal of Physical Chemistry* **100**:19357–19363.

Tapia O and O Goscinski 1975 Self-Consistent Reaction Field Theory of Solvent Effects. *Molecular Physics* **29**:1653–1661.

Taylor M B, G D Barrera, N L Allan, T H K Barron and W C Mackrodt 1997 Free Energy of Formation of Defects in Polar Solids *Faraday Discussions* **106**:377–387

Tirado-Reeves J, D S Maxwell and W L Jorgensen 1993. Molecular Dynamics and Monte Carlo Simulations Favor the α-Helical Form for Alanine-Based Peptides in Water. *Journal of the American Chemical Society* **115**:11590–11593.

Tobias D J and C L Brooks III 1988. Molecular Dynamics with Internal Coordinate Constraints. *Journal of Chemical Physics* **89**:5115–5126.

Torrie G M and J P Valleau 1977 Nonphysical Sampling Distributions in Monte Carlo Free-Energy Estimation Umbrella Sampling. *Journal of Computational Physics* **23** 187–199

Tuckerman M, K Laasonen, M Sprik and M Parrinello 1995a. *Ab Initio* Molecular Dynamics Simulation of the Solvation and Transport of Hydronium and Hydroxyl Ions in Water. *Journal of Chemical Physics* **103**:150–161

Tuckerman M, K Laasonen, M Sprik and M Parrinello 1995b *Ab Initio* Molecular Dynamics Simulation of the Solvation and Transport of H_3O^+ and OH^- Ions in Water. *Journal of Physical Chemistry* **99**:5749–5752

Wall I D, A R Leach, D W Salt, M G Ford and J W Essex 1999. Binding Constants of Neuraminidase Inhibitors: An Investigation of the Linear Interaction Energy Method. *Journal of Medicinal Chemistry* **42**:5142–5152

Wang W, J Wang and P A Kollman 1999. What Determines the van der Waals Coefficient β in the LIE (Linear Interaction Energy) Method to Estimate Binding Free Energies Using Molecular Dynamics Simulations? *Proteins: Structure, Function and Genetics* **34**:395–402.

Warshel A 1991. *Computer Modelling of Chemical Reactions in Enzymes and Solutions* New York, John Wiley & Sons

Warshel A and M Levitt 1976. Theoretical Studies of Enzymic Reactions Dielectric, Electrostatic and Steric Stabilization of the Carbonium Ion in the Reaction of Lysozyme. *Journal of Molecular Biology* **103**:227–249.

Warwicker J and H C Watson 1982 Calculation of the Electric Potential in the Active-Site Cleft Due to Alpha-Helix Dipoles. *Journal of Molecular Biology* **157**:671–679

Wodak S J and J Janin 1980 Analytical Approximation to the Solvent Accessible Surface Area of Proteins *Proceedings of the National Academy of Sciences USA* **77**:1736–1740.

Wong M W, K B Wiberg and M J Frisch 1992 Solvent Effects 3 Tautomeric Equilibria of Formamide and 2-Pyridone in the Gas Phase and Solution. An *Ab Initio* SCRF Study. *Journal of the American Chemical Society* **114**:1645–1652.

Yu H-A and M Karplus 1988 A Thermodynamic Analysis of Solvation *Journal of Chemical Physics* **89** 2366–2379

Zhang L and J Hermans 1994. 3_{10}-Helix versus α-Helix: A Molecular Dynamics Study of Conformational Preferences of Aib and Alanine. *Journal of the American Chemical Society* **116**:11915–11921.

Zhang X and C R A Catlow 1992 Molecular Dynamics Study of Oxygen Diffusion in $YBa_2Cu_3O_{6.19}$. *Physical Review* **B46**:457–462

Zwanzig R W 1954. High-temperature Equation of State by a Perturbation Method. I. Nonpolar Gases *Journal of Chemical Physics* **22**:1420–1426

CHAPTER TWELVE

The Use of Molecular Modelling and Chemoinformatics to Discover and Design New Molecules

Molecular modelling techniques are widely used in the chemical, pharmaceutical and agrochemical industries. Much of this modelling activity employs the tools that we have discussed in earlier chapters, such as energy minimisation, molecular dynamics and Monte Carlo simulations and conformational analysis. In this chapter, we will discuss a number of methods that do not fit naturally into any of these categories or which bring together several of these tools to create a new approach. Other techniques have been developed as a consequence of, or in conjunction with, some new technological advance such as combinatorial chemistry or high-throughput screening. Our discussion will often but not exclusively use examples drawn from the pharmaceutical industry, though many of the techniques are also applicable to molecular design in other areas.

12.1 Molecular Modelling in Drug Discovery

Most drugs produce their effect by interacting with a biological macromolecule such as an enzyme, DNA, glycoprotein or receptor. The interaction between a ligand and its target* may be due entirely to non-bonded forces, but in some cases a covalent interaction may be involved. Drugs which interact with receptor proteins can be classified as *agonists*, *antagonists* or *inverse agonists*. Agonists produce the same or elevated effect as the natural substrate or effector molecule, whereas antagonists inhibit the effect of the natural ligand. Inverse agonists create an effect which appears opposite to that of the agonist. Tight-binding ligands often have a high degree of complementarity with the target. This complementarity can be assessed and measured in various ways. Many ligands show significant shape complementarity with the region of the macromolecule where they bind (the binding

* We shall use the generic term 'ligand' to indicate the inhibitor or substrate and the term 'receptor' to indicate the macromolecule to which it binds, be it an enzyme, a gene or a receptor protein

site). This can be observed by constructing the molecular surfaces as illustrated in Figure 11.12 (colour plate section), which shows the molecular surfaces of biotin bound to streptavidin. The ligand often forms hydrogen bonds with the receptor. Some receptors have hydrophobic 'pockets', formed by groups of non-polar amino acids, into which the ligand can place a hydrophobic group of an appropriate size. It is also crucial to remember that a good drug does more than simply bind tightly to its target. After administration, a drug must get to the site of action This transport process often requires the drug to pass through cell membranes. A cell membrane is a hydrophobic environment and so the drug must be sufficiently lipophilic (lipid loving) to partition into the membrane but not so lipophilic that it stays there. Once inside the cell, the drug must access its target. During this process, the molecule may also be removed from the body by metabolism, excretion and other pathways.

Discovering and developing any new medicine is a long and expensive process A new compound must not only produce the desired response with minimal side-effects but must also be demonstrably better than existing therapies. Two key steps in many drug discovery programmes are the identification of hit molecules ('hits') and lead series ('leads'). A hit is a molecule that has some reproducible activity in a biological assay A lead series comprises a set of related molecules that usually share some common structural feature, and which show some variation in the activity as the structure is modified. This gives one confidence that further synthetic modification to the lead series (termed *lead optimisation*) has a good chance of resulting in a drug candidate with the desired potency and selectivity, lack of toxicity and the appropriate characteristics to enable it to reach its target *in vivo*. Such a drug candidate will then enter the early stages of development, where further large-scale investigations are undertaken.

Finding novel lead series can be a difficult problem. Serendipity often played an important role in the past, a classic example being the discovery of penicillin by Alexander Fleming. For many years pharmaceutical companies have screened soil and other biological samples to find new leads, but it can be difficult to extract and purify the bioactive ingredient. The 1990s saw the widespread adoption of high-throughput screening (HTS), which enables large numbers of compounds to be screened using highly automated, robotic techniques The molecules used for HTS come from compounds synthesised in earlier medicinal chemistry programmes, from compounds that can be purchased from chemical suppliers and from combinatorial chemistry (discussed in Section 12.14). However, although HTS makes it possible in principle to test every available compound against every biological assay, there are a number of practical reasons why this is not necessarily feasible, let alone desirable. The first reason is financial· although robotics and miniaturisation have significantly reduced the unit cost, the sheer number of samples now available in many companies means that the overall expense can be significant. A second reason is that some assays cannot be converted to a high-throughput format and so have to be conducted using more traditional techniques. Third, a significant proportion of the available samples might not be considered appropriate structures, suitable for taking forward to the next stage. For example, some molecules may contain functional groups which are known to react in a non-specific manner with biological targets Other molecules might interfere with the proper interpretation of the assay, such as a strong fluorophore. Yet more molecules may just be considered 'inappropriate', or not sufficiently 'drug-like'

For these and other reasons, it is often necessary to identify subsets of compounds. Computational techniques have a significant role to play in the ways in which such subsets can be constructed, with various techniques being available depending upon the type of molecule that one wishes to screen, what kind of information is available to assist the selection and what properties one wishes to take into account. Sections 12.2–12.11 describe a wide variety of methods that can be used either individually or in combination to select compounds. Some of these methods only use information about the underlying chemical structure of the molecule. These are often referred to as '2D' properties, as distinct from '3D' methods, which take into account the three-dimensional nature of a molecule (i.e. its conformation and properties dependent upon the conformation). Some of the methods can take into account information about the target protein or about other molecules that are known to be active at the target, whereas other methods are designed to produce 'diverse' collections of compounds for more general screening

Having tested a number of compounds, it is then usually desired to construct a model which relates the observed activity to the molecular structure. The model can then be used in the next iteration of the process Many different kinds of model are possible. A popular approach is to use statistical techniques to derive the model. Such statistical techniques are discussed in Sections 12.12–12.13.

12.2 Computer Representations of Molecules, Chemical Databases and 2D Substructure Searching

Substructure searching is probably the most basic approach to identifying compounds of interest. It is widely used for all kinds of problems. Most chemists take substructure searching for granted, testament to the decades of effort that has gone into the development of extremely powerful algorithms and database systems.

Many organisations maintain databases of chemical compounds. Some of these databases are publicly accessible; others are proprietary. A database may contain an extremely large number of compounds, several hundred thousand is common, and the database maintained by the American Chemicals Society contains more than 18 million compounds. A more recent development involves the creation of databases containing *virtual* molecules. These are compounds that do not yet exist but which could be synthesised readily, typically using combinatorial chemistry techniques. We do not have space to consider in any detail the nature of chemical database systems, save for a few key points. The first issue concerns the representation of molecular structures in a computer. We are all familiar with the chemical diagrams in journals and lab-books, but simply storing the chemical diagram itself (as an image) is of little value. Rather, most systems represent molecules as *molecular graphs*. A *graph* contains *nodes*, which are connected by *edges*. Two examples are shown in Figure 12 1 In a molecular graph, the nodes correspond to the atoms and the edges to the bonds, as shown for acetic acid in Figure 12.2. The locations of the nodes and edges of a graph on the page are irrelevant; only the way in which the nodes are connected together matters. The conformational search trees that we met in Section 9.2 are a special kind of graph. A *subgraph* is a subset of the nodes and edges of a graph; thus the graph for CH_3 is

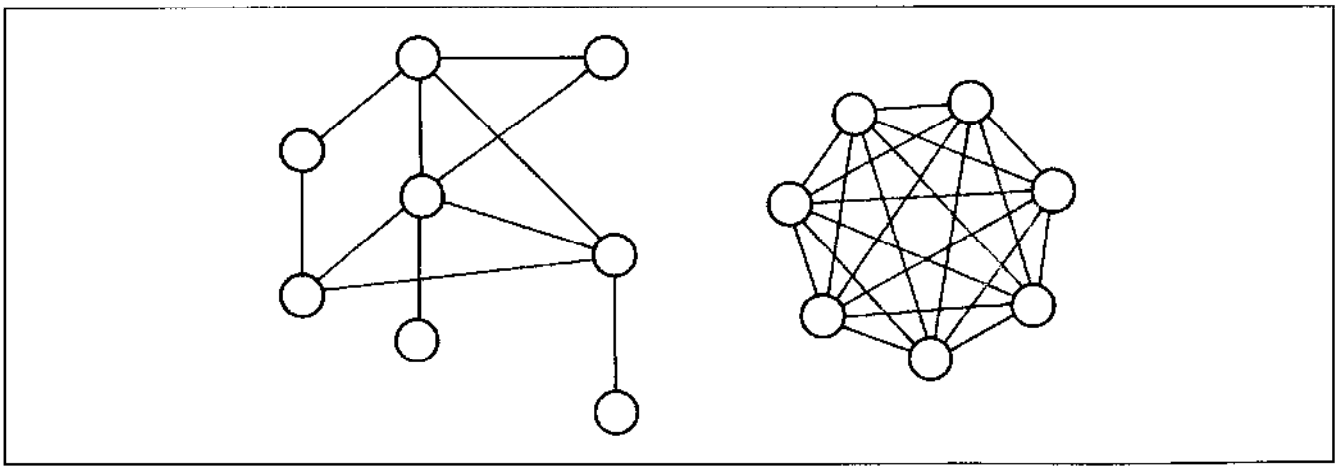


Fig. 12.1: Graphs contain nodes connected by edges. A completely connected graph (right) has an edge between all pairs of nodes.

a subgraph of the graph of acetic acid. A graph is said to be *completely connected* if there is an edge between all pairs of nodes. Only in rare cases is the molecular graph a completely connected graph, one example being the P_4 form of elemental phosphorus.

There are a number of different ways that the molecular graph can be communicated between the computer and the end-user. One common representation is the *connection table*, of which there are various flavours, but most provide information about the atoms present in the molecule and their connectivity. The most basic connection tables simply indicate the atomic number of each atom and which atoms form each bond; others may include information about the atom hybridisation state and the bond order. Hydrogens may be included or they may be implied. In addition, information about the atomic coordinates (for the standard two-dimensional chemical drawing or for the three-dimensional conformation) can be included. The connection table for acetic acid in one of the most popular formats, the Molecular Design mol format [Dalby *et al.* 1992], is shown in Figure 12.3.

An alternative way to represent molecules is to use a linear notation. A linear notation uses alphanumeric characters to code the molecular structure. These have the advantage of being much more compact than the connection table and so can be particularly useful for transmitting information about large numbers of molecules. The most famous of the early line notations is the Wiswesser line notation [Wiswesser 1954]; the SMILES notation is a more recent example that is increasingly popular [Weininger 1988]. To construct the Wiswesser

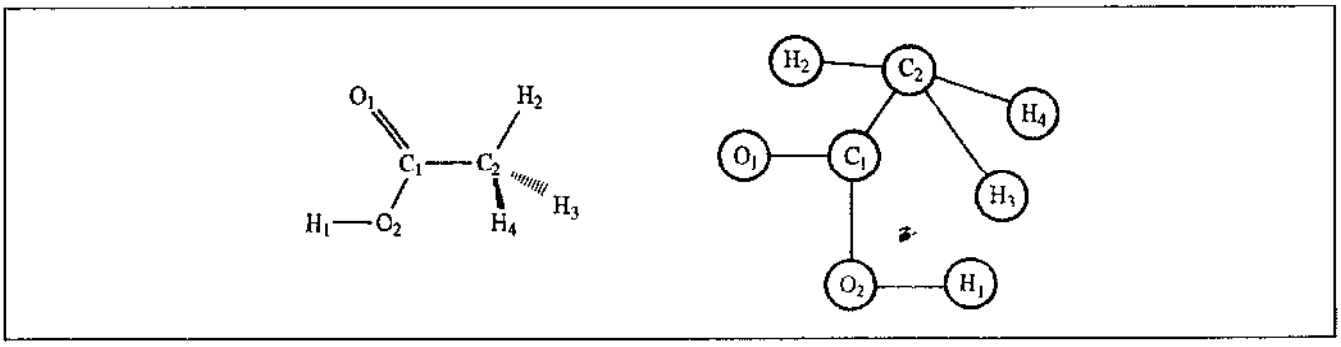


Fig. 12.2. The molecular graph of acetic acid.

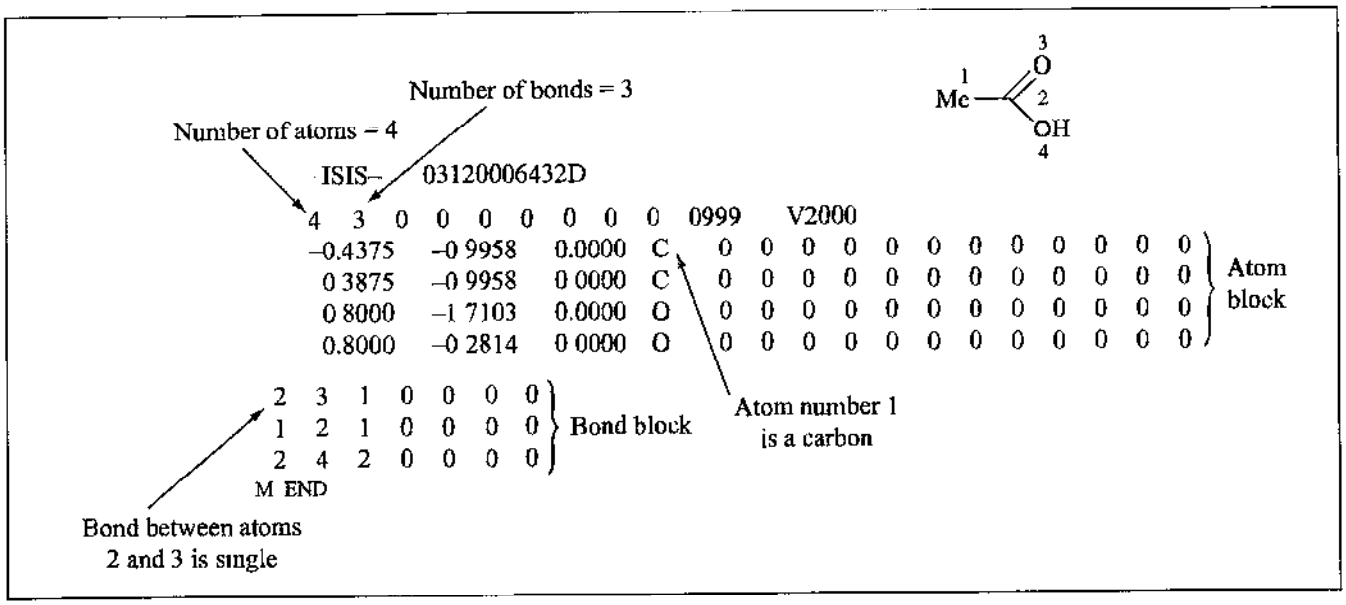


Fig 12.3. MDL mol file for acetic acid, in the hydrogen-suppressed form.

notation for a molecule requires the application of a complex series of rules. SMILES is rather simpler, and with just a few rules one can write and interpret most SMILES strings. Thus atoms are represented by their atomic symbol. Hydrogens are not explicitly represented, except in special cases, as it is a *hydrogen-suppressed* notation Upper case is used for aliphatic atoms, lower case for aromatic atoms. Single or aromatic bonds are not explicitly written but are assumed. Double bonds are represented by '=', triple bonds by '#'. The SMILES is constructed by 'walking' through the chemical diagram from one end to the other such that all atoms are visited just once Rings are dealt with by 'breaking' one of the ring bonds, which are then indicated by appending an integer to the relevant atoms. Branching is indicated using brackets; any level of nesting if possible. Thus the simplest SMILES is probably C (methane). Ethane is CC, propane is CCC, 2-methyl propane is CC(C)C. Cyclohexane is C1CCCCC1 (note the use of the integer to indicate the ring bond). Benzene is c1ccccc1. Acetic acid is CC(=O)O. The SMILES for ranitidine (see Figure 9 26) is CNC(=CN(=O)=O)NCCSCc1ccc(CN(C)C)o1. Information about stereochemistry and geometrical isomerism can also be included in the SMILES notation.

One feature of both the connection table and the SMILES string formats is that there may be many different ways to represent the same molecule. Thus one may choose to number the atoms in the connection table in a different order or to write the SMILES differently (for example, acetic acid can be represented as OC(=O)C, O=C(C)O, O=C(O)C, etc.). A key requirement for any chemical database system is that it can determine whether or not a new molecule is already present in the system. This is typically done by generating some form of *canonical representation* of the structure. The canonical representation is unique irrespective of the numbering of the atoms in a mol file or the order of the atoms in a SMILES string. A popular method for doing this is the Morgan algorithm [Morgan 1965], which considers the properties of each atom together with those of its neighbours; this would enable the methyl carbon in acetic acid to be differentiated from the carboxyl carbon. It is also possible to generate a unique SMILES string for each molecule [Weininger

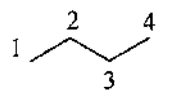
et al. 1989] (the canonical SMILES for acetic acid is CC(=O)O). Some algorithms can also incorporate information about stereochemistry and chirality into the canonicalisation. One immediate use of the canonical representation is that it can provide a very quick method to retrieve information about the compound. This is often done by generating a *hash key* from the structure. The hash key is typically an integer that is often used to indicate the location within a file where the requisite data is stored, so enabling the information to be retrieved very rapidly. The generation of the hash key can be done using well-established computer algorithms which can take a string of characters and generate the requisite integer.

A substructure search retrieves all the molecules from the database that contain the substructure. For example, we might wish to identify all compounds containing a carboxylic acid group. More complex queries are also possible in most systems; these would, for example, permit a query atom to match groups of atoms (e.g. 'any halogen') or features such as ring bonds or to specify stereochemistry. In the language of graph theory, substructure searching is known as *subgraph isomerism* – determining whether one graph is entirely contained within another. Even with the most efficient algorithms this is a relatively time-consuming process and so chemical database systems commonly use some form of screening method to rapidly eliminate molecules that cannot match the query. Such screens are often implemented using binary representations (a *bitstring*) and so operate very rapidly, especially if held in memory. There are two types of binary screen in common use. In a *structural key*, each position in the bitstring corresponds to a particular substructure. If that substructure is present in the molecule then the relevant bit in the molecule's key is set to 1. A predefined fragment dictionary is used to specify the substructures. As each molecule is added to the database a substructure search is performed for each fragment and the relevant bit assigned. Many different types of substructure can be incorporated, such as the presence or absence of particular elements, rings and common functional groups. It is also possible to assign bits which encode how many occurrences of a particular feature there are, such as 'at least two methyl groups'. The features in the fragment dictionary are defined to give optimal performance in 'typical' searches, depending upon the type of molecules in the database. This has the advantage that, when an effective choice of screens is used, the performance of the search should be very efficient, but if the dictionary is not appropriate then many molecules will pass through the screen and be subjected to the slow atom-based substructure search. Thus, a dictionary designed for typical 'organic' or 'drug-like' molecules might be inappropriate for a database containing just hydrocarbon molecules. The structural keys used by the MACCS and Isis systems from Molecular Design are probably the best known of this type of bitstring.

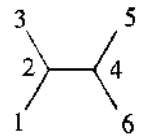
The alternative is to use a *hashed fingerprint*, which does not require a predefined fragment dictionary. Rather, an algorithmic approach is used to derive the bitstring, which initially contains all zeros. This method generates all possible linear paths of connected atoms through the molecule containing between 1 and a pre-defined number of atoms (e.g. 8). For example, in acetic acid the paths of length zero are just the atoms C and O, the paths of length 1 are CC, C=O and CO, and of length 2 are CCO and CC=O. Each path defines a pattern of atoms and bonds which serves as the input to a pseudo-random number generator, which produces a set of bits which are then set to the value 1. The hashing process typically sets 4 or 5 bits per pattern A bitstring might contain 1024 bits, and after all paths

have been examined a typical organic, drug-like molecule might have a total of 200–300 bits set to 1. Obviously, the greater the number of different paths in the molecule the more bits (on average) that are set. Note that the use of the hashing algorithm means that it is possible (indeed, quite likely for typical molecules) that any one bit could be set by more than one pattern. However, it is much less likely (though nevertheless still possible) that the same set of bits would be set by different patterns. Hashed fingerprints are used in a number of database systems and are particularly associated with the systems from Daylight Chemical Information Systems.

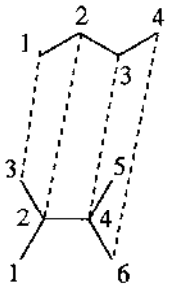
When using a bitstring screen, the first operation is to calculate the corresponding bitstring for the substructure query. This query bitstring is then compared with the bitstrings for all the molecules in the database. A molecule can only possibly match the query if it contains a '1' for every position in the bitstring where the query also has a '1'. This comparison can be performed very quickly and so the database can be screened very rapidly. Well-designed screens can eliminate up to 99% of the molecules during this phase. The presence of clashes in hashed fingerprints does not affect the final results of a substructure search, though they



$$\mathbf{S} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$



$$\mathbf{M} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$



$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{A(AM)}^{\mathrm{T}} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \mathbf{S}$$

Fig. 12.4: Illustration of the operation of the Ullmann algorithm using a 4-atom substructure and a 6-atom 'molecule'. The proposed match is shown in the bottom figure, together with the relevant matrices used in the calculation

might have an impact upon the screening efficiency, because more molecules need to be considered for the full substructure search.

Having eliminated molecules that could not match the query using the bitstring screen it is then necessary to undertake the more time-consuming atom-by-atom search for the molecules that remain. One commonly used method for the subgraph isomorphism problem was described by Ullmann [Ullmann 1976]. This algorithm represents the molecular graphs of both the query substructure and the potential molecular match by an *adjacency matrix*, which is a square, symmetric matrix such that the element (ij) has the value 1 if atoms i and j are bonded, and zero otherwise. Sample adjacency matrices are shown in Figure 12.4. If there are N_m atoms in the database molecule and N_s atoms in the substructure then the Ullmann algorithm tries to find matrices **A** such that $\mathbf{A}(\mathbf{AM})^T$ is identical to **S**, where **M** is the adjacency matrix of the molecule and **S** is the adjacency matrix of the substructure. The matrix **A** has N_m columns and N_s rows such that each row contains just one 1 and each column contains no more than one 1. This matrix represents a possible match between the substructure and the molecule such that if an element A_{ij} is set to 1 then the atom numbered j in the substructure matches atom i in the molecule. Figure 12.4 shows an example of a matrix **A** which does indeed correspond to a successful match. The simplest implementation of the Ullmann algorithm is to generate all possible matrices **A** systematically, testing each to see whether they meet the requirements and represent a match. However, refinements of this simplistic (and time-consuming) algorithm are possible. Indeed, in his original paper Ullmann showed that a consideration of the neighbours of each potential match could dramatically improve its performance. A query atom cannot match a database atom unless each of the neighbour atoms of the query atom also matches a neighbour of the database atom.

12.3 3D Database Searching

2D substructure searching is a very powerful and widely used technique for identifying molecules with some particular feature (or combination of features, as the substructure can contain disconnected fragments). However, it does have some serious limitations if we wish to discover novel molecules with the desired biological activity. Key to understanding these limitations is the fact that receptors do not recognise substructures – rather, it is the three-dimensional stereoelectronic features of a molecule that are important for molecular recognition. In a 3D database search one tries to find molecules that satisfy the chemical and geometric requirements of the receptor. As such, a 3D database contains information about the conformational properties and functionality features of the molecules contained within it. Moreover, in contrast to a 2D search, 3D searching can enable lead series to be identified that are structurally quite different from those already known. There are two general types of 3D database search, the choice of which to use being dependent on the information available about the target receptor. In the first case, detailed structural information about the target receptor is not available, but it may be possible to derive an abstract model called a *pharmacophore* that indicates the key features of a series of active molecules. In the second case, a three-dimensional structure of the target macromolecule is available from X-ray crystallography or NMR or comparative modelling.

12.4 Deriving and Using Three-dimensional Pharmacophores

In drug design, the term 'pharmacophore' refers to a set of features that is common to a series of active molecules. Hydrogen-bond donors and acceptors, positively and negatively charged groups, and hydrophobic regions are typical features. We will refer to such features as 'pharmacophoric groups'. These groupings can be considered an illustration of the important concept of *bioisosteres*, which are atoms, functional groups or molecules with similar physical and chemical properties such that they produce generally similar biological properties [Thornber 1979; Patani and LaVoie 1996]. Some common bioisosteric groups are shown in Figure 12.5. A *three-dimensional (3D) pharmacophore* specifies the spatial relationships between the groups. These relationships are often expressed as distances or distance ranges but may also include other geometric measures such as angles and planes. For example, a commonly used 3D pharmacophore for antihistamines contains two aromatic rings and a tertiary nitrogen distributed as shown in Figure 12.6. The development of methods for studying the conformations of ligands has stimulated an interest in the influence of the three-dimensional structures of molecules on their chemical and biological activity. The objective of a procedure known as *pharmacophore mapping* is to determine possible 3D pharmacophores for a series of active compounds and is usually used when an experimental structure of the target macromolecule is not available. Once a pharmacophore has been developed, it can then be used to find or suggest other active molecules.

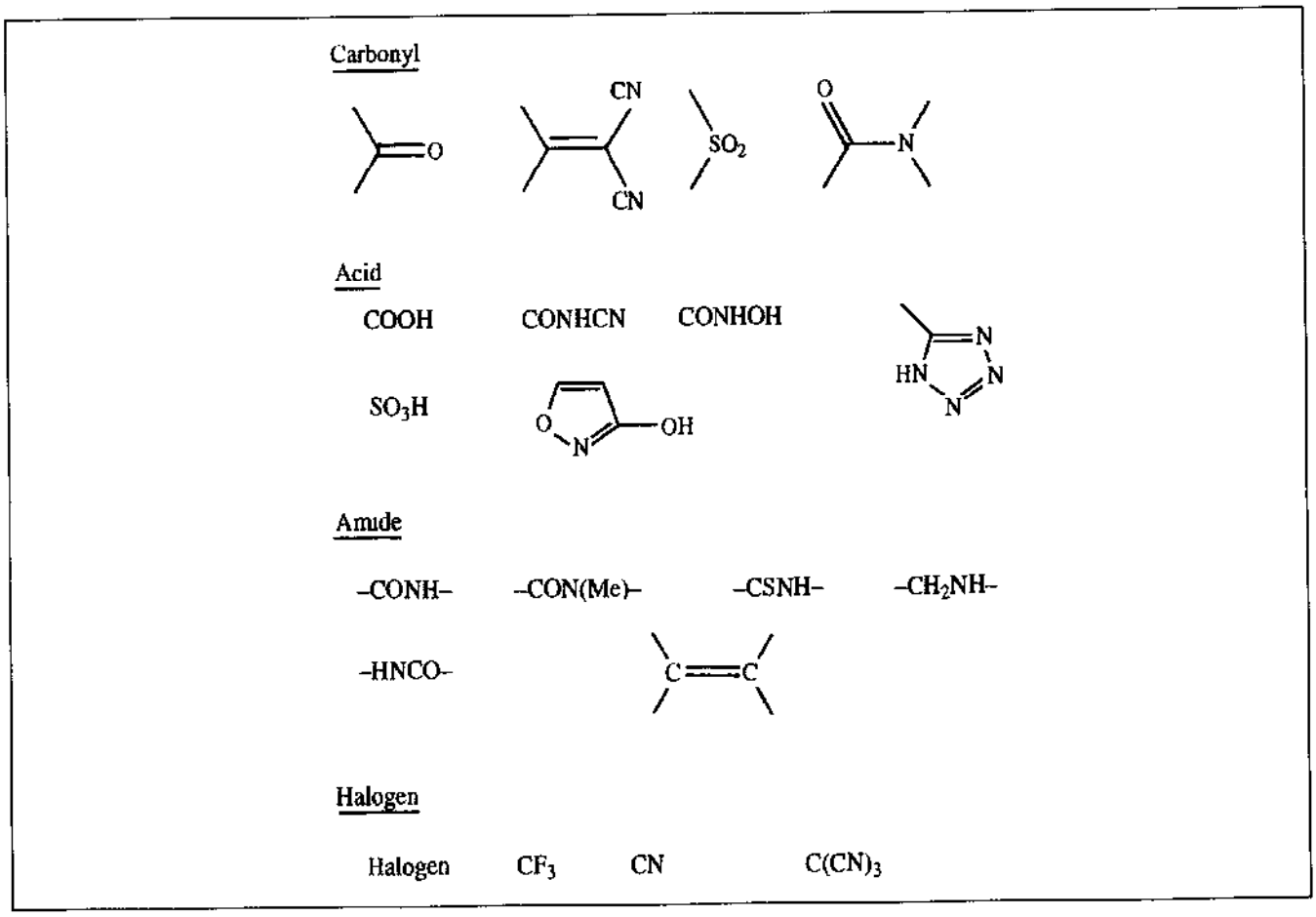


Fig. 12.5 Some common bioisosteric groups

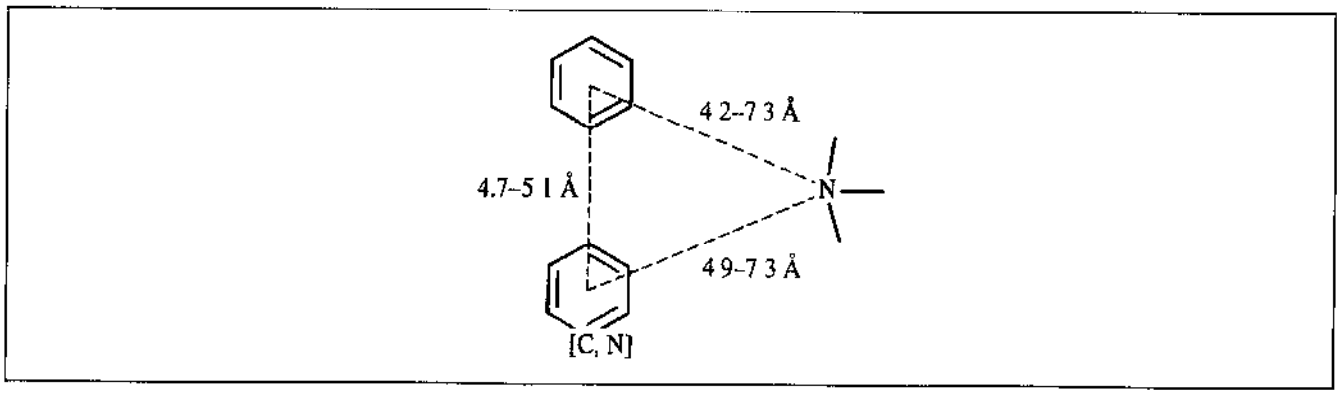


Fig 12 6. Antihistamine 3D pharmacophore

There are two problems to consider when calculating 3D pharmacophores First, unless the molecules are all completely rigid, one must take account of their conformational properties. The second problem is to determine which combinations of pharmacophoric groups are common to the molecules and can be positioned in a similar orientation in space. More than one pharmacophore may be possible; indeed, some algorithms can generate hundreds of possible pharmacophores, which must then be evaluated to determine which best fits the data. It is important to realise that all of these approaches to finding 3D pharmacophores assume that all of the molecules bind in a common manner to the macromolecule.

12.4.1 Constrained Systematic Search

In some cases, it is relatively straightforward to deduce which features are required for activity. A well-known example is the pharmacophore for the angiotension-converting enzyme (ACE), which is involved in regulating blood pressure. Four typical ACE inhibitors are shown in Figure 12.7, including captopril, which is widely used to treat hypertension. Angiotension-converting enzyme is a zinc metalloprotease whose X-ray structure has not yet been solved. Three features within the class of inhibitors such as captopril are required for activity: a terminal carboxyl group (believed to interact with an arginine residue in the enzyme), an amido carbonyl group (which hydrogen bonds to a hydrogen-bond donor in the enzyme), and a zinc-binding group. The problem is to determine conformations in which the inhibitors can position these three pharmacophoric groups in the same relative position in space.

One of the most widely used methods for tackling this problem is the constrained systematic search method of Dammkoehler, Motoc and Marshall [Dammkoehler *et al.* 1989]. At first sight, it would appear that a systematic search over 20–30 molecules would greatly magnify the combinatorial explosion associated with a systematic conformational analysis. In fact, one can significantly reduce the scale of the problem by making use of information about molecules whose conformational space has already been considered. Thus, we are only interested in those conformations that would enable the current molecule's pharmacophoric groups to be positioned in the same locations that have already been found for previous molecules. Dammkoehler and colleagues showed that it is possible to determine what torsion angles of the rotatable bonds will enable conformations consistent with the previous

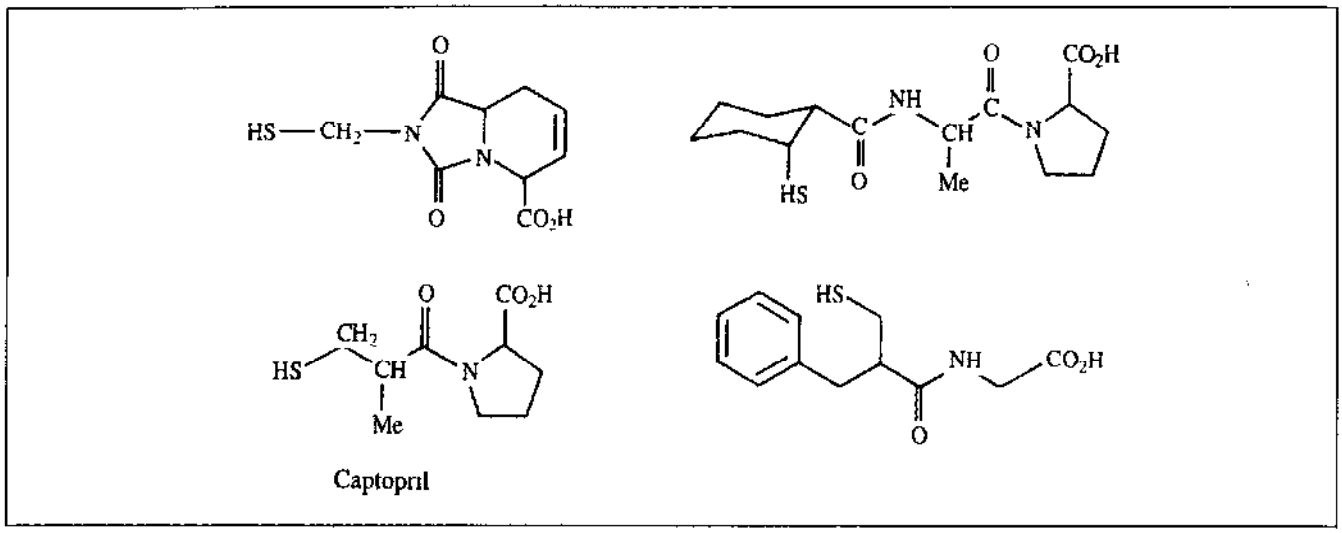


Fig 12.7: Four typical ACE inhibitors.

results to be obtained. It is best to choose the most conformationally restricted molecules first, as these will have a reduced conformational space.

To derive an ACE pharmacophore, four points were defined for each molecule. The derivation of these four points for captopril is shown in Figure 12.8. Five distances (also shown in Figure 12.8) were defined between these four points. Note that one of the points corresponds to the presumed location of the enzyme's zinc atom. The number of rotatable bonds in each inhibitor varied between 3 and 9 and the molecules were considered in order of increasing number of rotatable bonds. The entire conformational space was explored for the first (most

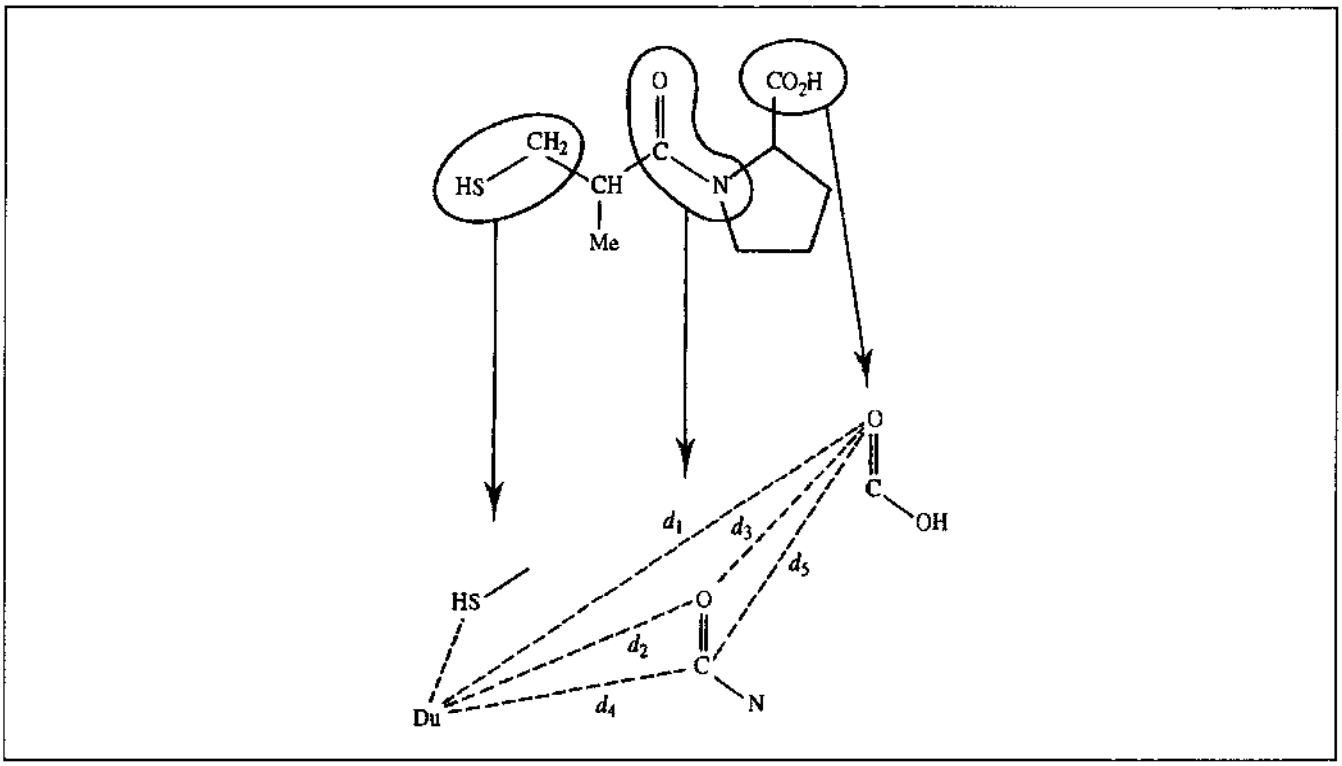


Fig 12 8. Four points and five distances define the ACE pharmacophore.

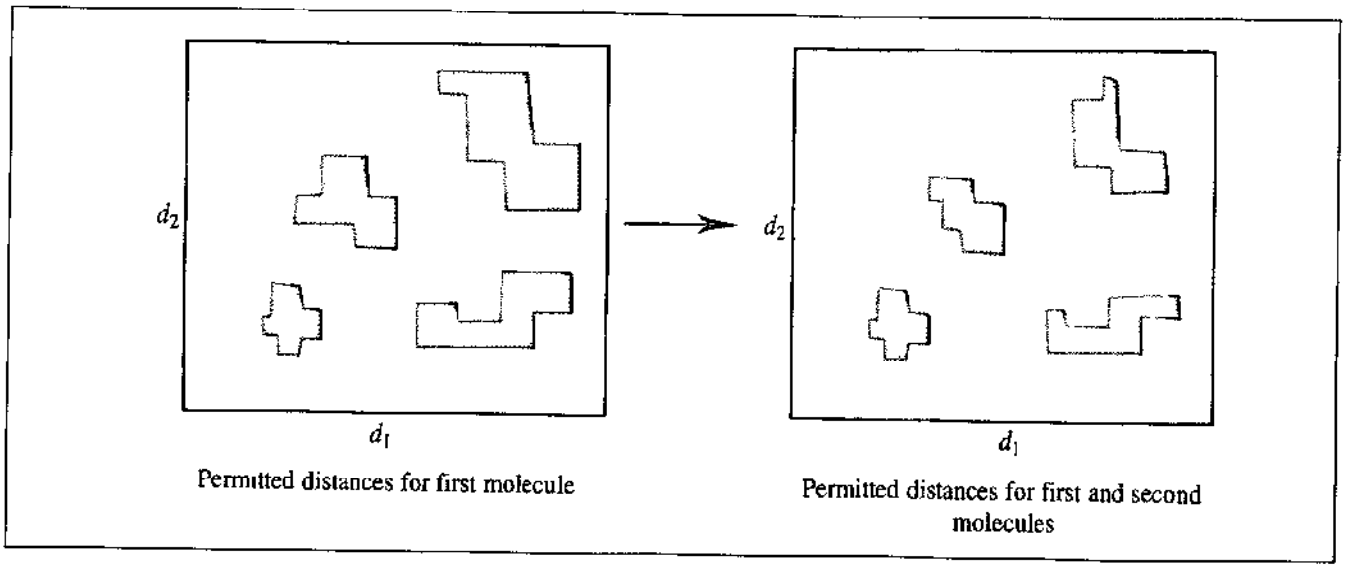


Fig 12 9 A distance map indicates the distances available to specified groups. As more molecules are considered, the permitted regions get smaller

inflexible) molecule. For each conformation, a point was registered in a five-dimensional hyperspace that corresponded to that particular combination of the five distances. When the second molecule was considered, only those torsion angles that would enable these distances to be achieved were permitted to the rotatable bonds. As more molecules were examined, so the common regions in the five-dimensional hypersurface were reduced, as illustrated schematically for a two-dimensional example in Figure 12.9. Two distinct 3D pharmacophores were obtained from the search, shown in Figure 12 10. The constrained search can be performed three orders of magnitude faster than the approach involving a separate systematic search on all the molecules.

12.4.2 Ensemble Distance Geometry, Ensemble Molecular Dynamics and Genetic Algorithms

A variant of distance geometry called *ensemble distance geometry* [Sheridan *et al.* 1986] can be used to simultaneously derive a set of conformations with a previously defined set of pharmacophoric groups overlaid. Ensemble distance geometry uses the same steps as

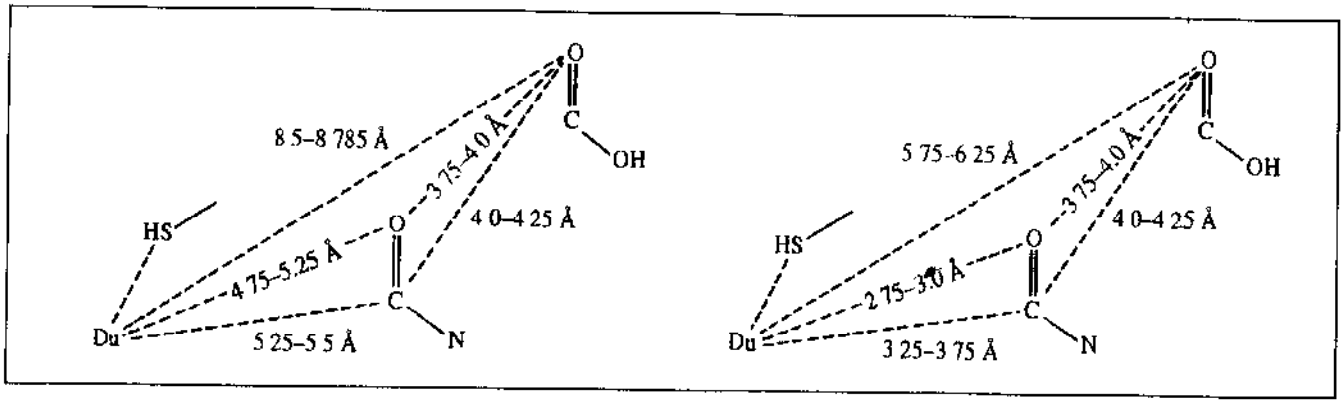


Fig 12 10 Two ACE pharmacophores identified by the constrained systematic search [Dammkoehler et al *1989].*

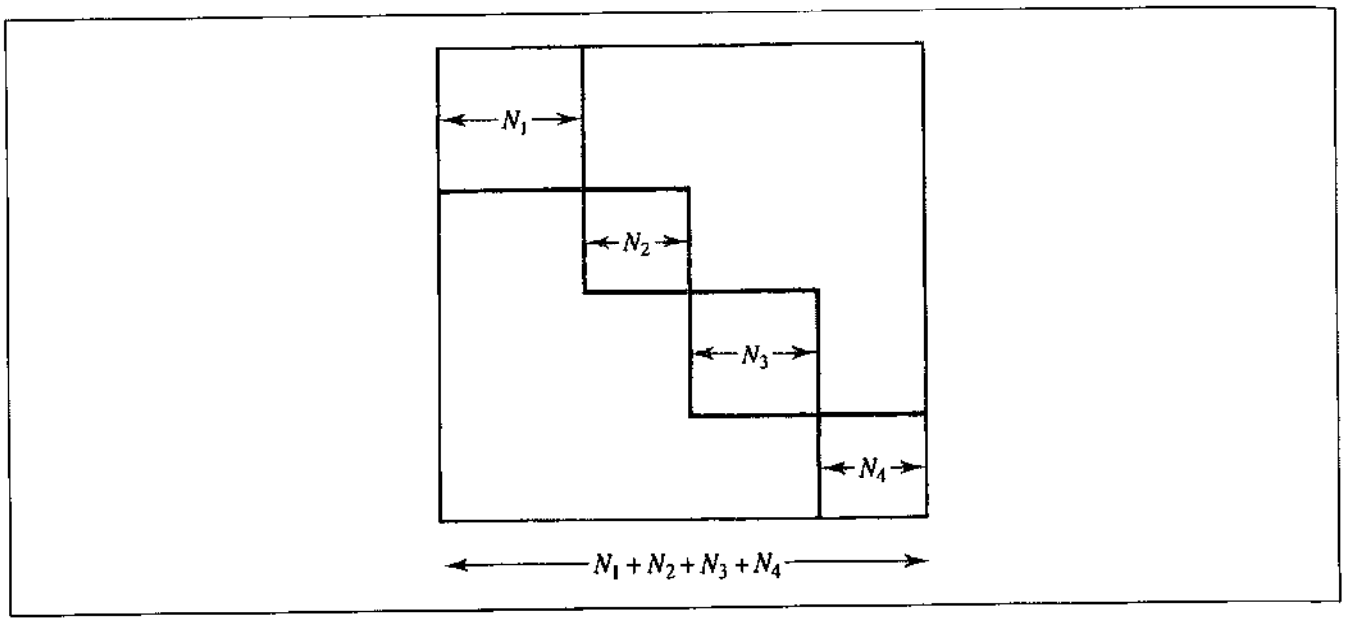


Fig 12.11· Distance matrix used in ensemble distance geometry. There are N_1 atoms in the first molecule, N_2 in the second, and so on.

standard distance geometry, with the special feature that the conformational spaces of all the molecules are considered simultaneously. This is done using much larger bounds and distance matrices, with dimensions equal to the sum of the atoms in all the molecules. In these matrices, elements 1 to N_1 correspond to the N_1 atoms of molecule 1, elements $N_1 + 1$ to $N_1 + N_2$ to the N_2 atoms of molecule 2, and so on (Figure 12 11). Elements (i, j) and (j, i) of the bounds matrix thus represent the upper and lower bounds between atoms i and j (which may or may not be in the same molecule). The upper and lower bound distances between two atoms that are in the same molecule are set in the usual way. The lower bounds for atoms that are in different molecules are set to zero. This enables the molecules to be overlaid in three-dimensional space. The upper bounds for pairs of atoms that are in different molecules are set to a large value, except for those atoms that need to be superimposed in the pharmacophore, which are set to a small tolerance parameter. Having defined the bounds matrix, the usual distance geometry steps are followed: smoothing, assignment of random distances, and optimisation against the initial bounds.

The first application of ensemble distance geometry was to derive a model of the nicotinic pharmacophore using the four nicotinic agonists shown in Figure 12.12. Three sets of atoms were selected as the pharmacophoric groups labelled A, B and C in Figure 12.12. The ensemble distance geometry algorithm generated several different solutions, but after eliminating those that contained distorted bond lengths or angles or unfavourable van der Waals contacts the remaining solutions corresponded to a single pharmacophore. This pharmacophore can be represented as a triangle (Figure 12.8). Note that the B–C distance is fixed at the length of the C=O bond. Also note that in (−)-nicotine the centroid of the pyridine ring is defined as one of the pharmacophoric points. The pharmacophore was then validated by confirming that low-energy conformations could be generated for other known nicotinic agonists that were consistent with the distance constraints of the pharmacophore.

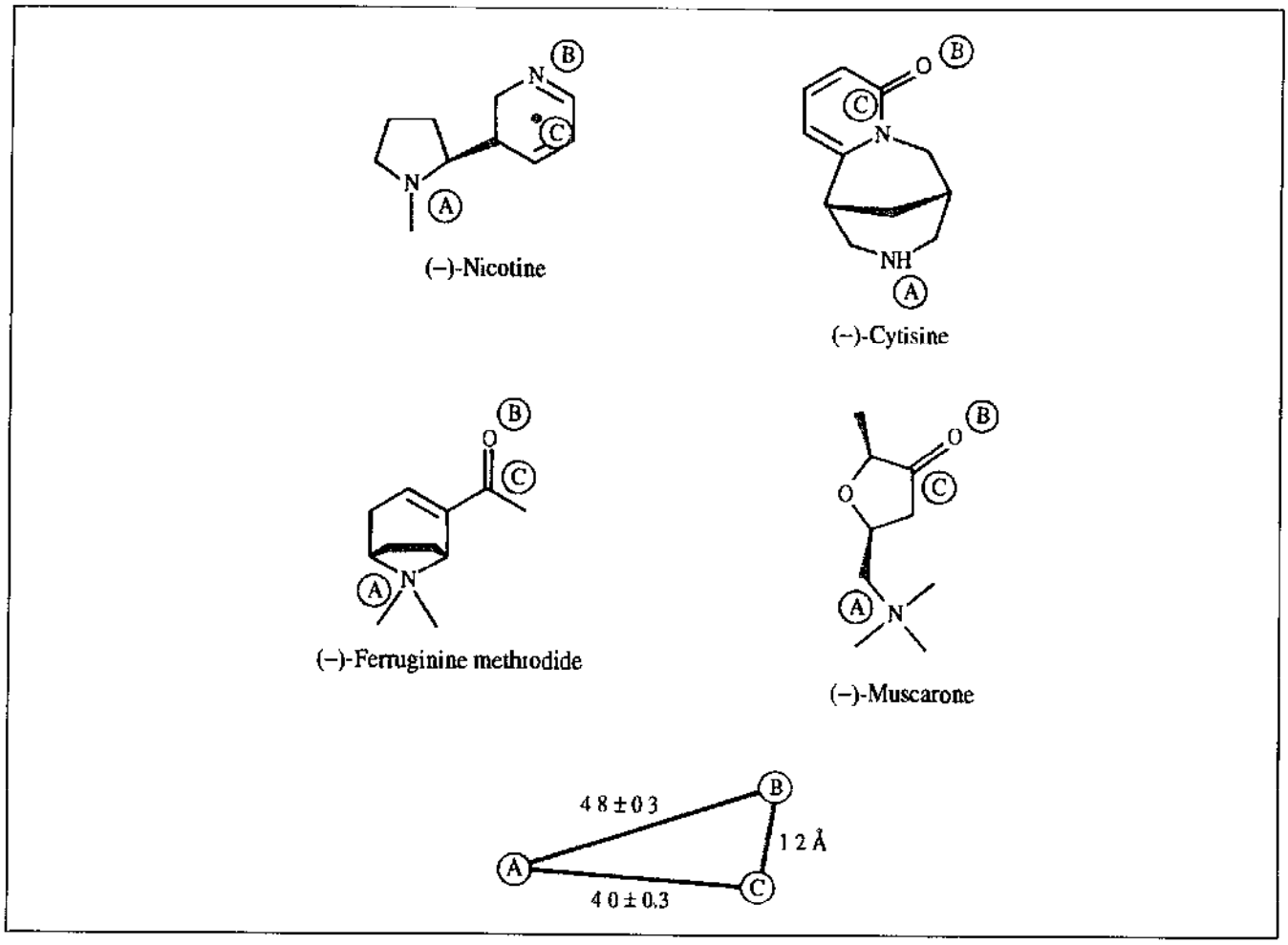


Fig. 12.12· Four molecules used to derive the nicotinic pharmacophore by distance geometry and the pharmacophore obtained

In a related way, *ensemble molecular dynamics* derives a pharmacophore using restrained molecular dynamics for a collection of molecules. A force field model is set up so that none of the atoms in each molecule 'sees' the atoms in any other molecule. This enables the molecules to be overlaid in space. A restraint term is included in the potential, which forces the appropriate atoms or functional groups to be overlaid in space.

In the ensemble distance geometry and ensemble molecular dynamics methods together with the constrained systematic search it is necessary to provide the sets of matching atoms. These are then used to constrain the conformational search space. By contrast, the genetic algorithm method [Jones *et al.* 1995a] explores not only the conformational degrees of freedom of the various molecules but also the possible feature matches. These are thus all encoded within the chromosome. A standard genetic algorithm (see Section 9.9.1) is then applied to generate possible pharmacophores.

12.4.3 Clique Detection Methods for Finding Pharmacophores

When many pharmacophoric groups are present in the molecule it may be very difficult to identify all possible combinations of the functional groups (there may be thousands of

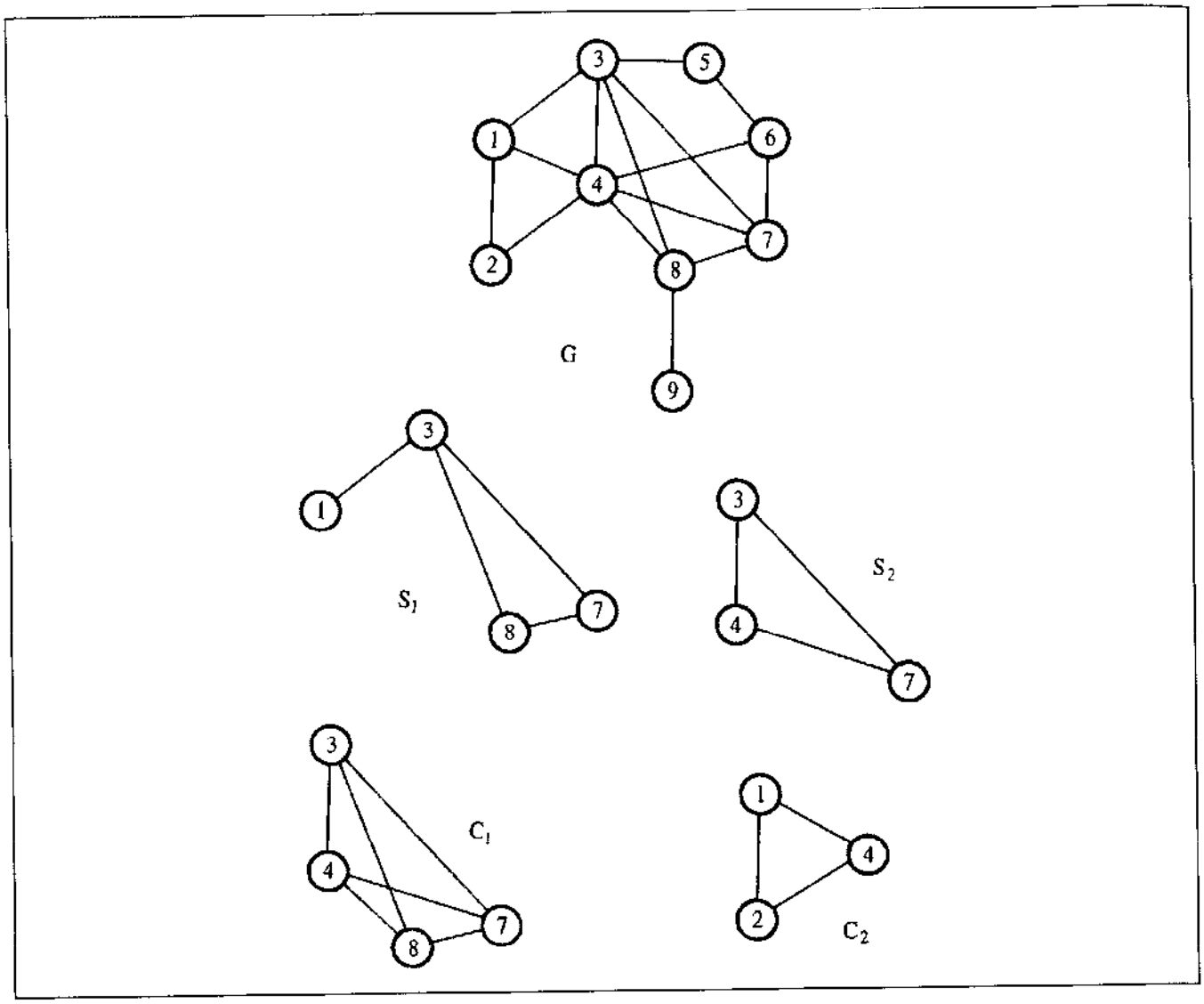


Fig 12.13: Identifying cliques in a graph.

possible pharmacophores). To tackle this problem, *clique detection* algorithms can be applied to a set of precalculated conformations of the molecules. Cliques are based upon the graph-theoretical approach to molecular structure that we discussed above.

A clique is defined as a 'maximal completely connected subgraph'. This definition is best understood by considering a simple example. Consider the graph G in Figure 12.13, together with various subgraphs. G is not a completely connected graph, because there is not an edge between all the nodes. The subgraph S_1 is not a completely connected subgraph, because there is no edge between nodes 1 and 8. The subgraph S_2 is a completely connected subgraph, because there are edges between all the nodes. However, S_2 is not a clique, because it is not a maximal completely connected subgraph; it is possible to add node 8 in order to obtain the clique C_1. A graph may contain many cliques; thus Figure 12.13 also shows a second clique, C_2. Finding the cliques in a graph belongs to a class of problems that are known as NP-complete. This means that the computational time required to find an exact solution increases in an exponential fashion with the size of the problem. Many algorithms

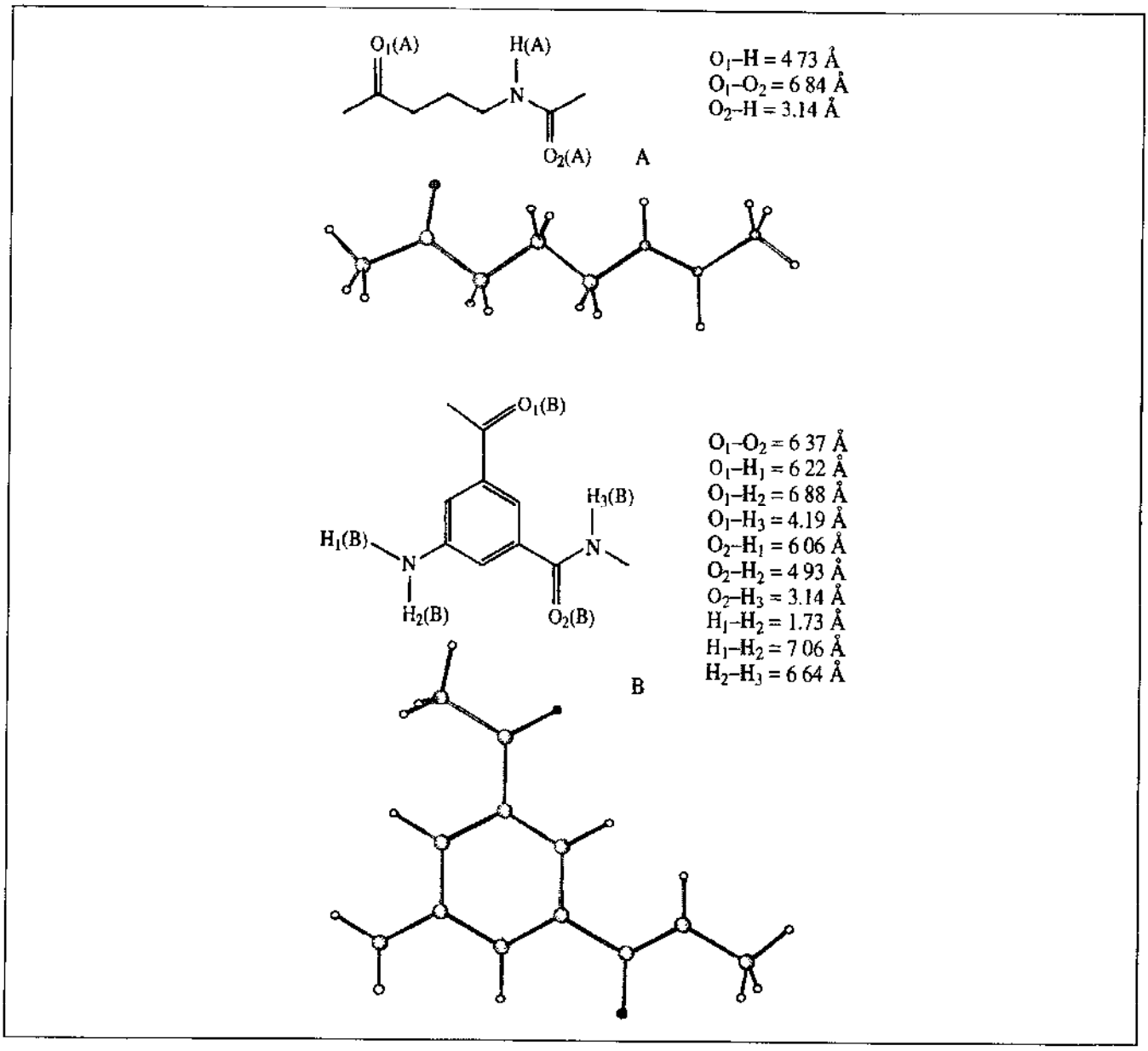


Fig 12.14: Two molecules used to illustrate clique detection

have been devised for finding cliques; the method of Bron and Kerbosch has been found to be suitably efficient for pharmacophore identification [Bron and Kerbosch 1973].

How is clique detection related to the identification of pharmacophores [Martin *et al.* 1993]? Let us suppose that we are comparing two conformations of two molecules, A and B (Figure 12.14). We construct a graph in which there is a node for every pair of matching pharmacophoric groups in the two structures. The two hydrogen-bond acceptors in molecule A (O_1(A) and O_2(A)) and the two in molecule B (O_1(B) and O_2(B)) give rise to four nodes in the joint graph. There is one hydrogen-bond donor in molecule A (H(A)) but three in molecule B (H_1(B), H_2(B) and H_3(B)), giving rise to three nodes in the graph. The intramolecular distances between these groups are indicated in Figure 12.14 An edge is drawn between each pair of nodes when the distance between the corresponding groups in the two

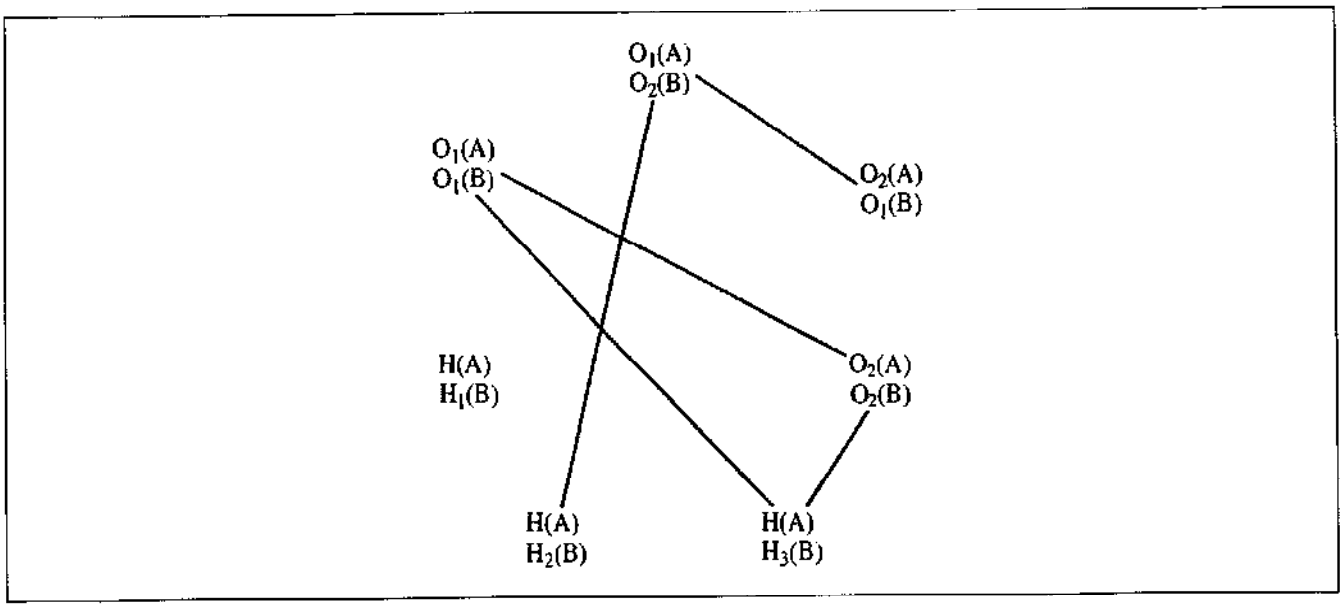


Fig. 12.15· The matching graph for the two molecules in Figure 12.14.

molecules is the same, within some tolerance. For example, the distance between $O_1(A)$ and $O_2(A)$ is 4.73 Å and the distance between $O_1(B)$ and $O_2(B)$ is 4.19 Å. If the tolerance is at least 0.54 Å then the two distances would be considered equal and so an edge is drawn between the corresponding pairs of nodes in the graph, as shown in Figure 12.15. The full graph is shown in Figure 12.15, where we have assumed a tolerance of 0.6 Å. Clique detection is used to find maximal sets of matching groups for the two molecules; in this simple example there are three cliques, two containing just two overlapping atoms and one containing three matching atoms (Table 12.1)

In the clique detection approach, the first step is to generate a family of low-energy conformations for the molecules. The molecule with the smallest number of conformations is used as the starting point, with each of its conformations being used in turn as the reference structure. Each conformation of every other molecule is then compared with the reference conformations and the cliques identified. The cliques for each molecule are obtained by combining the results for each of its conformations. Those cliques that are common to at least one conformation from each molecule can then be combined to give a possible 3D pharmacophore for the entire set.

Clique number	Atom from A	Atom from B
1	O_1	O_2
	H	H_2
2	O_1	O_2
	O_2	O_1
3	O_1	O_1
	O_2	O_2
	H	H_3

Table 12 1: Cliques found when matching the molecules in Figure 12.14

12.4.4 Maximum Likelihood Method

One limitation of clique detection is that it needs to be run repeatedly with different reference conformations and the run-time scales with the number of conformations per molecule. The maximum likelihood method [Barnum *et al.* 1996] eliminates the need for a reference conformation, effectively enabling every conformation of every molecule to act as the reference. Despite this, the algorithm scales linearly with the number of conformations per molecule, so enabling a larger number of conformations (up to a few hundred) to be handled. In addition, the method scores each of the possible pharmacophores based upon the extent to which it fits the set of input molecules and an estimate of its 'rarity'. It is not required that every molecule has to be able to match every feature for the pharmacophore to be considered.

Prior to the pharmacophore identification phase, a set of conformations is generated for each molecule. Typically, the 'poling' method (see Section 9.14) is used to produce a reasonably small but representative set of low-energy conformations. First, all possible combinations of pharmacophore features (e.g. donor–donor–acceptor, aromatic ring–donor–acceptor–hydrophobic region) are exhaustively considered. Possible geometric arrangements of the features in 3D space are identified by taking each molecule to be the reference structure and examining its conformations. These configurations are scored and ranked according to how well they describe the set of active molecules. Each configuration is considered an 'hypothesis' which can be used to assign a probability that a molecule is active depending on whether or not it matches the pharmacophore. If there are K features in the pharmacophore then the algorithm defines $K+2$ possible ways in which a molecule can match the pharmacophore on a scale from 0 to $K+1$. If the molecule matches all K features it is placed in the class $x = K+1$. If it does not match all K features or any of the subsets obtained by removing one of the features (there are K such subsets, each containing $K-1$ features) then the molecule is placed in the class $x = 0$. A molecule which can match one of the subsets with $K-1$ features is assigned to an intermediate class between $x = 1$ and $x = K$, depending upon which of the subsets (ordered by selectivity) it matches. Both the full match and each of the K partial matches are assigned a 'rarity value' depending upon the type of the features present and on their relative disposition. Pharmacophores that contain 'rare' features (such as positively ionisable groups) are scored more highly than those that contain more common features (such as hydrophobic regions). In addition, the greater the distribution of the features (as measured by the squared distance between the feature and the common centroid) the higher the score.

This rarity value is equated with the fraction of hits that would be returned by searching a large database of diverse molecules with the full pharmacophore (all K features) or the subset (with $K-1$ features) as appropriate. Labelling this fraction of hits as $p(x)$ we now define $q(x)$ as the fraction of the M active molecules (i.e. the molecules originally supplied as input to the procedure) which match each of the $K+1$ possible classes. The overall configuration is scored using:

$$\text{score} = M \sum_x q(x) \log_2 \left(\frac{q(x)}{p(x)} \right) \tag{12.1}$$

Higher scores are achieved by pharmacophores that have large values of $q(x)$ (i.e. are matched by more of the active molecules in the initial set) and lower values of $p(x)$ (i.e. it is less likely that a 'random' molecule would match). The actual numerical values $p(x)$ are obtained from a predefined regression equation.

The pharmacophores generated by this approach are typically expressed in terms of 'location constraints' rather than inter-feature distance ranges [Greene *et al.* 1994]. These location constraints are typically expressed as a point in 3D space surrounded by a spherical region. A molecule must be able to place the relevant features within the appropriate spheres (see Figure 12.16, colour plate section). The spheres can be of different sizes to reflect the fact that the various interactions have differing sensitivities to changes in distance. For example, the tolerance on a charge interaction might be smaller than on a hydrogen-bonding interaction to reflect the fact that the energy of an ionic interaction is more sensitive to changes in relative position than for the hydrogen bond. Another useful aspect of this approach is its requirement that the features must be 'surface accessible' in the molecule for it to be considered a match.

12.4.5 Incorporating Additional Geometric Features Into a 3D Pharmacophore

The features used to define a 3D pharmacophore are most easily derived from the positions of specific atoms within each molecule It may be more appropriate to consider locations around the molecule where the receptor might position its functional groups. This is especially relevant for hydrogen-bond donors and acceptors; two ligands may be able to hydrogen bond to the same protein atom with the ligand atoms being in a completely different location in the binding site, as illustrated in Figure 12.17. 3D pharmacophores may also be defined in terms of specific geometrical relationships between the pharmacophoric groups, such as the angle between the planes of two aromatic rings. The 3D pharmacophore may also contain features that are designed to mimic the presence of the receptor. These are commonly represented as *exclusion spheres*, which indicate locations within the 3D pharmacophore where no part of a ligand is permitted to be positioned. Some of these additional features are illustrated in Figure 12.18.

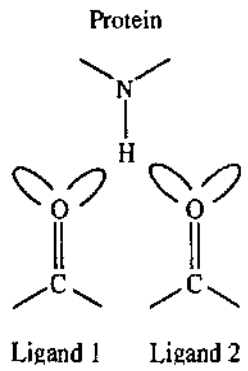


Fig 12 17: Two ligands may be able to position a hydrogen-bond acceptor in different locations in space yet still interact with the same hydrogen-bond donor

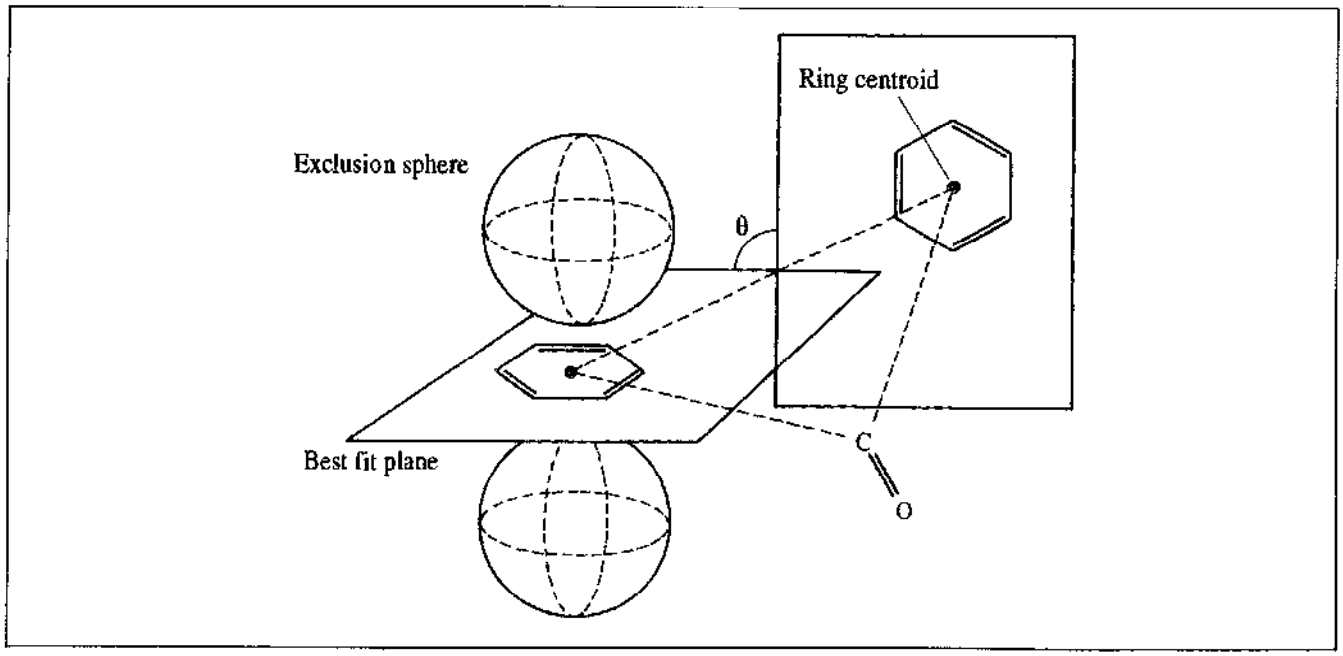


Fig 12 18: Features that can be incorporated into 3D pharmacophores

12.5 Sources of Data for 3D Databases

In a 3D database search one needs to consider the three-dimensional structures of the molecules. Where does such structural information come from? An obvious source is the Cambridge Structural Database, which contains experimental X-ray structures of more than 150 000 compounds. However, for most of the compounds in a typical compound database no crystal structure is available. Structure generation programs are designed to produce one or more low-energy conformations solely from the molecular graph. As the number of compounds may be very large, such programs must be able to operate automatically, rapidly and with little or no user intervention (i.e. without crashing!). The two most widely used structure generators to date are the CONCORD [Rusinko *et al.* 1988] and CORINA programs [Gasteiger *et al.* 1990], which both use a knowledge-based approach combined with energy minimisation.

Most structure generation algorithms produce only a single conformation for each molecule. With the possible exception of wholly rigid molecules, there is no guarantee that this structure corresponds to the conformation adopted when the molecule binds. We therefore need some way to take conformational flexibility into account during the 3D database search. The simplest way to do this is to store information about many conformations. To store conformations explicitly would usually require a large amount of disk space and so the information is usually compressed into a more compact form. To make the 3D search more efficient screening methods are commonly used, similar to those employed for '2D' substructure searches. One straightforward way to do this is to derive a set of distance 'keys' for the pairs of pharmacophoric groups in the molecule. Each key is a binary number in which each bit corresponds to a distance range between the appropriate pair of groups (donor–donor, donor–acceptor, etc.). For example, the first bit could correspond

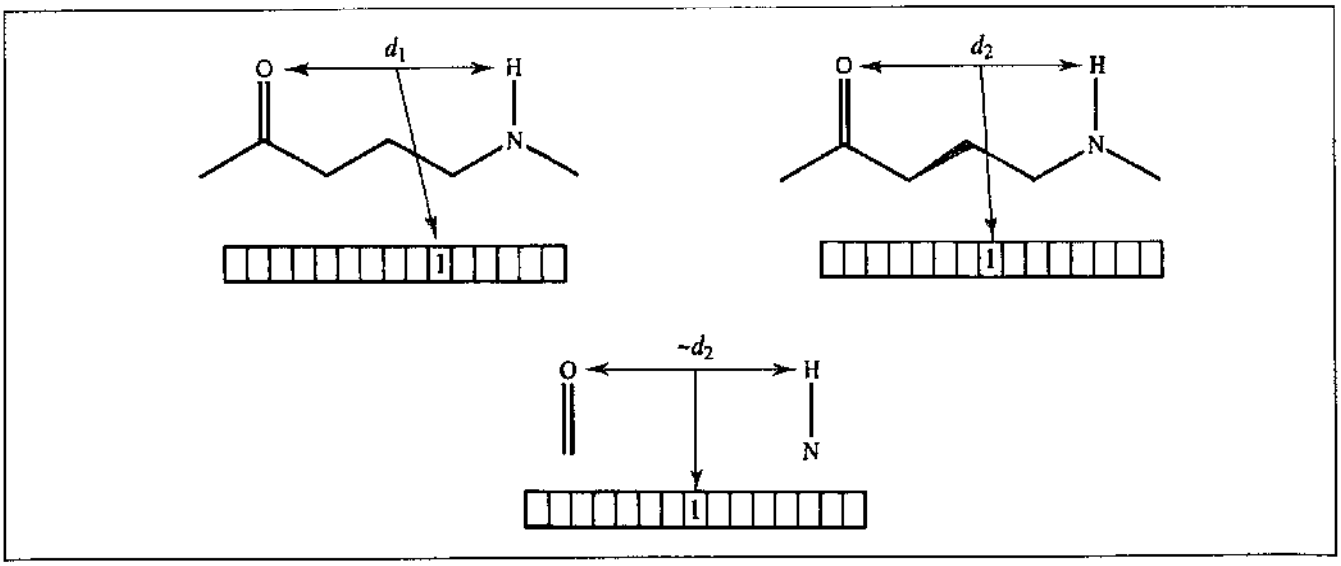


Fig. 12.19: 3D database searching. As each conformation is generated an appropriate bit is set in the binary key. At search time, the binary key appropriate to the pharmacophore is set up and compared with the keys in the database.

to a distance in the range 2.0–2.5 Å, the second bit to the range 2.5–3.0 Å, and so on. In fact, it is more efficient to use smaller bin sizes for the more common distances and larger bins for less common distances, because the distribution of distances between such groups in molecules is not uniform. The key initially contains all zeros. As each conformation is generated, the distances between the pharmacophoric groups are calculated and the appropriate bits in the relevant keys are changed to 1s (Figure 12.19). To search the database, the keys corresponding to the pharmacophore are calculated. The pharmacophore's keys are then compared with each molecular key, so identifying all molecules which could match the pharmacophore. Separate substructure-like screens are also used; these contain information about the number of each type of feature (e.g. number of donors). If the molecule does not contain the minimal number of groups in the pharmacophore, then it obviously cannot match and so can be discarded before its conformational properties need to be considered.

An alternative strategy is to explore the conformational space for each molecule during the database search. Systems which employ such an approach rely heavily upon screens which identify and reject molecules that could not satisfy the requirements of the pharmacophore before their conformational space is explored. These screens can be determined solely from the molecular graph and are typically represented as distance ranges. Triangle smoothing (Section 9.5) is one way in which such distance screens can be calculated; it provides the upper and lower bounds on interatomic distances. However, the distance ranges provided by triangle smoothing can be much wider than the actual distances that are observed in real structures. A simple example suffices to illustrate this point: the distance obtained when triangle smoothing is used to calculate the lower bound distance between the amide nitrogen and the carbonyl oxygen of the carboxylic acid group in 4-acetamido benzoic acid (Figure 12.20) is equal to the sum of the van der Waals radii (approximately 3.3 Å, depending upon the van der Waals radii used), compared to a distance of about 6.4 Å in all the accessible conformations of this molecule.

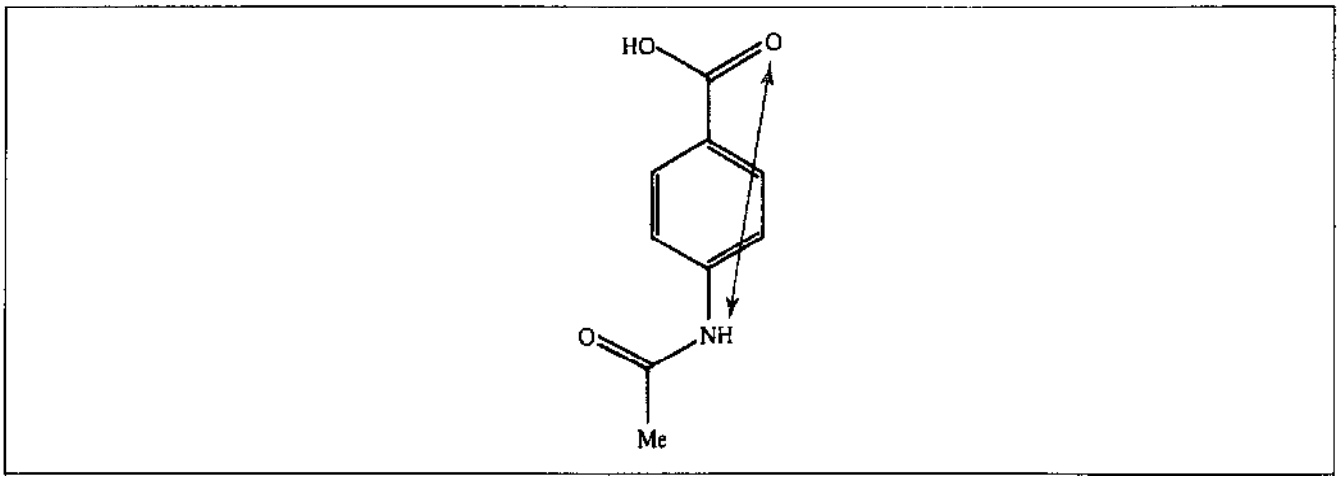


Fig. 12 20 4-Acetamido benzoic acid Triangle smoothing predicts that the lower bound distance between the amide nitrogen and the carbonyl oxygen is equal to the sum of the van der Waals radii. The actual distance is about 6 4 Å.

Having eliminated those molecules that could not possibly satisfy the geometric and chemical requirements of the pharmacophore, the program must explore the conformational degrees of freedom of the molecules that remain. This is done using methods to rapidly identify one or more conformations which satisfy the constraints of the pharmacophore. A natural method to use would be distance geometry, in which the pharmacophoric constraints would be incorporated into the bounds matrix, thereby leading to the generation of conformations that satisfy the constraints. However, distance geometry is rather too slow for this purpose. An alternative strategy is to 'adjust' or 'tweak' the conformation by rotating about single bonds, to force it to fit the pharmacophore Adjustment is usually performed in torsional space (i.e. only the torsion angles are varied) by minimising an appropriate potential function expressed in terms of distances.

12.6 Molecular Docking

In molecular docking, we attempt to predict the structure (or structures) of the intermolecular complex formed between two or more molecules. Docking is widely used to suggest the binding modes of protein inhibitors. Most docking algorithms are able to generate a large number of possible structures, and so they also require a means to score each structure to identify those of most interest. The 'docking problem' is thus concerned with the generation and evaluation of plausible structures of intermolecular complexes [Blaney and Dixon 1993].

The docking problem involves many degrees of freedom. There are six degrees of translational and rotational freedom of one molecule relative to the other as well as the conformational degrees of freedom of each molecule. The docking problem can be tackled manually, using interactive computer graphics. This 'hands-on' approach can be very effective if we have a good idea of the expected binding mode, for example because we already know the binding mode of a closely related ligand. However, even in such cases one must be wary; X-ray crystallographic experiments have revealed that even very similar inhibitors may adopt

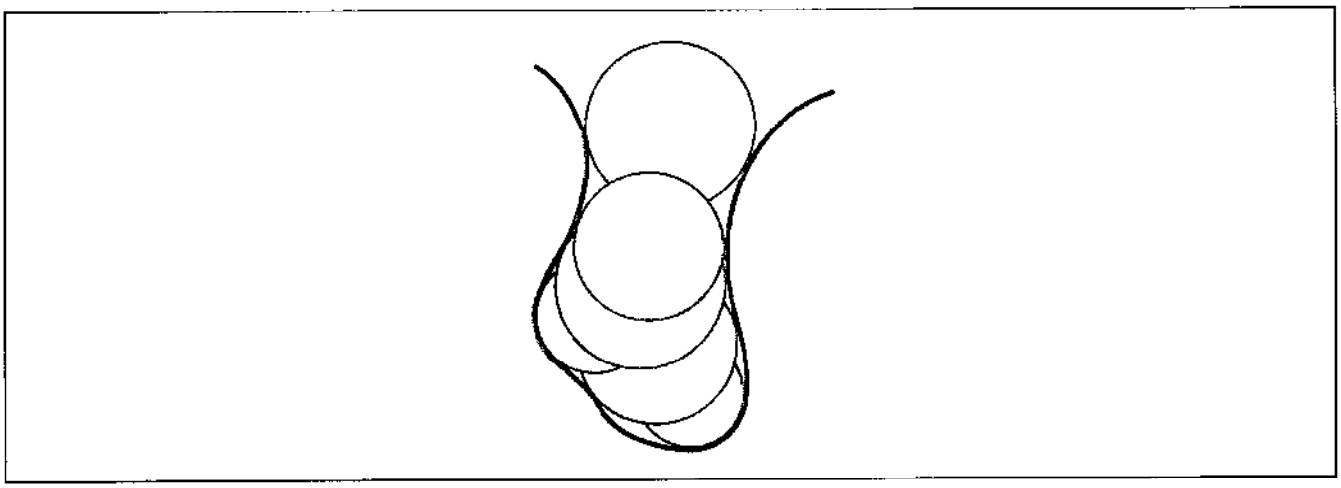


Fig. 12.21: A binding site represented as a collection of overlapping spheres.

quite different binding modes. Automatic docking algorithms can be less biased than human modellers and usually consider many more possibilities.

Various algorithms have been developed to tackle the docking problem. These can be characterised according to the number of degrees of freedom that they ignore. Thus, the simplest algorithms treat the two molecules as rigid bodies and explore only the six degrees of translational and rotational freedom. The earliest algorithms for docking small molecule ligands into the binding sites of proteins and DNA used this approximation. A well-known example of such an algorithm is the DOCK program of Kuntz and co-workers [Kuntz *et al.* 1982]. DOCK is designed to find molecules with a high degree of shape complementarity to the binding site. The program first derives a 'negative image' of the binding site from the molecular surface of the macromolecule This negative image consists of a collection of overlapping spheres of varying radii, each of which touches the molecular surface at just two points, as shown schematically in Figure 12.21. Ligand atoms are then matched to the sphere centres to find matching sets (cliques) in which all the distances between the ligand atoms in the set are equal to the corresponding sphere centre–sphere centre distances (within some user-specified tolerance). The ligand can then be oriented within the site by performing a least-squares fit of the atoms to the sphere centres, as shown in Figure 12.22. The orientation is checked to ensure there are no unacceptable steric interactions between the ligand and the receptor. If the orientation is acceptable then an interaction energy is computed to give the 'score' for that binding mode. New orientations are generated by matching different sets of atoms and sphere centres. The top-scoring orientations are retained for subsequent analysis.

To perform conformationally flexible docking the conformational degrees of freedom need to be taken into account. Most of the methods that attempt to include the conformational degrees of freedom only consider the conformational space of the ligand; the receptor is invariably assumed to be rigid. All of the common methods for searching conformational space have been incorporated at some stage into a docking algorithm. For example, Monte Carlo methods have been used to perform molecular docking, often in conjunction with simulated annealing [Goodsell and Olson 1990]. At each iteration of the Monte Carlo procedure the internal conformation of the ligand is changed (by rotating about a bond) or the entire molecule is randomly translated or rotated The energy of the ligand within

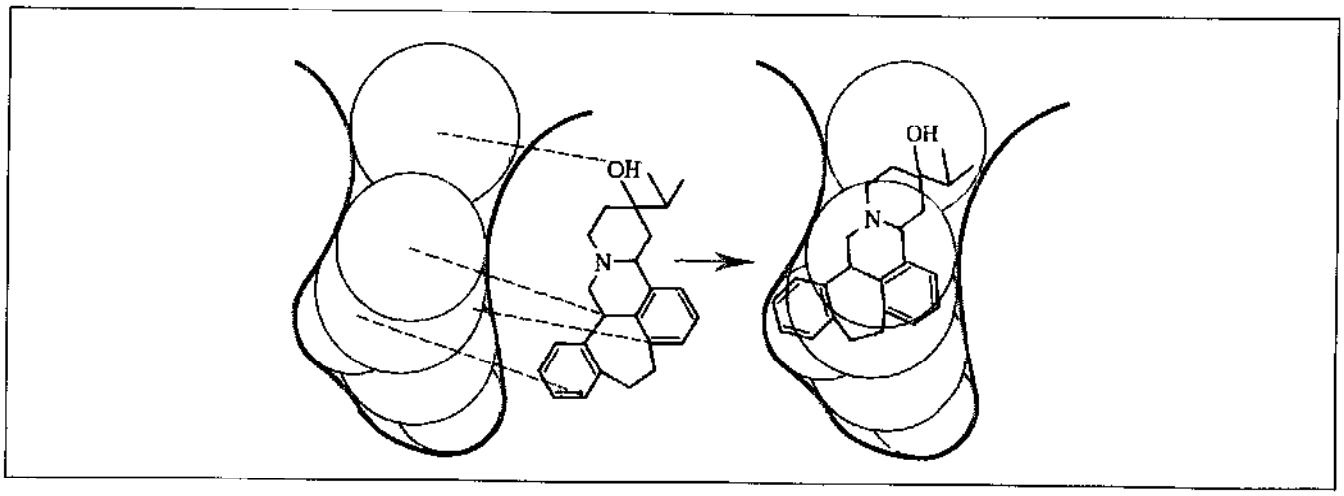


Fig 12 22: The DOCK algorithm [Kuntz et al. *1982]. Atoms are matched to sphere centres and then the molecule is positioned within the binding site*

the binding site is calculated using molecular mechanics and the move is then accepted or rejected using the standard Metropolis criterion. An interesting variant on the basic Monte Carlo approach is the tabu search [Baxter *et al.* 1998]. This maintains a record of those regions of the search space that have already been visited, so ensuring that the method is encouraged to explore more of the binding site.

Genetic algorithms can also be used to perform molecular docking [Judson *et al.* 1994; Jones *et al.* 1995b; Oshiro *et al.* 1995]. Each chromosome codes not only for the internal conformation of the ligand as described in Section 9.9.1 but also for the orientation of the ligand within the receptor site. Both the orientation and the internal conformation will thus vary as the populations evolve. The score of each docked structure within the site acts as the fitness function used to select the individuals for the next iteration.

Distance geometry can be used to perform molecular docking. The major problem to be addressed with this method is to find a way to generate conformations of the ligand within the binding site. One way to achieve this is by using a modified penalty function that forces the ligand conformation to remain within the binding site. For example, an additional penalty term can be added which has the effect of forcing the ligand to lie in the DOCK-derived cluster of spheres that represents the binding site

An approach that is used by a number of programs involves the incremental construction of the ligand [Leach and Kuntz 1990; Welch *et al.* 1996; Rarey *et al.* 1996]. This is similar in spirit to the depth-first systematic conformational search described in Section 9.2. The main difference, of course, is that in docking the conformational search is performed within the binding site. A typical incremental construction algorithm first identifies one or more 'base fragments' within the ligand These base fragments are often chosen to be a reasonably significant, fairly rigid part of the molecule such as a ring system. The base fragment(s) are docked into the binding site and may then be clustered to remove similar orientations. Each docked orientation of the base fragment(s) then represents the starting point for the conformational analysis of the rest of the ligand. One might anticipate that such an approach would be very time-consuming, as it is in effect necessary to perform the conformational

analysis for each orientation of the base fragments. However, it is often found that the protein provides a particularly useful constraint, enabling the search tree to be pruned very effectively.

The ideal docking method would allow both ligand and receptor to explore their conformational degrees of freedom. Perhaps the most 'natural' way to incorporate the flexibility of the binding site is via a molecular dynamics simulation of the ligand–receptor complex. However, such calculations are computationally very demanding and are in practice only useful for refining structures produced using other docking methods; molecular dynamics does not explore the range of binding modes very well except for very small, mobile ligands. For many systems, the energy barriers that separate one binding mode from another are often too large to be overcome. Some other attempts have been made to incorporate protein flexibility (at least at the level of the side chains [Leach 1994]) but these methods are generally in their infancy and take much longer than rigid-protein docking.

When the first docking methods were developed the speed of the typical computer was such that only rigid-body docking of single molecules was feasible. As computational performance increased it was recognised that rigid-body docking could be used to examine large numbers of molecules from a database. At approximately the same time, algorithms that addressed the conformational flexibility of the ligand were devised. With the passage of time, it is now possible to search databases using a flexible-ligand algorithm. However, there is still a clear distinction between the use of docking to predict the binding mode of a single active molecule, where one can afford to use a particularly thorough search, and the use of docking for searching databases for possible lead compounds.

12.6.1 Scoring Functions for Molecular Docking

Most docking algorithms are capable of generating a large number of potential solutions. Some of these can be rejected immediately because they have a high-energy clash with the protein. The remainder must be assessed using some scoring function. When we are only interested in how a single ligand binds to the protein then the scoring function need only be able to identify the docked orientation that most closely corresponds to the 'true' structure of the intermolecular complex. However, when docking a database of molecules then not only should the scoring function be able to identify the 'true' docking mode of a given ligand but it also needs to be able to rank one ligand relative to another. Moreover, the large number of orientations that may be generated during a docking run means that it must be possible to calculate the scoring function rapidly.

Many of the scoring functions in common use attempt to approximate the binding free energy for the ligand binding to the receptor We have previously encountered a number of ways in which simulation techniques can be used to predict (relative) free energies of binding (see Chapter 11), but these are far too slow to be of value in docking calculations. Faster, more approximate methods tend to be used. In contrast to the free energy perturbation approach these alternatives tend to consider that the free energy of binding can be written as an additive equation of various components to reflect the various contributions to binding [Bohm and Klebe 1996]. A complete equation of this kind would have the

following contributions [Ajay and Murcko 1995]:

$$\Delta G_{bind} = \Delta G_{solvent} + \Delta G_{conf} + \Delta G_{int} + \Delta G_{rot} + \Delta G_{t/r} + \Delta G_{vib} \quad (12.2)$$

where $\Delta G_{solvent}$ is the contribution due to solvent effects, arising from the balance of interactions between the solvent and the ligand, protein and intermolecular complex. Various methods can be used to determine these contributions. ΔG_{conf} arises from conformational changes in the protein and in the ligand. In many cases, the protein does not change much on binding (which is fortunate, given that most docking methods assume a rigid receptor). By contrast, the ligand changes from an ensemble of conformations in solution to what is often assumed to be a single dominant conformation in the bound state. Various analyses have been performed to try to determine the size of this energetic penalty for the ligand. When measured relative to the most significant conformation in solution, an average penalty of 3 kcal/mol was found [Bostrom *et al.* 1998]. ΔG_{int} is the free energy due to specific protein–ligand interactions. ΔG_{rot} is the free energy loss associated with freezing internal rotations of the protein and the ligand. This is mostly due to the entropic contribution. The simplest way to calculate this penalty is to assume that there are three states per rotatable bond (trans and ±gauche) of equal energy, thus leading to a free energy loss of $RT \ln 3$ ($\sim$ 0.7 kcal/mol) per rotatable bond. $\Delta G_{t/r}$ is the loss in translational and rotational free energy caused by the association of two bodies (the ligand and the receptor) to give a single body (the intermolecular complex). This is often assumed to be constant for all ligands and so is ignored when one is interested in the relative binding strengths of different ligands. ΔG_{vib} is the free energy due to changes in vibrational modes. This contribution is difficult to calculate and is usually ignored.

Each of the terms in Equation (12.2) has been the subject of considerable discussion in the literature, and for some of them there may be a number of different approaches to their estimation. However, many of these methods are unsuitable for docking, due to the calculation time required. Some very simple functions have been employed for docking, such as that originally used in the DOCK program (illustrated in Figure 12.23 together with another similar form, the piecewise linear potential [Gelhaar *et al.* 1995]). Despite their apparent simplicity, such functions continue to rate well in comparisons of different functional

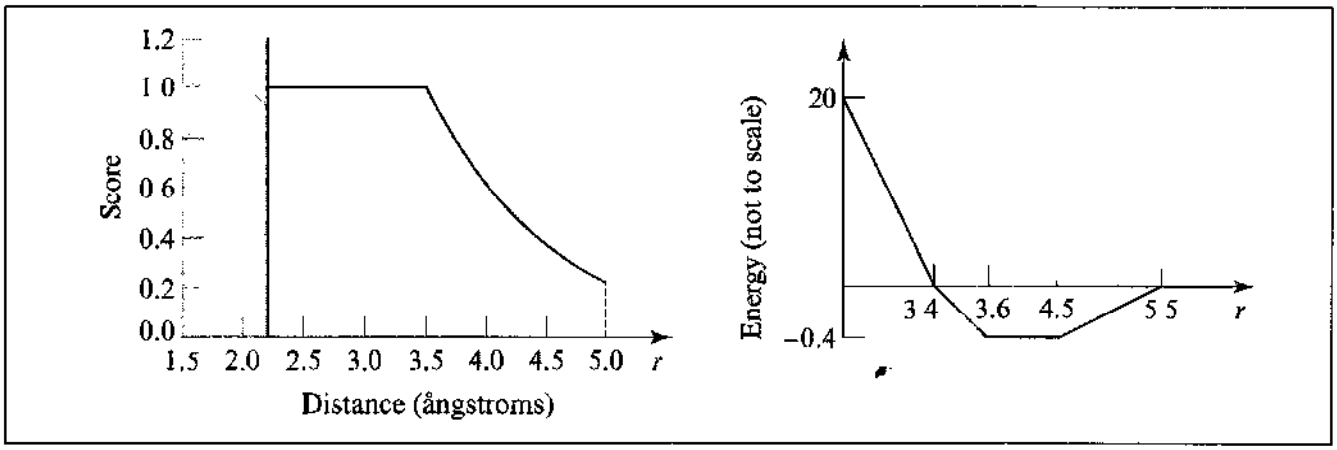


Fig. 12.23 Two simple scoring functions used in docking. On the left is the basic scoring scheme used by the DOCK program [Desjarlais et al *1988] On the right is the 'piecewise linear potential' [Gelhaar* et al *1995]*

forms. Molecular mechanics is also widely used to calculate the energy of interaction; one way in which such a calculation can be speeded up is to pre-calculate electrostatic and van der Waals 'potentials' on a regular grid which covers the binding site [Meng *et al.* 1992]. The computational effort required to calculate the energy of interaction between ligand and protein is then linear in the number of atoms in the ligand, rather than being proportional to the product of the number of ligand atoms multiplied by the number of protein atoms.

Simple molecular mechanics scoring functions are popular, but we can see from Equation (12.2) that they provide only part of the overall free energy of binding. Thus whilst they have proved successful in some cases (such as a study of HIV-protease inhibitors [Holloway *et al* 1995]), one should not be surprised if they do not always work. An interesting approach to this problem was suggested by Böhm. He tried to find a simple linear relationship between the free energy of binding and a variety of parameters which it was anticipated would be relevant to the overall free energy of binding and which could also be calculated rapidly [Böhm 1994]. The terms in this original formulation related to hydrogen bonding, ionic interactions, lipophilic interactions and the loss of internal degrees of freedom of the ligand:

$$\Delta G_{\text{bind}} = \Delta G_0 + G_{\text{hb}} \sum_{\text{h-bonds}} f(\Delta R, \Delta\alpha) + \Delta G_{\text{ionic}} \sum_{\text{ionic interactions}} f(\Delta R, \Delta\alpha) + \Delta G_{\text{lipo}} |A_{\text{lipo}}| + \Delta G_{\text{rot}} NROT \tag{12.3}$$

where ΔG_0 is a constant term, independent of the system, which was interpreted to correspond to the overall change in translational/rotational free energy ($\Delta G_{t/r}$ in Equation (12.2)). ΔG_{hb} corresponds to the contribution from an ideal hydrogen bond. This contribution is multiplied by a penalty function $f(\Delta R, \Delta\alpha)$ which accounts for large deviations of the hydrogen bond from the ideal geometry; ΔR is the deviation of the hydrogen-bond distance from its ideal value of 1.9 Å, and $\Delta\alpha$ is the deviation from the ideal angle of 180°. The same geometric dependency is applied to the ionic interactions. ΔG_{lipo} is a contribution from lipophilic interactions, which are assumed to be proportional to the lipophilic contact surface (i.e. involving non-polar atoms) between the protein and the ligand, A_{lipo}. ΔG_{rot} is the loss of free energy due to freezing a rotatable bond in the ligand upon binding. It is thus multiplied by the number of rotatable bonds in the ligand, *NROT*.

Experimental binding data on 45 protein–ligand complexes was extracted from the literature and then a multiple linear regression analysis (see Section 12.12.2) was performed to derive the parameters in this equation (i.e. the various ΔG values). The values of the parameters obtained from this analysis ($\Delta G_{\text{hb}} = -1.2$ kcal/mol, $\Delta G_{\text{ionic}} = -2.0$ kcal/mol, $\Delta G_{\text{lipo}} = -0.04$ kcal/mol Å^2, $\Delta G_{\text{rot}} = +0.3$ kcal/mol, $\Delta G_0 = +1.3$ kcal/mol) mostly correspond reasonably closely to values estimated from other approaches, with the exception of the constant term, ΔG_0, for which a value between 7 and 11 kcal/mol is generally agreed [Ajay and Murcko 1995]. The model reproduced the experimental binding data with a standard deviation of 1.7 kcal/mol. The exponential relationship between the binding free energy and the equilibrium constant means that a change of just 1.4 kcal/mol in the free energy corresponds to a ten-fold change in affinity This work has spawned a number of related studies, which differ in the terms included. For example, the surface area is commonly divided into polar and non-polar regions,

with different parameters for polar/polar, polar/non-polar and non-polar/non-polar interactions. Various statistical techniques have been used to derive the equation and various sources of data used to derive the function [Head *et al.* 1996; Böhm 1998; Eldridge *et al.* 1997]. One possible problem with such functions is that they are typically derived from ligands that bind very tightly to their receptor, whereas docking is increasingly used to identify ligands of only modest potency from a large database. For this particular problem, combining the results from more than one scoring function has been shown to give better results than just using individual scoring functions on their own, an approach referred to as 'consensus scoring' [Charifson *et al.* 1999].

12.7 Applications of 3D Database Searching and Docking

There are now a number of published studies that demonstrate the utility of 3D database searching in drug design, using both docking and pharmacophore searching. Kuntz's group has used the DOCK program against a number of targets, including HIV protease, DNA, thymidylate synthase and haemagglutinin [Kuntz 1992; Kuntz *et al.* 1994]. In each case, one or more inhibitors of modest potency were discovered. Information about hits from the first generation was then used to perform more exhaustive database searches to identify yet more potent compounds. The structures of some of the 'hits' were determined by X-ray crystallography, revealing that not all of the ligands bound in the same way as predicted by the docking algorithm. A degree of serendipity is still important even with automated docking methods. For this reason, it is important to assess the performance of any new docking methods against as many experimentally determined protein–ligand complexes as possible. The much larger number of X-ray structures now available means that it is possible to choose at least one hundred ligands that vary in size, shape, flexibility and functionality (charged, polar, hydrophobic) and which dock into many different proteins. Two good examples of the kind of analysis that is now possible are those evaluating the GOLD program [Jones *et al.* 1997] and the FlexX program [Kramer *et al.* 1999] GOLD uses a genetic algorithm, whereas FlexX uses an incremental construction method. There were some differences between the way in which each program was assessed, the most obvious approach being to calculate the RMS deviation between the theoretical and experimental structures, although this can sometimes be a rather simplistic and sometimes misleading metric. However, the best docking programs are able to get 'close' to the correct result for approximately 70% of the ligands.

Commercial 3D database systems for performing pharmacophore searches were available from the early 1990s, but it took several years for real applications to be reported in the literature, largely due to the confidential nature of many of the results. One example of a fairly typical study is that of Marriott and colleagues, who were looking for new lead molecules active against the muscarinic M_3 receptor [Marriott *et al* 1999]. Antagonists of this particular receptor have potential therapeutic value in conditions such as irritable bowel syndrome, chronic obstructive airway disease and urinary incontinence. Three active molecules were used to define a series of 3D pharmacophores (using the clique detection method). The initial list of five pharmacophores was pruned following visual

examination to give two very similar pharmacophores containing a positively charged amine, a hydrogen-bond acceptor atom and two hydrogen-bond donor sites. Searching a 3D database and combining the selected molecules gave 172, which were tested. Three compounds were found to have significant activity in the assay, one of which proved to be of particular interest, being a simple molecule particularly amenable to lead optimisation.

12.8 Molecular Similarity and Similarity Searching

Substructure and 3D pharmacophore searching involve the specification of a precise query, which is then used to search a database in order to identify molecules for screening. In such an approach, either a molecule matches the query or it does not Similarity searching offers a complementary approach, in that the query is typically an entire molecule. This query molecule is compared to all molecules in the database and a similarity coefficient calculated. The top-scoring database molecules (based on the similarity coefficient) are the 'hits' from the search. In a typical scenario the query molecule would be known to possess some desirable activity and the objective would be to identify molecules which will hopefully show the same activity. We therefore require some method for deciding how to compute the similarity between two molecules. In order to achieve this we need to choose a set of *molecular descriptors* for the compounds. These descriptors are then used to compute the similarity coefficient.

12.9 Molecular Descriptors

The descriptors that we will consider in this section are those which can be calculated readily from the molecular formula, the molecular graph or from one or more computed 3D conformations. It is also possible to include experimentally determined descriptors, but this is often not feasible due to the unavailability of experimental data or the expense in acquiring it, especially where there are many molecules to consider. Indeed, the molecules may not yet have been synthesised! Some descriptors can be calculated very rapidly, one obvious example being the molecular weight. Other descriptors may be time-consuming to calculate, such as those derived from quantum mechanics. Some descriptors have an obvious experimental counterpart with which the calculation can be compared, such as a partition coefficient. Others are purely computational, such as a binary fingerprint. Some descriptors refer to properties of the whole molecule; others refer to the properties of individual atoms. New descriptors are being invented continually, each purporting to provide novel insights into the relationship between a molecule's structure and its properties. Some commonly used descriptors are shown in Table 12.2.

12.9.1 Partition Coefficients

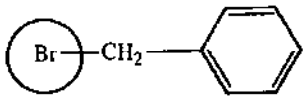
A very popular descriptor is $\log P$, the logarithm of the partition coefficient, most commonly for the partition between 1-octanol and water (the logarithm converts the

Descriptor	Information typically required for calculation	Comments
Molecular weight	Molecular formula	
Hashed fingerprints, structural keys	2D structure	See Section 12.2
Counts of specific atoms, rings or other features (e.g. hydrogen-bond donors, acceptors)	2D structure	Typically based on substructure searches
Octanol/water partition coefficient	2D structure	See Section 12.9.1
Calculated molar refractivity	2D structure	See Section 12.9 2
Molecular connectivity χ indices	2D structure	See Section 12.9.3
κ shape indices		See Section 12.9 3
Electrotopological indices		See Section 12.9 3
Atom pairs, topological torsions	2D structure	See Section 12.9.3
Dipole moment	3D structure	
Molecular volume, surface area, polar surface area	3D structure	Polar surface area is the amount of molecular surface due to polar atoms
Quantum mechanical descriptors (e.g. HOMO–LUMO energy gap)	3D structure	See Section 2 7 4
Partial atomic charge, polarisability	3D or 2D structure	See Section 4.9
Pharmacophore keys	Family of low-energy conformations	See Section 12.9.4
Geometric atom pairs, angles, torsions	3D structure	See Section 12 9.3

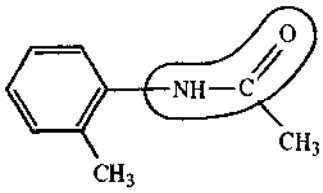
Table 12 2. A list of some of the more common descriptors Details of some of these descriptors can be found elsewhere as indicated. This table is restricted to those descriptors which can be computed; it therefore excludes certain classes (such as the Hammett substituent constants) which are derived from experimental studies (see Section 12.12).

value onto a free-energy scale). Experimental determination of the partition coefficient can be difficult, particularly for zwitterionic and very lipophilic or polar compounds. The octanol/water system was selected by Hansch as a model system for hydrophobicity and has proved very successful, as we shall see in our discussion of quantitative structure-activity relationships in Section 12.12. Nevertheless, in some cases it would be more appropriate to measure the partition coefficient in an alternative system, and it is now possible to measure partition coefficients between lipid membranes and water directly.

Various theoretical methods can be used to calculate partition coefficients. The partition coefficient is an equilibrium constant and so is directly related to a free energy change. Partition coefficients can be calculated using free energy perturbation methods, as discussed in Section 11.3.2. These methods suffer from the limitations of force field parametrisation and the large amount of computer time that is required to perform such calculations. More widely used are fragment-based approaches, in which the partition coefficient is calculated as a sum of individual fragment contributions plus a set of correction factors. Such an approach clearly depends critically upon the definition of the fragments. The widely used CLOGP program of Hansch and Leo [Leo 1993] uses a small number of compounds to accurately define a set of fragment values. CLOGP breaks a molecule into fragments by identifying 'isolating carbons', which are carbon atoms that are not doubly



Bromide fragment	0 480
1 aliphatic isolating carbon	0 195
6 aromatic isolating carbons	0 780
7 hydrogens on isolating carbons	1 589
1 chain bond	–0 120
Total	2 924



NH-amide fragment	–1 510
2 aliphatic isolating carbons	0 390
6 aromatic isolating carbons	0.780
10 hydrogens on isolating carbons	2 270
1 chain bond	–0 120
1 benzyl bond	–0 150
ortho substituent	–0 760
Total	0 900

Fig. 12 24· CLOGP calculations on benzyl bromide and o-methyl acetanilide

or triply bonded to a heteroatom. These carbon atoms and their attached hydrogens are considered hydrophobic fragments, with the remaining groups of atoms being the polar fragments. A partition coefficient is calculated by adding together appropriate values for the fragments and the isolating carbons, together with various correction factors. The process is illustrated in Figure 12.24 for two simple molecules, benzyl bromide and *o*-methyl acetanilide. Benzyl bromide contains one aliphatic isolating carbon and six isolating aromatic carbons, together with one bromide fragment. Each of these fragments contributes a characteristic score, to which are added values for the seven hydrogens on the isolating carbons and a contribution from one acyclic bond. *o*-Methyl acetanilide contains an amide fragment, two aliphatic isolating carbons and six isolating aromatic carbons. In addition, there are contributions from the hydrogen atoms, the acyclic bond, a benzyl bond (to the *o*-methyl group) and a factor due to the presence of an *ortho* substituent.

The CLOGP program remains the benchmark by which other methods for calculating octanol–water partition coefficients tend to be judged. One of its main drawbacks is the need for data for all the fragments in the molecule. Whilst the requisite data for a considerable number of fragments are included by default, the calculation will often not be correct for a fraction of the molecules in a typical pharmaceutical database. Of course, if the fragment is common to a molecular series of particular interest then it is usually straightforward to perform the experiment and add the necessary fragment value. An alternative is to use an atom-based approach to estimating the partition coefficient. This is very similar to the fragment-based method, but rather than checking for fragments the molecule is broken down into the atom types present. In the simplest case, the partition coefficient is given by a summation of the contributions from each atom type [Ghose and Crippen 1986; Ghose *et al* 1998; Wang *et al* 1997; Wildman and Crippen 1999]:

$$\log P = \sum_i n_i a_i \qquad (12\ 4)$$

where n_i is the number of atoms of atom type i and a_i is the atomic contribution. These contributions are determined by regression analysis The basic atomic contribution is in some cases moderated by correction factors to account for particular classes of molecules.

12.9.2 Molar Refractivity

The molar refractivity (MR) is given by:

$$\mathrm{MR} = \frac{(n^2 - 1)}{(n^2 + 1)} \frac{\mathrm{MW}}{d} \tag{12.5}$$

In Equation (12.5) MW is the molecular weight, d is the density and n is the refractive index. The refractive index does not vary much from one organic compound to another and as the molecular weight divided by the density equals the volume, MR gives some indication of the steric bulk of a molecule. The presence of the refractive index term also provides a connection to the polarisability of the molecule. Molar refractivity can be calculated using atomic values with some correction factors for certain types of bonding (the CMR program [Leo and Weininger 1995]).

As we have seen, a common route to calculating both the partition coefficient and molar refractivity is by combining in some way the contributions from the fragments or atoms in the molecule. The fragment contributions are often determined using multiple linear regression, which will be discussed below (Section 12.12.2). Such an approach can be applied to many other properties, of which we shall mention only one other here, solubility. Klopman and colleagues were able to derive a regression model for predicting aqueous solubility based upon the presence of groups, most of which corresponded to a single atom in a specific hybridisation state but also included acid, ester and amide groups [Klopman *et al.* 1992]. This gave a reasonably general model that was able to predict the solubility of a test set within about 1.3 log units. A more specific model which contained more groups performed better but was of less generic applicability.

12.9.3 Topological Indices

Many of the descriptors which can be calculated from the 2D structure rely upon the molecular graph representation because of the need for rapid calculations. Kier and Hall have developed a large number of *topological indices*, each of which characterises the molecular structure as a single number [Hall and Kier 1991]. Every non-hydrogen atom in the molecule is characterised by two 'delta' values, the simple delta δ_i and the valence delta δ_i^v:

$$\delta_i = \sigma_i - h_i, \quad \delta_i^v = Z_i^v - h_i \tag{12.6}$$

where σ_i is the number of sigma electrons for atom i, h_i is the number of hydrogen atoms bonded to atom i and Z_i^v is the number of valence electrons for atom i. Thus the simple delta value will differentiate CH_3 from $-CH_2-$. CH_3 has the same simple delta value as NH_2 but a different valence delta. For elements beyond fluorine in the periodic table the

valence delta expression is modified as follows:

$$\delta_i^v = (Z_i^v - h_i)/(Z_i - Z_i^v - 1) \tag{12.7}$$

where Z_i is the atomic number. The *chi molecular connectivity indices* are obtained by summing functions of these delta values. Thus the chi index of order zero is defined as follows:

$$^0\chi = \sum_{\text{atoms}} (\delta_i)^{-1/2}; \quad ^0\chi^v = \sum_{\text{atoms}} (\delta_i^v)^{-1/2} \tag{12.8}$$

The summations are over the atoms in the molecule. This particular index does not encode much information about the structure. The first-order chi index involves a summation over bonds:

$$^1\chi = \sum_{\text{bonds}} (\delta_i\delta_j)^{-1/2}; \quad ^1\chi^v = \sum_{\text{bonds}} (\delta_i^v\delta_j^v)^{-1/2} \tag{12.9}$$

Higher-order chi indices involve summations over sequences of two, three, etc., bonds. To illustrate the difference between these indices for a series of related structures we show in Figure 12.25 the $^0\chi$, $^1\chi$ and $^2\chi$ indices for the isomers of hexane.

The *kappa shape indices* are generated by assessing how a molecular structure compares to molecular graphs with extreme shapes. As with the molecular connectivity indices, there

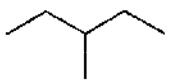
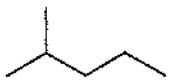
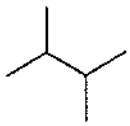
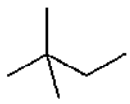
	Paths of length 2	Paths of length 3	Paths of length 4	Paths of length 5	$^0\chi$	$^1\chi$	$^2\chi$
	4	3	2	1	4 828	2 914	1 707
	5	4	1	0	4 992	2.808	1 922
	5	3	2	0	4.992	2 770	2 183
	6	4	0	0	5 155	2 643	2 488
	7	3	0	0	5 207	2 561	2 914

Fig. 12.25· Chi indices for the various isomers of hexane. (Figure adapted in part from Hall L H and L B Kier 1991 The Molecular Connectivity Chi Indexes and Kappa Shape Indexes in Structure-property Modeling In Lipkowitz K B and D B Boyd (Editors) Reviews in Computational Chemistry *Volume 2 New York, VCH Publishers, pp. 367–422.)*

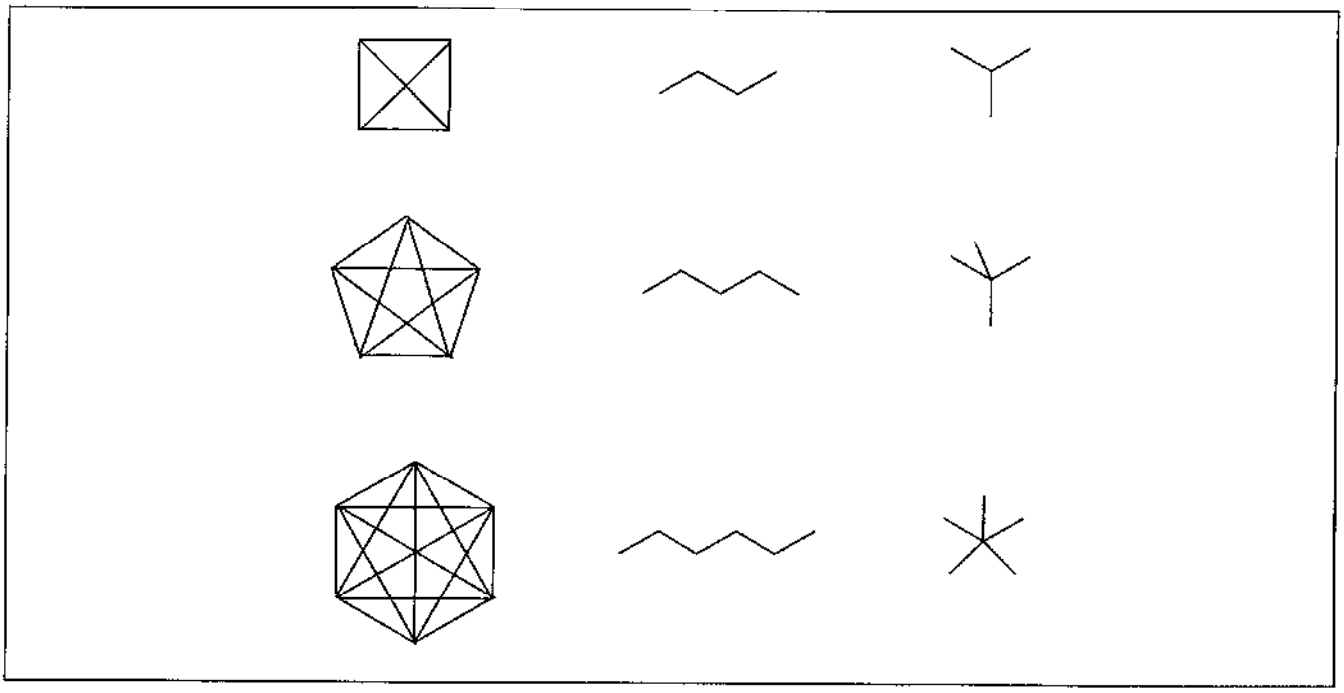


Fig. 12.26 First- and second-order extreme shapes for four-, five- and six-atom graphs (the linear molecule gives rise to the minimum in each case)

are shape indices of various order (first, second, etc.), with the first-order shape index involving a count over single-bond fragments, the second-order shape index involving a count of two-bond paths, and so on. In the first-order shape index the two extreme shapes are the linear molecule and the completely connected graph, where every atom is connected to every other atom (Figure 12.26). These two graphs contain $(A-1)$ and $A(A-1)/2$ bonds, respectively, where A is the number of atoms. If the number of bonds in our molecule is ${}^1P$ then ${}^1P_{max} \geqslant {}^1P \geqslant {}^1P_{min}$, where ${}^1P_{max}$ and ${}^1P_{min}$ are the maximum and minimum number of bonds for that number of atoms. The first-order kappa index is written:

$$ {}^1\kappa = \frac{2\,{}^1P_{max}\,{}^1P_{min}}{({}^1P)^2} = \frac{A(A-1)^2}{2({}^1P)^2} \tag{12.10} $$

The second-order kappa index is determined by the count of two-bond paths, written ${}^2P$. The maximum value is expressed by a 'star' shape (${}^2P_{max} = (A-1)(A-2)/2$) and the minimum value again corresponds to the linear molecule (${}^2P_{min} = A-2$). The second-order shape index is then:

$$ {}^2\kappa = \frac{2\,{}^2P_{max}\,{}^2P_{min}}{({}^2P)^2} = \frac{(A-1)(A-2)^2}{2({}^2P)^2} \tag{12.11} $$

As with the molecular connectivity indices, higher-order shape indices have also been defined. The kappa indices themselves do not include any information about the identity of the atoms. This is the role of the 'kappa–alpha' indices. The alpha value for each atom is a measure of its size relative to some standard (chosen to be the sp^3-hybridised carbon):

$$ \alpha_x = \frac{r_x}{r_{Csp^3}} - 1 \tag{12.12} $$

An alpha value is calculated for the molecule by summing the individual atomic alphas and then incorporating them into the shape indices as follows:

$$^{1}\kappa_{\alpha} = \frac{(A+\alpha)(A+\alpha-1)^2}{2(^{1}P+\alpha)^2} \tag{12.13}$$

$$^{2}\kappa_{\alpha} = \frac{(A+\alpha-1)(A+\alpha-2)^2}{2(^{2}P+\alpha)^2} \tag{12.14}$$

The final graph theoretical index popularised by Kier and Hall that we shall consider is the electrotopological state index [Hall *et al.* 1991]. Unlike the molecular connectivity and shape indices this is determined for each atom (including the hydrogen atoms, if so desired). This index depends upon the *intrinsic state* of an atom, which for an atom i (in the first row of the periodic table) is given by:

$$I_i = \frac{\delta_i^{v} + 1}{\delta_i} \tag{12.15}$$

This intrinsic state is a reflection of the electronic and topological characteristics of the atom i. The effects of interactions with the other atoms are incorporated by determining the number of bonds between the atom i and each of the other atoms, j. If this path length is r_{ij} then a perturbation is defined as:

$$\Delta I_i = \sum_j \frac{I_i - I_j}{r_{ij}^2} \tag{12.16}$$

The sum of ΔI_i and I_i gives the state of each atom (the *E state*). This descriptor is considered to encode the electronegativity of each atom (including the inductive effects of other atoms), together with its topological state. These atomic topological states can be combined into a whole-molecule descriptor by calculating the mean square value for the atoms. Vector or bit-string representations can also be produced. There is a finite number of possible I values, and so a bitstring representation can be obtained by setting the appropriate bit for each of the different I values present in a molecule. Alternatively, one can compute the sum or mean of the different E state values for each unique intrinsic state, to give a vector of real numbers.

Atom pairs [Carhart *et al.* 1985] and topological torsions [Nilakantan *et al.* 1987] are a related set of structural descriptors. Each atom pair descriptor codes the elemental type of a pair of atoms in the molecule together with the number of non-hydrogen atoms to which they are bonded, how many π-bonding electrons they have and the length of the shortest path between them. A topological torsion codes a sequence of four connected atoms together with their types, number of non-hydrogen connections and number of π electrons. Geometric atom pairs are a 3D equivalent, measuring the actual distance (in ångströms) between pairs of atoms, and likewise for other geometric descriptors

12.9.4 Pharmacophore Keys

The development of 3D pharmacophore methods has spawned a new type of descriptor, the *pharmacophore key*. This is an extension of the binary keys used to facilitate 3D database

searching. When large sets of molecules are being considered then one is often restricted to properties that can be calculated relatively rapidly. This often precludes many of the properties dependent upon the 3D structure. Moreover, even when this 3D information is used it is often based upon a single conformation. The pharmacophore key can be computed relatively rapidly and it takes into account conformational flexibility and the pharmacophoric features in a molecule. In its simplest form, pharmacophores containing three features are represented. During the conformational analysis the pharmacophore features within each acceptable conformation are identified. All possible combinations of three features are enumerated, together with the distances between them (e.g. 'hydrogen-bond donor 6 Å from an aromatic ring centroid and 4 Å from a basic nitrogen with the third distance being 7 Å', Figure 12.27). Each distance is assigned to a distance bin, as described earlier. Every 3-point pharmacophore (distinguished by the features it contains and the distances)

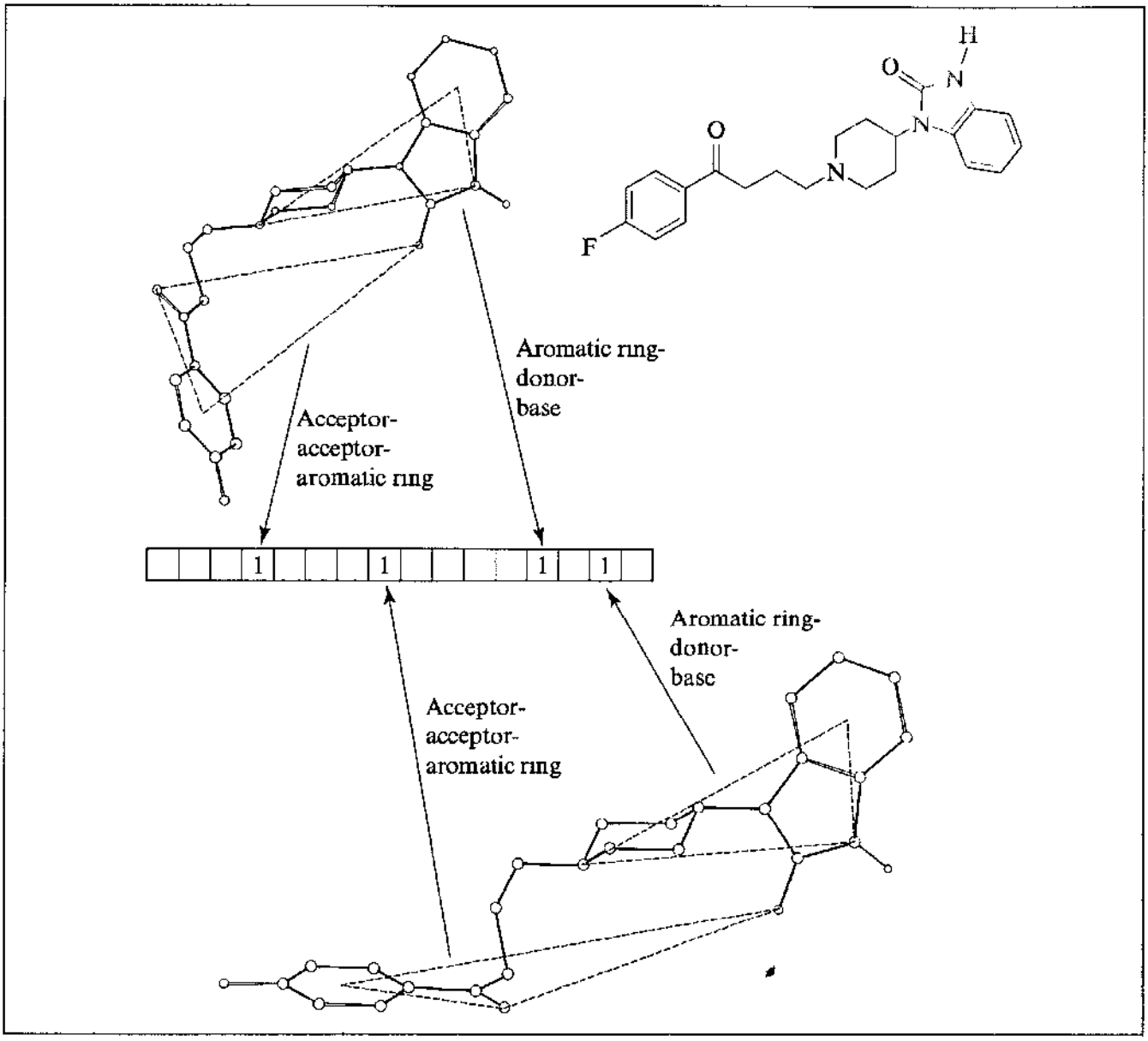


Fig 12 27: The generation of 3-centre pharmacophore keys, illustrated using benperidol Two different conformations are shown, together with two different combinations of three pharmacophore points

is associated with a particular bit in the pharmacophore key bitstring. The pharmacophore key thus codes all possible 3-point pharmacophores that the molecule could express. These pharmacophore keys can be used in the same manner as any other binary descriptor. In addition to the use of 3-point pharmacophore keys [Good and Kuntz 1995; Pickett *et al.* 1996], 4-point pharmacophores are also possible [Mason *et al.* 1999]. These are claimed to contain more information and be more discriminatory than the 3-point keys (four points is required to differentiate stereoisomers, for example). However, the number of bits in a 4-point pharmacophore is considerably more than for the 3-point, and it is often not practical to store them simply as a single, large bitstring but as a sequence of integers, each of which identifies the bits that are set on

12.9.5 Calculating the Similarity

The descriptors of a molecule can be considered a vector of attributes. These attributes may be real numbers or they may be binary in nature; in the case of the latter a value of 1 often indicates the presence of some feature and a value of 0 its absence. Having defined the descriptors, the next step is to compute a quantitative measure of the similarity [Willett *et al.* 1998]. Many similarity coefficients are in the range 0 to 1, with 1 indicating maximum similarity (note that this does not necessarily mean that the molecules are identical). Similarity is often considered to be complementary to distance, such that subtraction of the similarity coefficient from one gives the 'distance' between two molecules. Such distances may then be used in methods such as cluster analysis (see Section 9.13).

Here we will introduce three similarity coefficients that have been widely used for both real-valued (i.e. continuous) and binary (*dichotomous*) descriptors: the Tanimoto coefficient, the Dice coefficient and the Cosine coefficient The formulae used to compute these coefficients are given in Table 12.3, where, for completeness, we have also provided the Euclidean and Hamming expressions that were introduced in Section 9.13. Different expressions are used for real-valued data (where the molecule is represented by a vector containing N real values x_i) and for binary data (where each molecule is represented by N binary values). For binary data, we additionally define a to be the number of bits 'on' in the bitstring for A, b to be the number of bits 'on' in the bitstring for B, and c to be the number of bits that are 'on' in both A and B (calculated using the AND operator).

Of the three similarity coefficients, the one most commonly used for binary molecular data (such as structural keys or hashed fingerprints) is the Tanimoto coefficient. It is important to recognise that there are some subtle differences between the way in which these metrics quantify the similarity between a series of compounds. These differences can be particularly marked for simple molecules which do not contain much functionality. Consider chlorpromazine and methoxypromazine, two phenothiazine neuroleptics (Figure 12.28). The Hamming 'distance' between these two molecules when calculated using the Daylight hashed fingerprints is 61 and the Soergel distance (equal to the complement of the Tanimoto coefficient for dichotomous data) is 0.28. These two molecules differ only by the substitution of a methoxy group for a chlorine atom. By contrast, the Hamming and Soergel distances between two smaller molecules that differ in the same way (methyl chloride and dimethyl

Name	Formula for continuous variables	Formula for binary (dichotomous) variables
Tanimoto similarity coefficient Also known as the Jaccard coefficient Complement equals the Soergel distance for dichotomous data	$S_{AB} = \frac{\sum_{i=1}^{N} x_{iA}x_{iB}}{\sum_{i=1}^{N}(x_{iA})^2 + \sum_{i=1}^{N}(x_{iB})^2 - \sum_{i=1}^{N} x_{iA}x_{iB}}$ Range: −0 333 to +1	$S_{AB} = \frac{c}{a+b-c}$ Range: 0 to 1
Dice similarity coefficient Also known as the Hodgkin index	$S_{AB} = \frac{2\sum_{i=1}^{N} x_{iA}x_{iB}}{\sum_{i=1}^{N}(x_{iA})^2 + \sum_{i=1}^{N}(x_{iB})^2}$ Range: −1 to +1	$S_{AB} = \frac{2c}{a+b}$ Range: 0 to 1
Cosine similarity coefficient Also known as the Carbo index	$S_{AB} = \frac{\sum_{i=1}^{N} x_{iA}x_{iB}}{[\sum_{i=1}^{N}(x_{iA})^2 \sum_{i=1}^{N}(x_{iB})^2]^{1/2}}$ Range: −1 to +1	$S_{AB} = \frac{c}{(ab)^{1/2}}$ Range 0 to 1
Euclidean distance	$D_{AB} = \left[\sum_{i=1}^{N}(x_{iA} - x_{iB})^2\right]^{1/2}$ Range: 0 to ∞	$D_{AB} = [a+b-2c]^{1/2}$ Range: 0 to N
Hamming distance Also known as the Manhattan or city-block distance	$D_{AB} = \sum_{i=1}^{N} \lvert x_{iA} - x_{iB}\rvert$ Range: 0 to ∞	$D_{AB} = a+b-2c$ Range: 0 to N

Table 12 3: Formulae for various commonly used ways to compute the similarity or distance between molecules For the binary data a is defined to be the number of bits 'on' in molecule A, b is the number of bits 'on' in molecule B and c is the number of bits that are 'on' in both A and B. Table based on [Willett et al. *1998].*

thioether) are 16 and 0.80, respectively. The Hamming distance measure thus appears to suggest that the two smaller molecules are 'closer together' (more similar) than the pair of larger molecules. This contrasts with the Soergel/Tanimoto result. One reason for this difference is that the Hamming distance considers a common absence of features to indicate similarity, unlike the Soergel/Tanimoto measures. Moreover, the denominator in the Tanimoto coefficient has the effect of normalising the results according to the size of the molecule.

Another important feature of the Tanimoto coefficient when used with bitstring data is that small molecules, which tend to have fewer bits set, will have only a small number of bits in common and so can tend to give inherently low similarity values. This can be important when selecting 'dissimilar' compounds, as a bias towards small molecules can result.

A generalisation of the similarity formulae for binary data can be derived, based on the work of Tversky [Tversky 1977; Bradshaw 1997]. This takes the form:

$$S_{\text{Tversky}} = \frac{c}{\alpha(a-c) + \beta(b-c) + c} \tag{12.17}$$

where α and β are user-defined constants. The Tanimoto coefficient is recovered if $\alpha = \beta = 1$ and the Dice coefficient if $\alpha = \beta = \frac{1}{2}$. One interesting feature is that the Tversky coefficient is

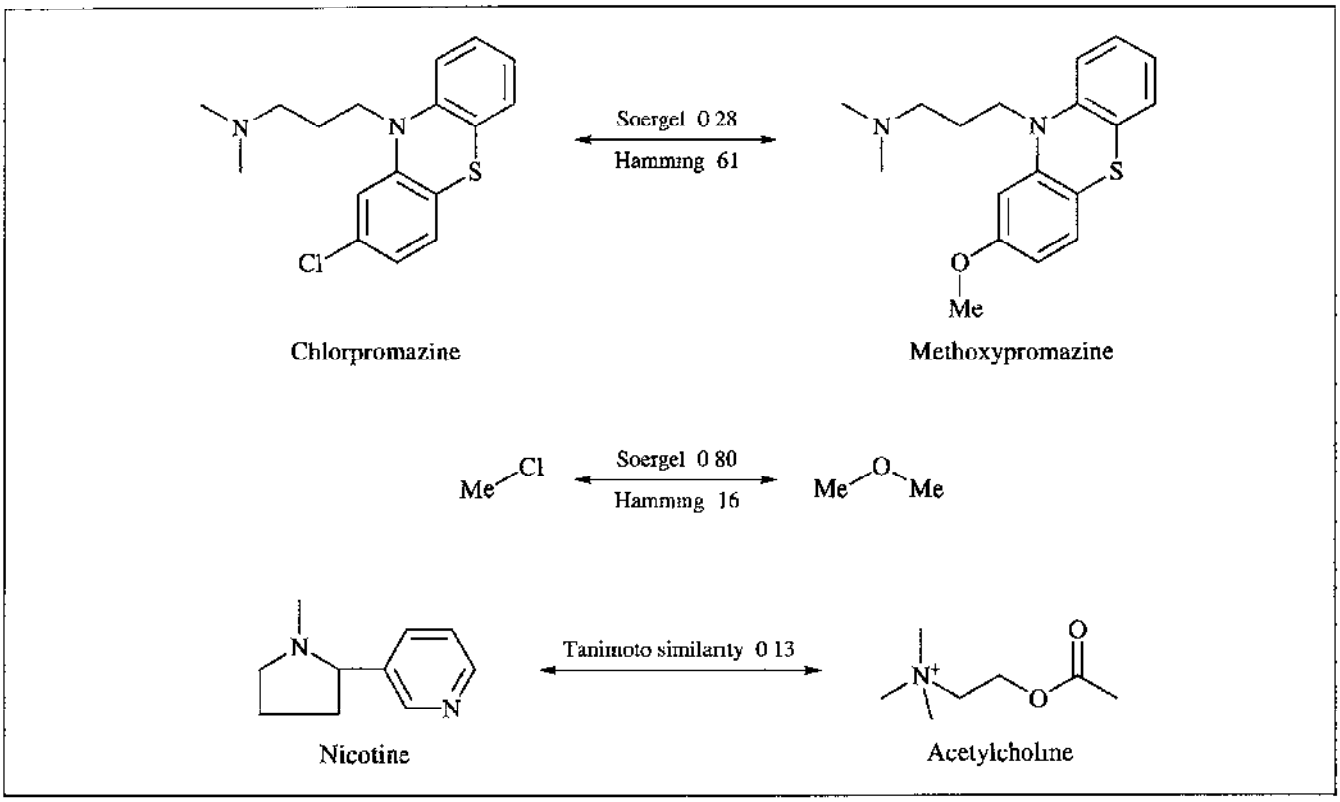


Fig. 12.28: The differences between the Soergel and Hamming distance measures for various molecules (see text)

asymmetric, such that $S_{\text{Tversky}}(A, B) \neq S_{\text{Tversky}}(B, A)$. If $\alpha = 1$ and $\beta = 0$ then the Tversky similarity value can be interpreted as being the fraction of the features in A which are also in B; a value of 1 with these parameters indicates that A is a 'substructure' of B.

12.9.6 Similarity Based on 3D Properties

Similarity methods based on rapidly calculated properties (particularly those derived from structural keys or hashed fingerprints) have become very popular, particularly for dealing with large numbers of molecules. Similarity measures derived from the 2D structure will tend to associate molecules with common substructures. However, molecular recognition depends on the three-dimensional structure and properties (e.g. electrostatics and shape) of a molecule rather than the underlying substructure. A simple illustration of this is provided by nicotine and acetylcholine (Figure 12.28), which have a very low similarity (0.13, as determined using the Tanimoto coefficient from the Daylight hashed fingerprints), despite the fact that they act at the same biological receptor. A simple 3D pharmacophore model to rationalise this comprises a positively charged or basic group an appropriate distance from an acceptor. For reasons such as this there has been much interest in similarity measures based upon three-dimensional properties.

Several measures of the shape and electronic similarity between pairs of molecules have been devised. The *Carbo index* was developed to enable the electron density of two molecules in some relative orientation to be compared [Carbo *et al.* 1980]. It is essentially the Cosine

coefficient (Table 12.3):

$$S_{AB} = \frac{\int \rho_A \rho_B \, d\nu}{(\int \rho_A^2 \, d\nu)^{1/2} (\int \rho_B^2 \, d\nu)^{1/2}} \tag{12.18}$$

The electron densities at each point are determined in the usual way from the square of the wavefunction. The value of the Carbo index runs from 0 (no similarity) to 1 (perfect similarity). Unfortunately, the electron density is not an ideal measure of similarity, because the density is strongest near the atomic nuclei and so the Carbo formula will be dominated by the extent to which the nuclei overlap. The electrostatic potential is a more appropriate property as it emphasises electronic effects away from the nuclei. Another drawback of the Carbo index is that it does not depend upon the magnitude of the property at a point but just its sign. This means that a location where the potential is positive from one molecule and equal but negative from the other would be weighted the same, irrespective of the magnitude of the potential. Hodgkin and Richards suggested the following alternative measure of similarity for use with the electrostatic potential [Hodgkin and Richards 1987]:

$$S_{AB} = \frac{2 \int \phi_A(\mathbf{r}) \phi_B(\mathbf{r}) \, d\mathbf{r}}{\int \phi_A^2(\mathbf{r}) \, d\mathbf{r} + \int \phi_B^2(\mathbf{r}) \, d\mathbf{r}} \tag{12.19}$$

A positive value of the Hodgkin–Richards index is obtained if large charges of the same sign are located in approximately the same regions of space; a negative value is obtained if large charges of opposite sign are located in the same regions of space. This index is effectively the Dice coefficient.

The integrals in the Hodgkin–Richards approach can be evaluated in a number of ways. One approach is to position the molecules within a rectangular grid and to evaluate the electrostatic potential due to each molecule at each grid point. The integrals in Equation (12.19) are then determined numerically by summing over the grid points. This can be rather slow, particularly if the molecules are allowed to vary their relative orientations and conformations in order to find the location of maximum similarity. An alternative is to represent the potential using an analytical function. For example, linear combinations of Gaussian functions can be fitted to the potential, enabling the similarity measure to be computed much more rapidly [Good *et al.* 1993].

3D similarity methods such as these provide the means to generate a structural alignment of molecules based upon some suitable property (electrostatics or shape). Methods such as pharmacophore mapping also provide a mechanism to align molecules, but in this case based upon their pharmacophore features. A structural alignment of a set of active molecules can be very useful in drug design, particularly when the structure of the target receptor is. Some of the varied techniques, such as 3D database searching and comparative molecular field analysis, which make use of structural alignments, are discussed in this chapter. The abundance of techniques for generating such alignments reflects the complex nature of the problem, in part a consequence of the need to consider the conformational flexibility of the molecules together with their relative orientations in space [Lemmen and Lengauer 2000].

12.10 Selecting 'Diverse' Sets of Compounds

Chemical diversity is a term which has been widely used and vigorously debated in the literature and at conferences since the advent of the era of high-throughput screening and combinatorial synthesis. But what is chemical diversity, and how can it be quantified? Given a finite number of screening slots or a finite number of compounds that could be synthesised, one could argue that it is desirable that the molecules are as diverse as possible, to maximise the chances of identifying one or more lead compounds. We therefore require methods which can be used to select diverse sets of compounds and techniques for comparing the diversity of one set against another. As there are $N!/k!(N-k)!$ possible ways to select a subset of k compounds from a total of N compounds it is obviously not feasible to examine all possible cases (for example, there are more than 10^{10} ways to select ten compounds out of just 50). Three popular ways to select 'diverse' sets of compounds are cluster analysis, dissimilarity-based methods and partition-based methods. Before applying any of these methods, however, it is usually recommended that the descriptors used to characterise the molecules are examined to determine whether some form of data manipulation is required.

12.10.1 Data Manipulation

Several tests and manipulations can be performed on a data set. For example, there is nothing gained by including a descriptor which shows no variation over the molecules in the set. It may also be useful to consider the distribution of values; some methods assume that the data are distributed according to a normal distribution, and significant deviations from this distribution will lead to invalid results. This is particularly the case with some of the methods used to derive quantitative structure–activity relationships. Two of the more common deviations from normality are skewness and kurtosis; the former indicates that the distribution is no longer symmetrical and the latter measures how 'peaked' the distribution is (Figure 12.29). Skewness and kurtosis are related to the third and fourth moments of the data (the mean and the variance corresponding to the first and second moments); these were encountered in our discussion of the moments theorem (see Section 4.23). Sometimes the distribution is bimodal, with two means. The *coefficient of variation* is a metric that can be used to identify descriptors which have a good spread over the range

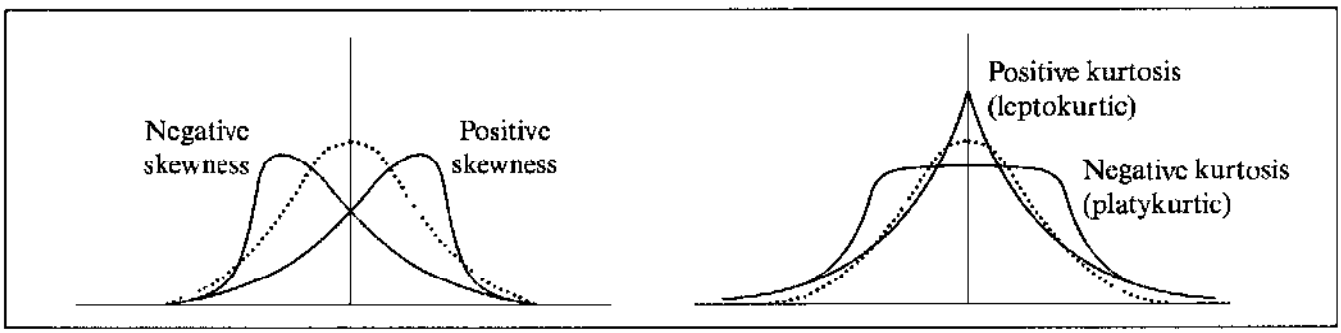


Fig 12 29 Deviations from the normal distribution: skewness and kurtosis.

of the descriptor. This coefficient is equal to the standard deviation divided by the mean, and it automatically adjusts for different measurement scales. The larger the value the better the spread of values. Not all of the techniques described in this section need necessarily be used, but it certainly makes sense to perform some basic checks

If the descriptors are on different scales then those which naturally occupy a 'larger' scale may be given more weight in the subsequent analysis, simply because of their natural units. In *autoscaling* the descriptors are scaled to zero mean and a standard deviation of 1.

$$x_i' = \frac{x_i - \bar{x}}{\sigma} \tag{12.20}$$

An alternative to autoscaling is range scaling, where the denominator in Equation (12.20) equals the range (the difference between the maximum and minimum values) Range scaling gives a set of new values between −0.5 and +0.5.

It is also important to check for correlations between the descriptors. Highly correlated descriptors could lead to the information that they encode being over-represented. A straightforward way to determine the degree of correlation between two properties is to calculate a correlation coefficient. Pearson's correlation coefficient is given by:

$$r = \frac{\sum_{i=1}^{N}(x_i - \langle x\rangle)(y_i - \langle y\rangle)}{\sqrt{\left[\sum_{i=1}^{N}(x_i - \langle x\rangle)^2\right]\left[\sum_{i=1}^{N}(y_i - \langle y\rangle)^2\right]}} \tag{12.21}$$

where $\langle x\rangle$ and $\langle y\rangle$ are the arithmetic means of x and y, and N is the number of compounds. A value of 1.0 indicates a perfect positive correlation such that the x, y coordinates lie on a straight line with a positive slope. A value of −1.0 indicates a perfect negative correlation with the x, y coordinates on a straight line with a negative slope. A value of 0.0 indicates either no correlation or that the x, y coordinates follow a non-linear scatter.

Another way to identify correlations is to plot the values of the parameters in graphical form; this can help to identify any correlations and the presence of 'outliers'. A *Craig plot* is a two-dimensional scatterplot of one parameter against another; ideally, the molecules should sample from all four quadrants of the plot.

One way to try to alleviate the problem of correlated descriptors is to perform a principal components analysis (see Section 9.13). Those principal components which explain (say) 90% of the variance may be retained for the subsequent calculations Alternatively, those principal components for which the associated eigenvalue exceeds unity may be chosen, or the principal components may be selected using more complex approaches based on cross-validation (see Section 12.12.3). It may be important to scale the descriptors (e.g. using autoscaling) prior to calculating the principal components. However, unless each principal component is largely associated with any particular descriptor it can be difficult to interpret the physical meaning of any subsequent results.

An alternative to principal components analysis is *factor analysis* This is a technique which can identify multicollinearities in the set – these are descriptors which are correlated with a linear combination of two or more other descriptors. Factor analysis is related to (and

often confused with) principal components analysis. Factor analysis seeks to express each descriptor as a linear combination of factors For example, if we have some descriptor x_i then each of the N values of this descriptor ($x_{i,j}$ where j runs from 1 to N, corresponding to the N molecules in the set) can be written in terms of the factors as follows:

$$x_{i,j} = a_i^1 F_j^1 + a_i^2 F_j^2 + a_i^3 F_j^3 + \cdot \cdot + a_i^d F_j^d + E_{i,j} \tag{12.22}$$

This can be expressed in matrix form $\mathbf{X} = \mathbf{F}\mathbf{A}^{\mathrm{T}} + \mathrm{E}$. $F_j^1, F_j^2 \ldots F_j^d$. are the d common factors; for each factor there is a value for each of the N data items. Note that to achieve a data reduction d should be less than the total number of descriptors in the set. $E_{i,j}$ is the unique factor specific to the descriptor x_i for molecule j. The coefficients a_i^1, a_i^2, etc., are known as the *loadings* of the descriptor values onto the common factors. Recall that in principal components analysis each principal component is expressed as a linear combination of the variables; factor analysis uses a similar linear expression but one in which the variables themselves are expressed in terms of the factors. To retrieve the value of a descriptor for a particular molecule one uses Equation (12.22) with the appropriate factor values corresponding to that molecule

The unique factors are usually removed as they are considered to represent irrelevant information specific to the particular descriptors alone (such as experimental error). This leaves the common factors, which like the principal components are orthogonal. It sometimes happens that several descriptors may be loaded onto one or more factors. The factors are often 'rotated' in order to try to arrange for each factor to be largely associated with as few variables as possible. Thus if any given factor is largely associated with only one or two variables then it can be much easier to interpret any subsequent analysis. Rotation can also be applied to a set of principal components.

12.10.2 Selection of Diverse Sets Using Cluster Analysis

Cluster analysis was considered in our discussion of conformational analysis (see Section 9.13); for compound selection one would typically want to select a representative molecule or molecules from each cluster. A practical consideration when deciding which cluster analysis method to use is that for large numbers of molecules some algorithms may not be feasible because they require an excessive amount of memory or may have a long execution time. Another consideration with cluster analysis (and with some of the other methods that we will discuss) is the need to calculate the distance between each pair of molecules from the vector of descriptors (or from their scaled derivatives or from a set of principal components, if these are being used). For binary descriptors such as molecular fingerprints this distance is often given by $1 - S$, where S is the similarity coefficient (Table 12.3).

Two examples of the use of cluster analysis in compound selection are the studies of Downs, Willett and Fisanick [Downs *et al.* 1994] and Brown and Martin [Brown and Martin 1996]. In the former study, each molecule was described by 13 properties. The objective was to determine how well each of the different clustering methods considered was able to predict these property values The property value for each molecule was

predicted as the mean of the property values for the other molecules in the cluster. The predicted values for each molecule were compared with the actual values to score each cluster method. Both hierarchical and non-hierarchical methods were used and different numbers of clusters were formed. As would be expected, the more clusters that were formed the more accurate the predictions (because the molecules in each cluster were more alike). However, a balance is required because at least one other molecule needs to be in the cluster for a prediction to be made Of the methods considered, the hierarchical algorithms performed significantly better than the non-hierarchical Jarvis-Patrick method in terms of their predictive ability.

The Brown and Martin study considered a variety of clustering methods, together with several structural descriptors such as structural keys, fingerprints and pharmacophore keys. The methods were evaluated according to their ability to separate a set of molecules so that active and inactive compounds were in different clusters. Four different data sets were considered, the results suggesting that a combination of 2D descriptors (particularly the structural keys) and hierarchical clustering methods were most successful. This was a rather surprising result which caused much subsequent debate, not least because the structural keys were not particularly designed for use in compound selection but rather for fast substructural searching. In particular, there was much discussion concerning the nature of the data sets used in this experiment, which contained a rather higher proportion of structurally similar, active molecules than might be the case in a typical high-throughput screening scenario.

12.10.3 Dissimilarity-based Selection Methods

In dissimilarity-based compound selection the required subset of molecules is identified directly, using an appropriate measure of dissimilarity (often taken to be the complement of the similarity). This contrasts with the two-stage procedure in cluster analysis, where it is first necessary to group together the molecules and then decide which to select. Most methods for dissimilarity-based selection fall into one of two categories: maximum dissimilarity algorithms and sphere exclusion algorithms [Snarey *et al.* 1997].

The maximum dissimilarity algorithm works in an iterative manner; at each step one compound is selected from the database and added to the subset [Kennard and Stone 1969]. The compound selected is chosen to be the one most dissimilar to the current subset. There are many variants on this basic algorithm which differ in the way in which the first compound is chosen and how the dissimilarity is measured. Three possible choices for the initial compound are (a) select it at random, (b) choose the molecule which is 'most representative' (e.g. has the largest sum of similarities to the other molecules) or (c) choose the molecule which is 'most dissimilar' (e.g. has the smallest sum of similarities to the other molecules).

To decide which molecule to add at each iteration requires the dissimilarity values between each molecule remaining in the database and those already placed into the subset to be calculated. Again, this can be achieved in several ways. Snarey *et al.* investigated two common definitions, MaxSum and MaxMin. If there are m molecules in the subset then

the scores for a molecule i using these two measures are given by:

$$\text{MaxSum: score}_i = \sum_{j=1}^{m} D_{i,j} \tag{12.23}$$

$$\text{MaxMin: score}_i = \min(D_{i,j,\,j=1,m}) \tag{12.24}$$

where $D_{i,j}$ is the dissimilarity between two individual molecules i and j. The molecule i that has the largest value of score_i is the one chosen. A useful modification of these two methods is to reject any compound that is too close to one already chosen, typically assessed using the Tanimoto coefficient.

At each iteration of the sphere-exclusion algorithm [Hudson *et al.* 1996], a compound is selected for inclusion in the subset and then all other molecules in the database which have a dissimilarity to this compound less than some threshold value are removed from further consideration. Variation is possible depending upon the way in which the first compound is selected, the threshold value, and the way in which the 'next' compound is selected at each stage. It is typical to try to select this next compound so that it is 'least dissimilar' to those already selected. Hudson *et al.* suggested the use of a 'MinMax' method, where the molecule with the smallest maximum dissimilarity with the current subset is selected. However, it is also possible to select this 'next' compound at random from those still remaining.

The behaviour of some of these methods is illustrated using a two-dimensional example in Figure 12.30. If the 'most dissimilar' compound is chosen as the first molecule in the maximum-dissimilarity cases then the MaxSum method tends to select compounds at the extremities of the distribution. This is also the initial behaviour of the MaxMin approach, but it then starts to sample from the middle. The sphere exclusion methods typically start somewhere in the middle of the distribution and work outwards.

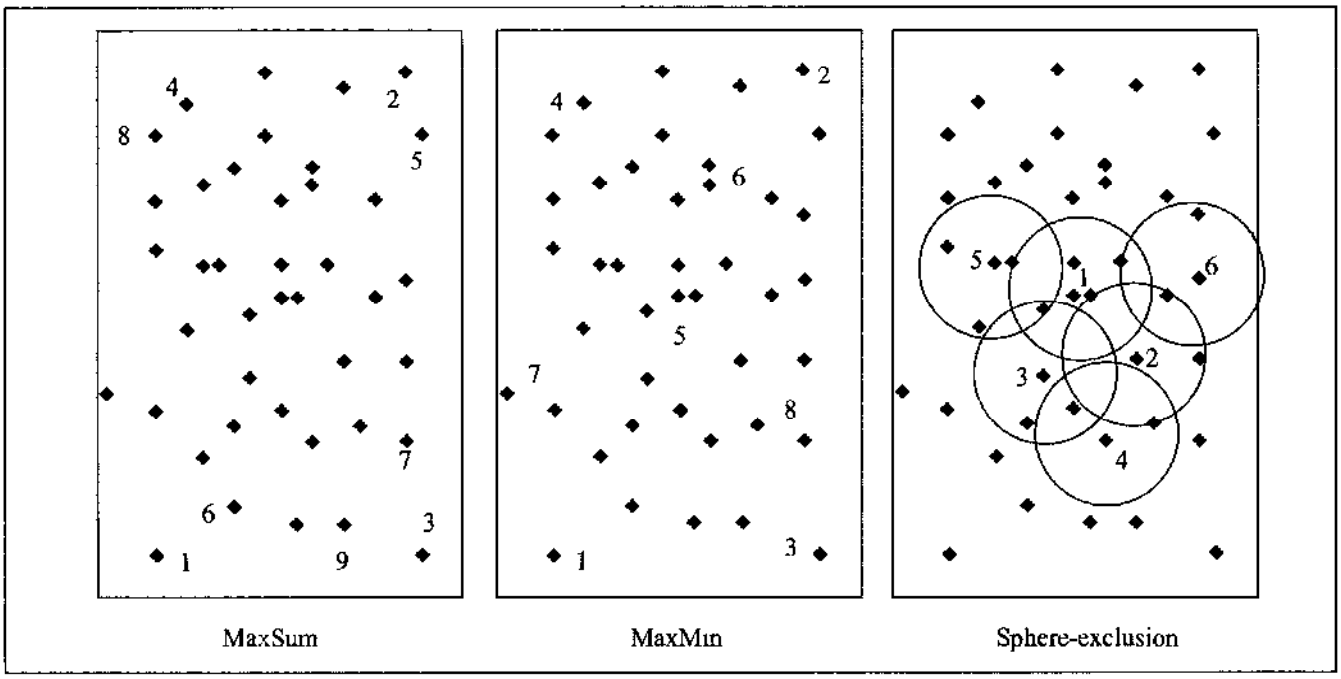


Fig 12 30 Schematic comparison of various dissimilarity selection methods The numbers indicate the order in which the molecules are selected

To compare the behaviour of these various algorithms Snarey *et al.* performed a series of experiments using a set of compounds taken from the World Drug Index, a database that contains thousands of compounds, each with some biological activity. Every molecule in this database is assigned to one or more activity classes (e.g. antibiotics, antihistamines, analgesics). The objective of the exercise was to select subsets of compounds and determine the number of different activity classes. The more activity classes selected the 'better' the performance of the algorithm. In addition to investigating the various dissimilarity measures (MaxSum, MaxMin, etc.) there are a number of ways in which the dissimilarity between any pair of molecules can be quantified (D_{ij} in Equations (12.23) and (12.24)). The molecules were represented by hashed fingerprints or by a series of topological indices and physical properties, with the dissimilarity being given by the complement of either the Tanimoto or the Cosine coefficient. For the Cosine coefficient, a fast procedure is available which enables the dissimilarity between a molecule and a subset to be computed in a single operation rather than having to loop over all molecules currently in the subset [Holiday *et al.* 1995]. There was relatively little to choose between the best of these methods (though some proved significantly worse than a random selection), but the MaxMin maximum-dissimilarity algorithm was generally considered to be effective and efficient.

An alternative to the iterative procedures discussed thus far (where one compound at a time is added to the set) is to select the entire set as a single entity. A standard optimisation procedure such as a Monte Carlo search (often with simulated annealing) can be used to perform this [Hassan *et al.* 1996; Agrafiotis 1997] A set is chosen at random, and an initial value for the diversity function (using an appropriate function, such as MaxMin) is calculated. At each iteration, a small number of compounds are replaced and a new diversity value computed. The change is accepted if the diversity has improved. If not, the Metropolis condition, $\exp[-\Delta E/k_B T]$, is applied (or, alternatively, the Felsenstein formula, $1/(1 + \exp[\Delta E/k_B T])$, which restricts the probability to less than 0.5 and so prevents the system just performing a random walk [Agrafiotis 1997]).

12.10.4 Partition-based Methods for Compound Selection

A major potential drawback with cluster analysis and dissimilarity-based methods for selecting diverse compounds is that there is no easy way to quantify how 'completely' one has filled the available chemical space or to identify whether there are any 'holes'. This is a key advantage of the partition-based approaches (also known as cell-based methods). A number of axes are defined, each corresponding to a descriptor or some combination of descriptors. Each axis is divided into a number of 'bins'. If there are N axes and each is divided into b_i bins then the number of cells in the multidimensional space so created is:

$$\text{Number of cells} = \prod_{i=1}^{N} b_i \qquad (12.25)$$

Each molecule is allocated to one cell according to its values along each axis. It is then a straightforward matter to select a 'representative' set of molecules; one just chooses one

(or more) molecules from each cell. The empty cells correspond to regions of the space not yet covered, which one might wish to target in order to increase the 'diversity' of the set.

The drawback with this is that a relatively low-dimensional space is required, as the number of cells increases exponentially with the number of dimensions, N. For this reason, it is not feasible to employ the binary descriptors that are commonly used to calculate intermolecular distances or dissimilarities (a 1024-long bitstring would contain 2^{1024} cells, an astronomically large number). It is therefore necessary to identify a reasonably low-dimensional space within which to work. This problem is nicely discussed by Lewis, Mason and McLay, who described the use of a partitioned space based upon a variety of molecular descriptors [Lewis *et al.* 1997]. The aim was to find a set of descriptors that would measure six key properties: hydrophobicity, polarity, shape, hydrogen-bonding properties and aromatic interactions. These properties were chosen because of their perceived importance in ligand–receptor interactions. A statistical analysis was performed to identify a set of six weakly correlated descriptors, each of which primarily measured one of the six properties. The partitions for each descriptor were chosen after plotting the distribution of each for approximately 47 000 molecules, after which two, three or four bins were chosen to give approximately equal areas of occupancies. Even such a relatively crude division gives nearly 600 partitions, which if the molecules were evenly distributed (which they are not) would provide about 80 molecules per cell. A representative set of compounds to act as a general-purpose screening set was then determined by selecting three representative molecules per cell, for a total of about 1000 compounds. One reason for the low occupancy of some cells is that they correspond to combinations of properties that are somewhat unlikely to exist (such as a very hydrophobic molecule with many hydrogen-bonding groups).

One approach to the problem of the exponential number of cells is to use principal components analysis or factor analysis to define a smaller set of orthogonal axes that are linear combinations of the original descriptors. This was the approach taken in a comparative study of various chemical databases [Cummins *et al.* 1996]. The initial descriptors comprised the computed free energy of solvation and a large set of topological indices. Descriptors with little variation, or which were highly correlated with another variable, were removed and then a factor analysis was performed. Four factors were able to explain 90% of the variation in the data. This four-dimensional space was partitioned into cells and the distribution of molecules from five databases was computed. Most of the molecules occupied a relatively small region of the space and so an iterative procedure was used to remove outliers, so enabling the resolution to be increased in the area populated by the majority of the molecules. Pairs of databases were compared by counting how many cells they had in common. Two of the databases contained only biologically active molecules and so it was of particular interest to identify the regions of space they occupied.

An alternative to the use of principal components or factor analysis is the BCUT method of Pearlman [Pearlman and Smith 1998]. In this method, three square matrices are constructed for each molecule. Each matrix is of a size equal to the number of atoms in the molecule and has as its elements various atomic and interatomic parameters. One matrix is intended to represent atomic charge properties, another represents atomic polarisabilities and the third hydrogen-bonding capabilities. These quantities can be computed with semi-empirical

quantum mechanical methods, but for large numbers of molecules more approximate methods are recommended. For each matrix, the highest and lowest eigenvalues are computed; for three matrices this gives six values per molecule, which form the axes for the partitioned space.

Finally, 3D pharmacophores can be used to provide a naturally partitioned space. By combining the pharmacophore keys of a set of molecules one can determine how many of the potential 3- or 4- point pharmacophores are accessible to the set and easily identify those which are not represented. This use of pharmacophores is the basis of a method named 'Pharmacophore-Derived Queries' (PDQ) [Pickett *et al.* 1996]. One feature of this particular method is that most molecules will occupy more than one 'cell' (as nearly all molecules will contain more than one 3-point pharmacophore due to the functionality present and conformational flexibility). This contrasts with the usual situation, wherein each molecule occupies just one cell

12.11 Structure-based *De Novo* Ligand Design

Database searching is an attractive way to discover new lead compounds; in favourable cases the hits can be tested immediately or the molecule can be synthesised using a published method. However, database searching does not provide molecules that are structurally 'novel', although their new-found activity may be. Moreover, many databases are biased towards particular classes of compounds, so limiting the range of structures that can be obtained. In *de novo* design, the three-dimensional structure of the receptor or the 3D pharmacophore is used to design new molecules. There are two basic types of *de novo* design algorithm. The first class of methods have been described as 'outside-in' methods [Lewis and Leach 1994]. Here, the binding site is first analysed to determine where specific functional groups might bind tightly. These groups are connected together to give molecular skeletons, which are then converted into 'real' molecules. In the 'inside-out' approach, molecules are grown within the binding site under the control of an appropriate search algorithm, with each suggestion being evaluated using an energy function. These two approaches are compared in Figure 12.31.

12.11.1 Locating Favourable Positions of Molecular Fragments Within a Binding Site

One of the most widely used tools in structure-based ligand design is the GRID program [Goodford 1985]. A regular grid is superimposed upon the binding site. A probe group is then placed at the vertices of the grid and the interaction energy of the probe with the protein is determined using an empirical energy function. The result is a three-dimensional grid with an energy value at each vertex; this data can then be analysed to find those locations where it might be favourable to position a particular probe An example of the output produced by GRID is shown in Figure 12.32 (colour plate section) for the binding site of neuraminidase. Parameters for many probes have been developed, covering a variety of small molecules and common functional groups. An alternative to the use of a grid is to

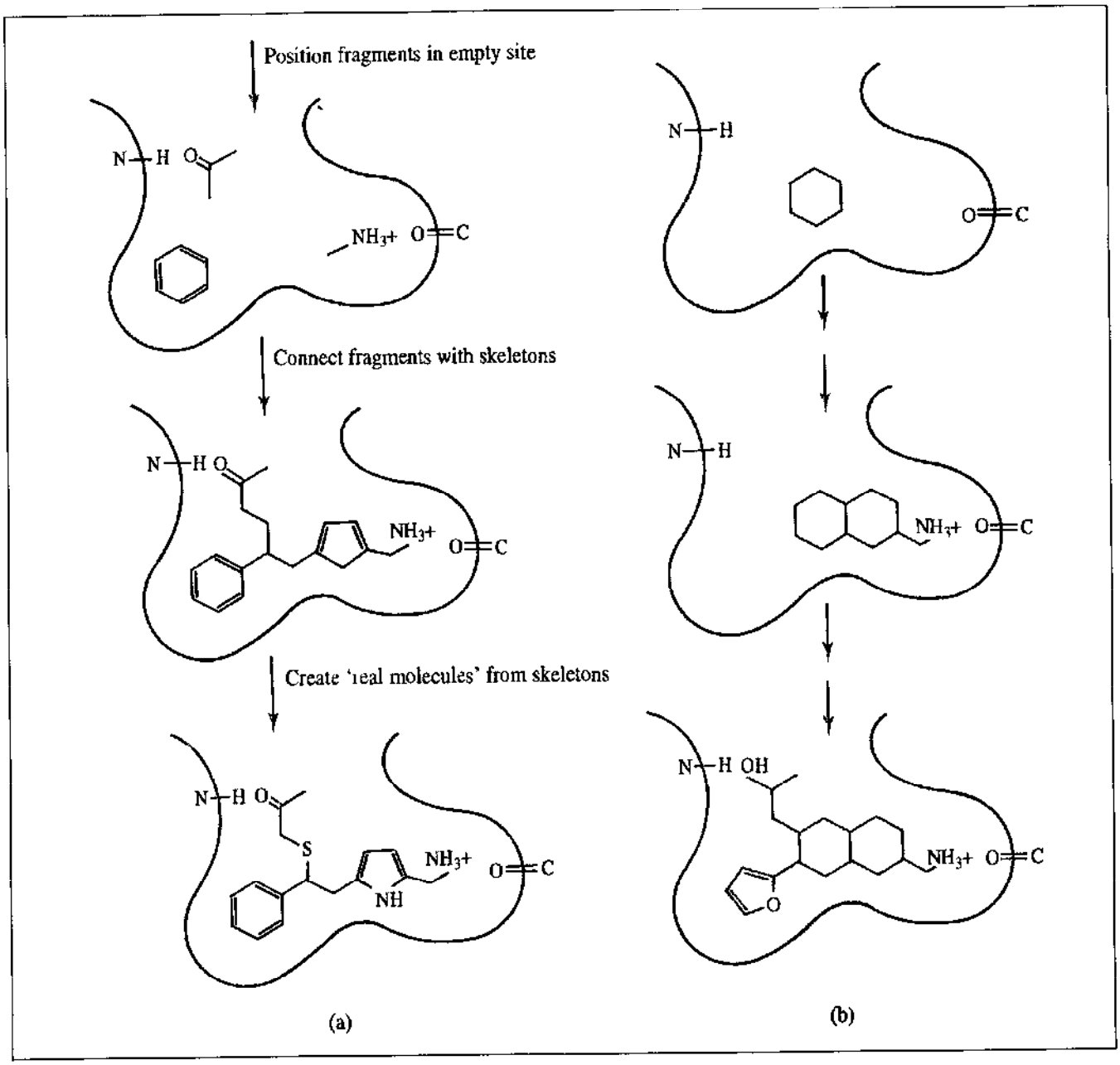


Fig. 12.31 Approaches to de novo *design: (a) outside in, (b) inside out*

permit each fragment to explore the entire binding site using energy minimisation or some form of simulation method. In the multiple-copy simultaneous search (MCSS) approach [Miranker and Karplus 1991] the binding site is initially filled with many copies of the same fragment, distributed randomly. A molecular mechanics energy model is used in which the protein interacts simultaneously with all of the fragments, but there are no interactions between the individual fragments. Energy minimisation is then used to try to identify energetically favourable positions. The minimisation is done in a series of stages, with the orientations of the fragments being clustered at the end of each stage to remove duplicates.

An alternative to the energy-based methods such as GRID is to suggest possible binding positions using a knowledge-based approach. Analysis of experimentally determined

structures of ligand–receptor complexes reveals that they often contain certain types of interaction. For example, many ligands form hydrogen bonds with their receptors. The knowledge-based approaches generate binding modes that contain these commonly observed interactions, with the fragments being positioned to reproduce the most commonly observed geometries. For example, in most hydrogen bonds the distance between the donor hydrogen and its acceptor is close to 1.8 Å and the angle subtended at the hydrogen is rarely less than 120°. Information about the preferred geometries of such interactions can be obtained from analyses of X-ray crystallographic databases (described in Section 9.11). A program called LUDI has been widely used to dock small molecular fragments in protein binding sites using such an approach [Böhm 1992].

The knowledge-based docking approach to ligand design requires the receptor site to be surveyed to identify possible hydrogen-bonding donor and acceptor sites and regions where other groups might favourably be positioned. The results of such an analysis are often converted into a distribution of *site points*. A site point is a location within the binding site where an appropriate ligand atom or group could be placed. For example, a hydrogen-bonding analysis would typically result in a series of donor and acceptor site points. When generating the site points one should take account of any preferred geometries for that particular type of interaction. There is usually more than one site point associated with each donor or acceptor atom in the receptor to reflect the fact that a distribution of geometries is found in the crystal structure analyses. The range of preferred geometries can also be represented as a continuous region. Having surveyed the binding site, each molecular fragment is examined to determine which features it contains, and the fragment is positioned in the site by fitting the appropriate atoms to their corresponding site points.

The natural binding affinity of the small molecules typically used in a LUDI-type search is invariably rather low, and so it has been difficult to assess the accuracy of such computational techniques properly. However, it is now possible to use either X-ray crystallography or NMR to determine where such fragments bind to the protein. The technique of SAR-by-NMR in particular has attracted much attention [Shuker *et al.* 1996]. In SAR-by-NMR, a mixture of several highly soluble small molecules and a ^{15}N-labelled protein is analysed using NMR. It is then possible to identify not only which small molecules bind but also where. Moreover, having identified one binder others can be found by repeating the assay with the protein together with the first 'hit', so identifying pairs of fragments that bind in a synergistic fashion.

12.11.2 Connecting Molecular Fragments in a Binding Site

Having positioned molecular fragments in favourable positions within the binding site using either an energy scheme or a knowledge-based approach, the next stage is to connect the fragments together into 'real' molecules. One way to tackle this problem is by searching a database of molecular connectors. Bartlett's CAVEAT program was one of the first methods for tackling this problem [Lauri and Bartlett 1994]. The relationship between each pair of fragments can be considered in terms of two vectors that represent the bonds they make

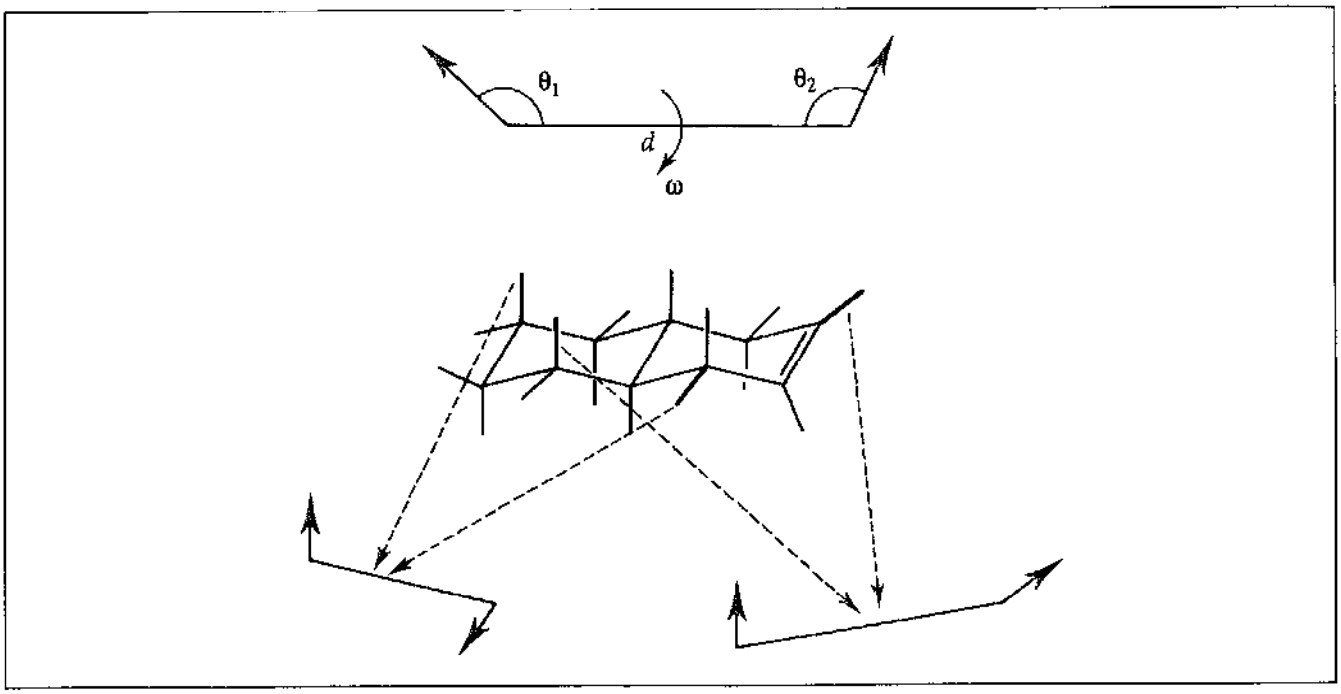


Fig. 12.33: The relationship between two bond vectors can be represented using a distance, two angles and a torsion angle as indicated (top). To derive the data for the database all possible pairs of exocyclic vectors are considered and these four geometric parameters calculated.

to the linker. The geometrical relationship between each pair of linking vectors can be described using a distance, two angles and a dihedral angle as shown in Figure 12.33. CAVEAT searches its database to find connectors that also contain two bond vectors with the same geometric parameters. The data is stored in an efficient form in the database, and so the search is very rapid. The connectors used by the first versions of CAVEAT consisted of ring systems extracted from the Cambridge Structural Database. The four geometric parameters were calculated between all pairs of bond vectors exocyclic to the ring, as shown in Figure 12.33. Connectors can also be generated using structure-generation programs.

Not all problems are amenable to the database searching approach exemplified by CAVEAT; the molecular fragments may be further apart than the longest connector in the database. Moreover, with such an approach one is inherently restricted to the fragments contained in the database. An alternative strategy is to create a skeleton that connects the fragments. The skeleton can be grown one atom at a time or by joining molecular templates. The templates typically comprise rings and acyclic fragments commonly found in drug molecules [Gillett *et al.* 1993].

The final stage in the outside-in approach to *de novo* design is to create 'real' molecules from the molecular skeletons that are generated. This is the most difficult part of the procedure, as there are so many ways in which even a small set of atom types can be assigned to a skeleton. The objective is to produce molecules that can be synthesised relatively easily and that also enhance the binding of the ligand to the binding site. It is very difficult to incorporate the concept of 'ease of synthesis' into a computer program, though some attempts have been made [Myatt 1995].

As can be seen from Figure 12.31, the outside-in procedure breaks the problem into a number of distinct stages. The inside-out method grows a ligand within the site in one step. Such an approach has been used successfully to generate peptides in protein binding sites [Moon and Howe 1991]. In this case, ligands are constructed from templates that are low-energy conformations of amino acids. Both systematic and random search algorithms can be used to explore the space of possible combinations of templates. In the systematic search, all of the amino acid building blocks are added to the growing ligand in turn. Each new structure is checked to ensure that it does not interact unfavourably with the protein and that it does not contain any high-energy intramolecular interactions. A molecular mechanics energy is then calculated for the structure. It is impractical to keep all of the structures from one stage to the next due to the combinatorial explosion and so only the lowest-energy structures are retained for the next iteration. Alternatively, a Monte Carlo simulated annealing search can be performed in which the Metropolis criterion is used at each stage to decide whether to accept or reject a given structure, based upon its energy and that of its predecessor. Genetic algorithm approaches have also been used to explore the search space [Glen and Payne 1995]. The advantage of applying this method to peptides is that there is a defined way of connecting the building blocks together, and the synthesis of peptides is straightforward. It is more difficult to generate general 'organic' molecules.

12.11.3 Structure-based Design Methods to Design HIV-1 Protease Inhibitors

An impressive example of the application of structure-based methods was the design of an inhibitor of the HIV protease by a group of scientists at DuPont Merck [Lam *et al.* 1994]. This enzyme is crucial to the replication of the HIV virus, and inhibitors have been shown to have therapeutic value as components of anti-AIDS treatment regimes. The starting point for their work was a series of X-ray crystal structures of the enzyme with a number of inhibitors bound. Their objective was to discover potent, novel leads which were orally available. Many of the previously reported inhibitors of this enzyme possessed substantial peptide character, and so were biologically unstable, poorly absorbed and rapidly metabolised.

The X-ray structures of the HIV protease revealed several key features that were subsequently incorporated into the designed inhibitor. The enzyme is a dimer with C_2 symmetry. It is a member of the aspartyl protease family, with the two aspartate residues lying at the bottom of the active site. Many of the crystal structures contained a tetra-coordinated water molecule that accepted two hydrogen bonds from the backbone amide hydrogens of two isoleucine residues in the 'flaps' of the enzyme and donated two hydrogen bonds to the carbonyl oxygens of the inhibitor (Figure 12.34, colour plate section).

A flow chart showing the various phases leading to the final compound is reproduced in Figure 12.35. The first step was a 3D database search of a subset of the Cambridge Structural Database. The pharmacophore for this search comprised two hydrophobic groups and a

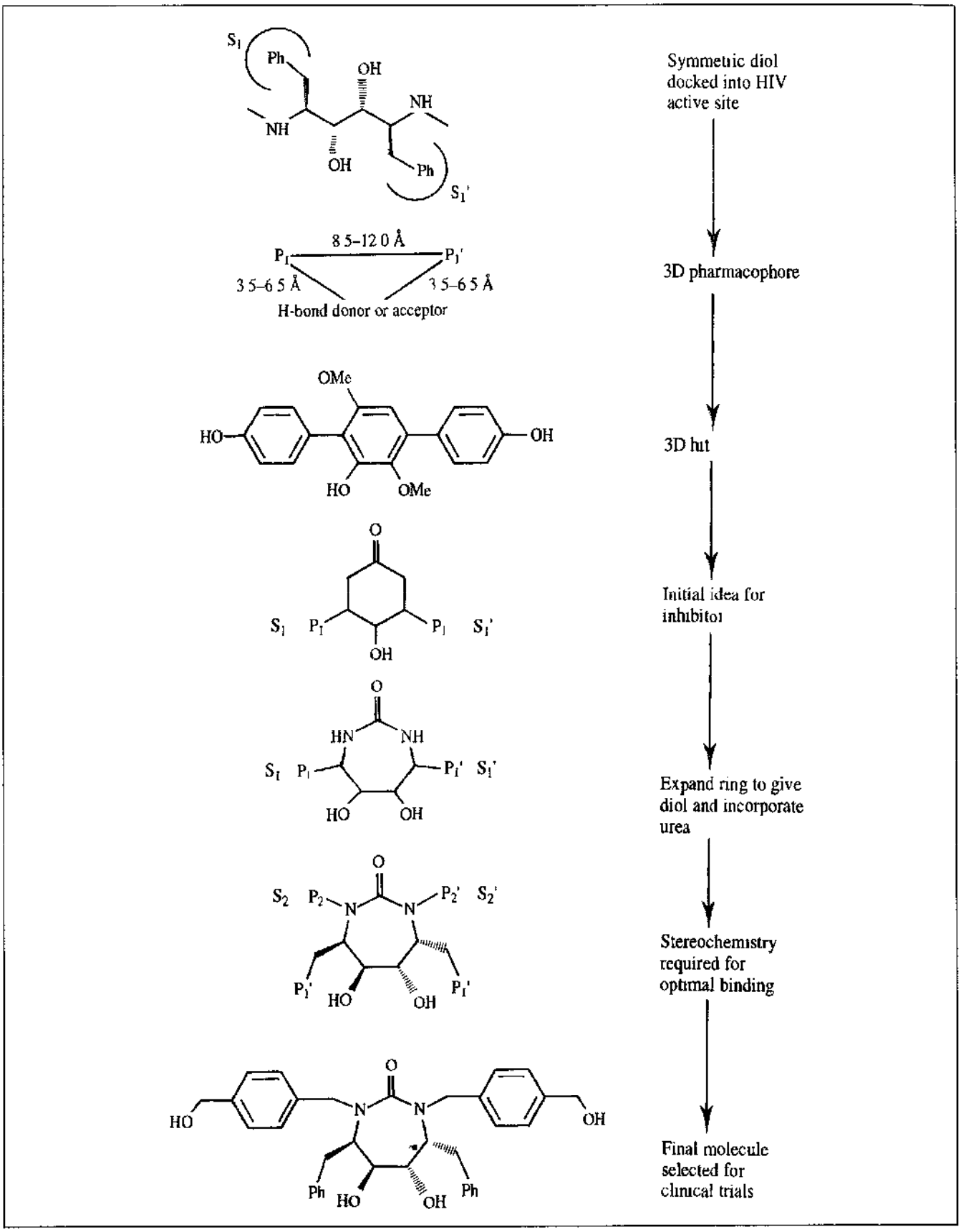


Fig 12.35: Flow chart showing the design of novel orally active HIV-1 protease inhibitor (Figure adapted from Lam P Y S, P K Jadhav, C E Eyermann, C N Hodge, Y Ru, L T Bacheler, J L Meek, M J Otto, M M Rayner, Y N Wong, C-H Chang, P C Weber, D A Jackson, T R Sharpe and S Erickson-Viitanen 1994. Rational Design of Potent, Bioavailable, Nonpeptide Cyclic Ureas as HIV Protease Inhibitors Science **263** *380–384)*

hydrogen-bond donor or acceptor. The hydrophobic groups were intended to bind in two hydrophobic pockets (the S_1 and S_1' pockets) and the hydrogen-bond donor or acceptor to bind to the catalytic aspartate residues. The search yielded the hit shown in Figure 12.35. This molecule not only contained the desired elements of the pharmacophore but it also had an oxygen atom that could displace the bound water molecule Displacement of the water was expected to be energetically favourable due to the increase in entropy. The benzene ring in the original compound was changed to a cyclohexanone, which was able to position the substituents in a more appropriate orientation

The DuPont Merck group had previously explored a series of peptide-based diols that were potent inhibitors but with poor oral bioavailability. They were keen to retain the diol functionality, and so the next step was an expansion of the ring to a seven-membered diol. The ketone was then changed to a cyclic urea to strengthen the hydrogen bonds to the flaps and to aid the synthesis. Further modelling studies based upon the X-ray structure were performed to predict the optimal stereochemistry and the conformation required for optimal interaction with the enzyme. The results of these studies showed that the 4*R*, 5*S*, 6*S*, 7*R* configuration was most appropriate. Nitrogen substituents were predicted to bind to the S_2 and S_2' pockets of the enzyme, and so various analogues were synthesised in order to enhance the potency whilst maintaining the desired pharmacological properties. The compound eventually chosen for further studies, leading to clinical trials, was a *p*-hydroxymethylbenzyl derivative (Figure 12.35).

An increasing number of case histories of structure-based design have now been published in the literature, reflecting the widespread use of protein structures for drug design [Babine and Bender 1997, Kubinyi 1998]. All of the major pharmaceutical companies use structure-based design methods as part of their search for new drugs. Indeed, some smaller companies focus exclusively on structure-based design.

12.11.4 Structure-based Design of Templates for Zeolite Synthesis

Although *de novo* design is most commonly associated with the design of biologically active molecules, it is not restricted to this area. One particularly interesting application is in the design of templating agents for the synthesis of microporous materials, such as zeolites. These materials are very important for processes such as catalysis, ion exchange and gas separation. Zeolites can be considered to have the general formula TO_2, where T is a tetrahedrally coordinated atom (silicon, aluminium or phosphorus, depending on the constitution). The variable constitution can result in the T sites having an average oxidation state less than +4, giving a net negative charge. The charge imbalance can be rectified by the presence of Na^+ ions. These then exchange with NH_4^+ ions, which donate protons to the framework oxygen atoms to give ammonia, leaving the zeolite as a solid Brønsted acid catalyst.

The synthesis of zeolites is traditionally performed by crystallisation from a sol–gel mixture comprising reagents such as silica, sodium aluminate, sodium hydroxide and water. Another key component of the sol–gel mixture is a base whose main role is to regulate the pH of the mixture. If an organic base is used then a *templating* effect may also be observed

whereby the base acts to control the shape and size of the zeolite pores. There are many factors involved in the templating process, but qualitatively at least it is necessary that there is a good fit between the base and the framework such that the template fills the empty space in the zeolite cavity.

Traditionally, the templates were chosen by trial and error or exhaustive enumeration. A computational method named ZEBEDDE (ZEolites By Evolutionary De novo DEsign) has been developed to try to introduce some rationale into the selection of templates [Lewis *et al.* 1996; Willock *et al.* 1997]. The templates are grown within the zeolite cavity by an iterative 'inside-out' approach, starting from a seed molecule. At each iteration an action is randomly selected from a list that includes the addition of new atoms (from a library of fragments), random translation or rotation, random bond rotation, ring formation or energy minimisation of the template. A cost function based on the overlap of van der Waals spheres is used to control the growth of the template molecule:

$$f = \frac{\sum_{i=1}^{N_t} d(i, \text{host})}{N_t} \tag{12.26}$$

where $d(i, \text{host})$ is the closest contact distance between the template atom i and its nearest host atom The function is normalised by the number of atoms in the template, N_t. In addition, the template is checked to ensure that there are no unfavourable intramolecular contacts or conflicts with any of its symmetrically related images (it is possible to have more than one template molecule per unit cell). The procedure continues so long as the cost function decreases and stops when it has fallen below a predefined value. Different template molecules can be compared by their ability to minimise the cost function.

The molecules typically used for templating agents are relatively simple and this is reflected in the fragment library (in the published examples just nine fragments were used: methane, ethane, ammonia, benzene, ammonium, propane, pyrrole, adamantane and cyclohexane [Willock *et al.* 1997]). The cost function does not have an electrostatic component and so some restrictions were imposed on the number of nitrogen atoms (no more than two per molecule and no N–N bonds). The type of molecule that can be produced is obviously governed by the template library but can also be biased by applying different weighting schemes for the fragment addition step. Fragments are typically added by replacement of a hydrogen atom in the template molecule. Giving a higher weight to the hydrogen atoms in new fragments tends to encourage the growth of linear chains, whereas a lower weighting tends to produce highly substituted template molecules. The method was applied to zeolites for which templates were already known, to check that these could be 'discovered' and also to determine what types of structure were produced by the various weighting schemes. In general, the known templates were found (or at least close analogues) together with some new possibilities. However, unfavourable conformations were observed in some of the suggestions, but these could sometimes be relieved by forming a ring. Other suggested molecules were not commercially available, despite the limited fragment library. For these targets as well as the biological ones the synthetic accessibility of *de novo* designed molecules is a major issue

12.12 Quantitative Structure–Activity Relationships

A quantitative structure–activity relationship (QSAR) relates numerical properties of the molecular structure to its activity by a mathematical model. The term 'quantitative structure–property relationship' (QSPR) is also used, particularly when some property other than biological activity is concerned. In drug design, QSAR methods have often been used to consider qualities beyond *in vitro* potency. The most potent enzyme inhibitor is of little use as a drug if it cannot reach its target. The *in vivo* activity of a molecule is often a composite of many factors. A structure–activity study can help to decide which features of a molecule give rise to its overall activity and help to make modified compounds with enhanced properties. The relationship between these numerical properties and the activity is often described by an equation of the general form:

$$v = f(p) \tag{12.27}$$

where v is the activity in question, p are structure-derived properties of the molecule (i.e. descriptors), and f is some function. An early example of a structure–activity relationship was the discovery by Meyer and Overton of a correlation between the potencies of narcotics and the partition coefficient of the compounds between oil and water. Overton's interpretation was that the narcotic effect is due to physical changes caused by the dissolution of the drug in the lipid component of cells.

The first use of QSARs to rationalise biological activity is usually attributed to Hansch [Hansch 1969]. He developed equations which related biological activity to a molecule's electronic characteristics and hydrophobicity For example:

$$\log(1/C) = k_1 \log P - k_2(\log P)^2 + k_3\sigma + k_4 \tag{12.28}$$

where C is the concentration of the compound required to produce a standard response in a given time, $\log P$ is the logarithm of the partition coefficient of the compound between 1-octanol and water, which was chosen by Hansch as a suitable measure of relative hydrophobicity, σ is the Hammett substituent parameter and k_1–k_4 are constants.

The hydrophobic component was considered to model the ability of the drug to pass through cell membranes. Hansch recognised that there is an optimal value of the hydrophobicity: too low and the drug would not partition into the cell membrane; too high and the compound would partition into the membrane but tend to remain there rather than proceeding to the actual target. This explains the parabolic dependence of the activity upon $\log P$. An alternative way to express the Hansch equation uses a parameter π. This is the logarithm of the partition coefficient of a compound with substituent X relative to a parent compound in which the substituent is hydrogen:

$$\pi = \log(P_X/P_H) \tag{12.29}$$

Thus

$$\log(1/C) = k_1\pi - k_2\pi^2 + k_3\sigma + k_4 \tag{12.30}$$

The Hammett substituent parameter was used by Hansch as a concise measure of the electronic characteristics of the molecules. Hammett and others (such as Taft) showed that

the positions of equilibrium and the reaction rates of series of related compounds such as substituted benzoic acids could be expressed in the following way:

$$\log\left(\frac{k}{k_0}\right) = \rho\sigma \quad \text{or} \quad \log\left(\frac{K}{K_0}\right) = \rho\sigma \tag{12.31}$$

where k_0 and K_0 are the rate constant and equilibrium constant, respectively, for a 'reference' compound (usually a hydrogen-substituted compound). The substituent parameter σ depends only upon the nature of the substituent and whether it is *meta* or *para* to the carboxyl group. The reaction constant ρ is fixed for a given process under specified experimental conditions. The 'standard' reaction is the dissociation of benzoic acids, which have $\rho = 1$. A full discussion of linear free-energy relationships can be found in many physical organic chemistry textbooks. A reading of such material will reveal that many modifications to the original Hammett scale have been suggested. One important development was the introduction by Swain and Lupton of the field (F) and resonance (R) components [Swain and Lupton 1968; Swain *et al.* 1983]. It was suggested that any set of σ values could be expressed as a weighted linear combination of these two components. This greatly simplified the problem of selecting the 'correct' set of substituent values for any individual system.

An enormous number of QSAR equations have been reported in the literature, many having a functional form much more complicated than the original Hansch equation. Many different parameters have been used in QSAR equations, designed to represent the hydrophobic, electronic or steric characteristics of the molecule. The properties chosen for inclusion in the QSAR equation should be as uncorrelated with each other as possible. Many of the early QSARs were derived for sets of related series of compounds that differed in just one part of the molecule. These differences can often be characterised using appropriate substituent constants, which were available in published tables. There has been a trend in recent years towards the analysis of noncongeneric series of compounds, for which there is no 'parent' structure and the consequent use of whole-molecule descriptors that are calculated directly. Some of these are given in Table 12.2 but there are many others. An example would be molecular shape analysis, which includes descriptors that measure the relative shape of the compounds [Rhyu *et al.* 1995]. A conformational analysis of the compounds is performed to identify the minimum energy structures. These conformations are then overlaid on the reference structure (usually one of the most active compounds in the series). From these overlaid structures it is then possible to calculate the common overlap volume and the non-overlap volume, which can then be included in the QSAR equation together with other parameters.

Another type of 'parameter' that often appears in published QSAR equations is an *indicator variable*. Indicator variables are used to extend a QSAR equation over a variety of different types of molecule and so make the equation more generally applicable. For example, Hansch and colleagues derived the following equation for the binding constants of sulphonamides ($X\text{-}C_6H_4\text{-}SO_2NH_2$) to human carbonic anhydrase [Hansch *et al.* 1985]:

$$\log K = 1.55\sigma + 0.64 \log P - 2.07 I_1 - 3.28 I_2 + 6.94 \tag{12.32}$$

I_1 takes the value 1 for *meta* substituents (0 for others) and I_2 is 1 for *ortho* substituents (0 for others).

12.12.1 Selecting the Compounds for a QSAR Analysis

The derivation of a QSAR equation involves a number of distinct stages. First, it is obviously necessary to synthesise the compounds and determine their biological activities. When planning which compounds to synthesise, it is important to cover the range of properties that may affect the activity. This means applying the data-checking and -manipulation procedures discussed earlier. For example, it would be unwise to make a series of compounds with almost identical partition coefficients if this is believed to be an important property.

Experimental design techniques can be used to help to decide which compounds to synthesise to ensure that the most information can be extracted from the smallest number of molecules. A variety of experimental design methods have been devised, of which the most straightforward to understand is probably full factorial design. Suppose there are two variables (formally called *factors*) that might influence the outcome (often called the *response*) of an experiment. If the experiment were a chemical synthesis, the factors might be temperature and the pH, with the response being the product yield. In our case, the experiment might involve the inhibition of an enzyme where the factors could be the molecule's log P and the Hammett substituent parameter. The response could be the degree of inhibition, measured as the IC50 (a commonly used measure of binding affinity, being the concentration of inhibitor to reduce the binding of a ligand or the rate of reaction by one-half). Determination of an IC50 requires less data than to determine the dissociation equilibrium constant but, unlike equilibrium constants, IC50 values measured under different conditions or for different receptors cannot usually be compared. Suppose, moreover, that we are interested in just two different values of each factor. Four possible experiments could be performed; in the case of the chemical synthesis these would be T_1pH_1, T_2pH_1, T_1pH_2, T_2pH_2, where T_1 and T_2 are the two temperatures and pH_1 and pH_2 are the two pH values. The first three experiments would measure the effect of changing just one variable at a time, whereas the fourth experiment (T_2pH_2) measures the effect of changing both variables and could indicate possible interactions between the factors. If there were three factors (with two values for each) then such a *full factorial design* would involve $2^3 = 8$ experiments. With three variables there is the possibility of interactions between pairs of factors, or between all three factors. In general, it is found that single factors on their own are more important than pairwise interactions, which are more important than three-factor effects, and so on. A *fractional factorial design* involves fewer experiments than the full factorial design. A half-fractional factorial design involves half the number of experiments as are in the full factorial design; a quarter-fractional design involves one-quarter of the experiments. However, it may be less easy to determine unambiguously the most important factors or combinations of factors from a fractional factorial design.

Factorial design methods cannot always be applied to QSAR-type studies. For example, it may not be practically possible to make any compounds at all with certain combinations of factor values (in contrast to the situation where the factors are physical properties such as temperature or pH, which can be easily varied). Under these circumstances, one would like to know which compounds from those that *are* available should be chosen to give a well-balanced set with a wide spread of values in the variable space. *D-optimal design* is one technique that can be used for such a selection. This technique chooses subsets of

molecules from those that are possible in such a way as to maximise the determinant of the variance–covariance matrix (also referred to as the 'information matrix'). If **A** is a matrix with n rows (corresponding to n molecules) and p columns (corresponding to the p descriptors) then the variance–covariance matrix is $\mathbf{AA}^{\mathrm{T}}$ and is of size $n \times n$. D-optimal design aims to find the subset of n molecules that optimises the determinant of this matrix A maximal value of the determinant corresponds to maximum variance and minimum covariance. In other words, the molecules have a wide spread of values for the descriptors (large variance) but a small degree of correlation between them (small covariance).

12.12.2 Deriving the QSAR Equation

The most widely used technique for deriving QSAR equations is *linear regression*, which uses least-squares fitting to find the 'best' combination of coefficients in the QSAR equation (the technique is also referred to as ordinary least-squares). We can illustrate the least-squares technique using the simple case where the activity is a function of just one property (when the technique is known as simple linear regression). We therefore want to derive an equation of the form:

$$y = mx + c \tag{12.33}$$

where y is known as the dependent variable (the observations) and x is the independent variable (the parameters). For example, y might be the activity and x might be $\log P$. The objective of a regression analysis is to find the coefficients m and c that minimise the sum of the deviations of the observations from the fitted equation, as shown in Figure 12.36. The least-squares coefficients m and c in the linear regression equation (12 33) are given by:

$$m = \frac{\sum_{i=1}^{n}(x_i - \langle x\rangle)(y_i - \langle y\rangle)}{\sum_{i=1}^{n}(x_i - \langle x\rangle)^2}; \quad c = \langle y\rangle - m\langle x\rangle \tag{12.34}$$

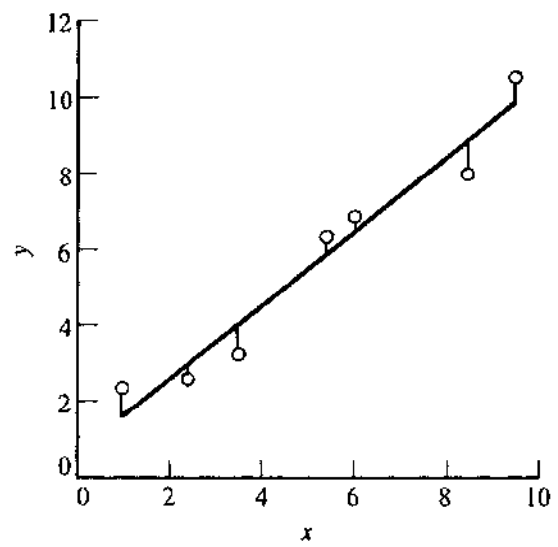


Fig 12 36: The regression equation is the best-fit line through the data that minimises the sum of the deviations

The regression equation passes through the point $(\langle x \rangle, \langle y \rangle)$, where $\langle x \rangle$ and $\langle y \rangle$ are the means of the dependent and independent variables, respectively. The 'quality' of a simple linear regression equation is often reported as the squared correlation coefficient, or r^2 value. This indicates the fraction of the total variation in the dependent variables that is explained by the regression equation. To determine r^2, the total sum of squares (TSS) of the deviations of the observed y values from the mean $\langle y \rangle$ is calculated together with the explained sum of squares (ESS), which is the sum of squares of the deviations of the y values calculated from the model, $y_{\text{calc},i}$, from the mean:

$$\text{TSS} = \sum_{i=1}^{N} (y_i - \langle y \rangle)^2; \quad \text{ESS} = \sum_{i=1}^{N} (y_{\text{calc},i} - \langle y \rangle)^2; \quad \text{RSS} = \sum_{i=1}^{N} (y_i - y_{\text{calc},i})^2 \tag{12.35}$$

$y_{\text{calc},i}$ is obtained by feeding the appropriate x_i value into the regression equation. Another common squared term is the residual sum of squares (RSS), which is the sum of squares of the differences between the observed and calculated y values. TSS is equal to the sum of RSS and ESS. The r^2 is then given by:

$$r^2 = \frac{\text{ESS}}{\text{TSS}} \equiv \frac{\text{TSS} - \text{RSS}}{\text{TSS}} = 1 - \frac{\text{RSS}}{\text{TSS}} \tag{12.36}$$

r^2 can adopt values between 0.0 and 1.0; a value of 0.0 indicates that none of the variation in the observations is explained by variation in the independent variables, whereas a value of 1.0 indicates that all of the variation in the observations can be explained. A disadvantage of the standard r^2 value is that it is dependent upon the number of independent variables, with higher r^2 values being obtained for larger data sets. More sophisticated statistical measures should ideally be used. These can, for example, help to determine whether the addition of a particular descriptor contributes significantly to the model.

It is straightforward to extend this analysis to more than one independent variable (known as *multiple linear regression*), such calculations are tedious to perform by hand but can be performed using a statistical package. Whereas simple linear regression involves fitting a straight line to the data, multiple linear regression corresponds to fitting a multidimensional surface The quality of the regression is indicated by the multiple correlation coefficient, which we will write as R^2 (lowercase r^2 is also used). Another quantity that is commonly reported is the F statistic. This is the ratio of the explained mean square divided by the residual mean square. Values of F are available in statistical tables at different levels of confidence; if the calculated value is greater than the tabulated value then the equation is said to be significant at that particular level of confidence. It is important to note that the value of F depends upon the number of independent variables in the equation and the number of data points. As the number of data points increases and/or the number of independent variables falls so the value of F which corresponds to a particular confidence level also decreases. This is because we would like to be able to explain a large number of data points with an equation containing as few variables as necessary; such an equation would be expected to have greater predictive power. This is formally taken into account via the number of *degrees of freedom* associated with each parameter. A simple or multiple linear regression is associated with $N - 1$ degrees of freedom because the fitted line always passes through the means of the dependent and independent variables. The total

sum of squares is associated with $N-1$ degrees of freedom. If there are p independent variables in the equation then there are $N-p-1$ degrees of freedom associated with the residual sum of squares and p degrees of freedom associated with the explained sum of squares. Thus the explained mean square equals ESS divided by p, and the residual mean square equals RSS divided by $N-p-1$, and so F is given by:

$$F = \frac{\text{ESS}}{p} \frac{N-p-1}{\text{RSS}} \tag{12.37}$$

As to the significance of the individual terms in the equation, this can be assessed using the t statistic, which is obtained by dividing the relevant regression coefficient by the standard error of the coefficient. If k is the regression coefficient associated with a variable x, then the t statistic is obtained as follows:

$$t = \left|\frac{k}{s(k)}\right|, \quad s(k) = \sqrt{\frac{\text{RSS}}{N-p-1} \frac{1}{\sum_{i=1}^{N}(x_i - \bar{x})^2}} \tag{12.38}$$

The value of t is compared with tabular values, which are listed according to the number of degrees of freedom associated with the residual sum of squares and for various significance levels. If the computed value is larger than the tabulated number then the coefficient can be considered significant.

There are some important criteria to consider when using multiple linear regression. To achieve statistically significant results there should be sufficient data; it is often considered that at least five compounds are required for each descriptor included in the regression analysis. The various checks on the data described in Section 12.10 should be performed to ensure that the selected compounds have a good spread of descriptor values, which should be as uncorrelated as possible. Compounds which have a value for some descriptor that is greatly different from the remainder (i e. a significant outlier) should be examined very closely.

However, it is not sufficient to identify a set of non-correlated, well-distributed parameters and then simply 'press the button' and derive a multiple linear regression equation. To derive a QSAR equation properly requires a lot of care. Some of the descriptors may have little or no relevance to the property being modelled. Moreover, one generally wants to achieve a balance between an equation that captures the essence of the problem and yet is predictive. Fortunately, there are a number of procedures that can help with some of these problems.

One of the problems in deriving a QSAR equation is in deciding which descriptors to use. Two related procedures, forward-stepping regression and backward-stepping regression, can help in this. As the names imply, forward-stepping regression starts with an equation involving just one variable (the one that makes the most contribution, typically assessed using the t value). The second and subsequent terms are then added, again choosing the descriptor which makes the most contribution. Backward-stepping regression works in the reverse sense; initially, an equation is derived using all the descriptors, which are then removed (e.g the one with the smallest t statistic). For both forward- and backward-stepping regression the equation finally chosen may be the one with the best fit to the data (as might be assessed using the F value)

Genetic algorithms can also be used to derive QSAR equations [Rogers and Hopfinger 1994]. The genetic algorithm is supplied with the compounds, their activities and information about their properties and other relevant descriptors. From this data, the genetic algorithm generates a population of linear regression models, each of which is then evaluated to give the fitness score. A new population of models is then derived using the usual genetic algorithm operators (see Section 9.9.1), with the parameters in the models being selected on the basis of the fitness. Unlike other methods, the genetic algorithm approach provides a family of models from which one can either select the model with the best score or generate an 'average' model.

12.12.3 Cross-validation

Cross-validation is a widely used and strongly recommended technique for checking the quality of a regression model. The technique (also known as jack-knifing) involves removing some of the values from the data set, deriving a regression model for the remainder and then predicting the values for the data left out. The most common form of cross-validation is 'leave-one-out', in which each data value is left out in turn and a model derived using the remainder of the data. A value can then be predicted for the data left out and compared with the true observed value. This is repeated for every data point in the set and permits the calculation of a 'cross-validated R^2' value (also written R^2_{cv} or Q^2, or their lowercase equivalents). Cross-validated R^2 values are typically lower than the normal R^2 values but are considered more indicative of the predictive ability of the equation. Indeed, Q^2 can have negative values (unlike R^2). Thus, whereas R^2 is a measure of goodness of fit, Q^2 is a measure of goodness of prediction. A more robust alternative to leave-one-out is to divide the data set into four or five groups, each of which is left out in turn to perform a cross-validation experiment. This process can be repeated many times (100 is typical) using different, randomly selected groups to obtain a mean Q^2. The 'final' model (which might be used to predict the behaviour of as yet untested compounds) can then be derived in the normal way from all of the data; however, for a well-behaved system one would not expect the regression coefficients to change very much during the jack-knife procedure. Moreover, if the R^2 value from the whole data set is significantly larger than the mean Q^2 from the cross-validation experiment, then it is likely that the data has been over-fit. Another measure of predictive ability is the predictive residual sum of squares, PRESS, which is calculated in the same manner as the residual sum of squares except that, rather than using values $y_{calc,i}$, which are calculated from the model, we now use predicted y values $y_{pred,i}$, which are for data not used to derive the model. Q^2 is given by the following expression (compare with Equation (12.36)):

$$Q^2 = 1 - \frac{\text{PRESS}}{\sum_{i=1}^{N}(y_i - \langle y \rangle)^2}; \quad \text{PRESS} = \sum_{i=1}^{N}(y_{pred,i} - y_i)^2 \tag{12.39}$$

Strictly, the mean observed values, $\langle y \rangle$, which appear in Equation (12.39) should correspond to the mean of the values for each cross-validation group as appropriate rather than the overall mean value of the dependent variables, though often the mean of the entire data set will be used instead.

12.12.4 Interpreting a QSAR Equation

What does one do with a QSAR equation once it has been derived? An obvious use is for predicting the activities of as yet untested, and possibly not yet synthesised, molecules. The predictive ability of a QSAR is generally more accurate for interpolative predictions (i.e. for compounds that have parameter values within the range of those considered in the data set) than for extrapolative predictions (compounds that are outside the range). A QSAR equation may provide insights into the mechanism of the process being studied. As we have already noted, the presence of a parabolic relationship between the activity and the logarithm of the partition coefficient has been interpreted in terms of the transport of a compound to the receptor. An alternative model for transport is the *bilinear model*, in which the activity is related to the partition coefficient by an equation of the following form:

$$\log(1/C) = k_1 \log P - k_2(\log(\beta P + 1)) + k_3 \tag{12.40}$$

The bilinear model enables the ascending and descending parts of the function to have different slopes, whereas the parabolic equation is symmetrical. The parabolic model is generally most applicable to complex *in vivo* systems, where a drug must cross several barriers to reach its target, whereas the bilinear model often gives the best fit to the data for less complex *in vitro* systems

Quantitative structure–activity relationships are often interpreted in terms of specific interactions with the macromolecular target In a number of cases, the crystal structure of the ligand–receptor complex was subsequently determined and so it has been possible to use computer molecular graphics to discover whether the parameters in the QSAR equation have any real meaning [Hansch and Klein 1986] For example, the presence of $\log P$ in the QSAR equation for the inhibition of carbonic anhydrase (Equation (12.32)) was interpreted as a hydrophobic interaction with the enzyme. The crystal structure of the enzyme revealed the presence of just such a hydrophobic surface along which a *para*-substituted group X could lie. The negative coefficients of the indicator variables for *meta* and *ortho* substitution also had a clear interpretation· such substituents would clash with the enzyme.

The absence of a correlation may also provide useful insights. For example, if one set of parameters gives a better correlation than another then this may indicate one particular mechanism is operating. If there is no correlation with a parameter (e.g. a steric measure) for a series of compounds then this may indicate that the associated property (i.e. steric volume) is of less importance.

12.12.5 Alternatives to Multiple Linear Regression: Discriminant Analysis, Neural Networks and Classification Methods

Whilst multiple linear regression is probably the most common technique used in QSAR and QSPR there are some other methods that have proved useful. One significant technique is partial least-squares, which is discussed separately in Section 12 13. Here we will describe a few of the other alternatives

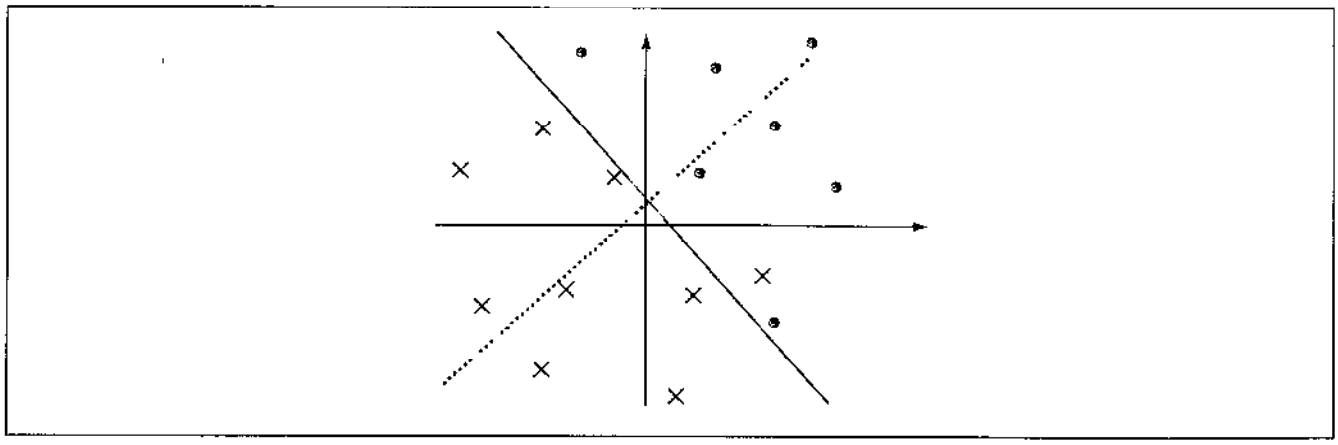


Fig. 12.37: Discriminant analysis defines a discriminant function (dotted line) and a discriminant surface (solid) line.

Multiple linear regression is strictly a 'parametric supervised learning technique'. A parametric technique is one which assumes that the variables conform to some distribution (often the Gaussian distribution); the properties of the distribution are assumed in the underlying statistical method. A non-parametric technique does not rely upon the assumption of any particular distribution. A supervised learning method is one which uses information about the dependent variable to derive the model. An unsupervised learning method does not. Thus cluster analysis, principal components analysis and factor analysis are all examples of unsupervised learning techniques.

Discriminant analysis is a supervised learning technique which uses *classified dependent data.* Here, the dependent data (y values) are not on a continuous scale but are divided into distinct classes There are often just two classes (e.g. active/inactive; soluble/not soluble; yes/no), but more than two is also possible (e.g. high/medium/low, 1/2/3/4). The simplest situation involves two variables and two classes, and the aim is to find a straight line that best separates the data into its classes (Figure 12.37). With more than two variables, the line becomes a hyperplane in the multidimensional variable space. Discriminant analysis is characterised by a *discriminant function,* which in the particular case of linear discriminant analysis (the most popular variant) is written as a linear combination of the independent variables:

$$W = c_1x_1 + c_2x_2 + \cdot + c_Nx_N \tag{12.41}$$

The surface that actually separates the classes is orthogonal to this discriminant function, as shown in Figure 12.37, and is chosen to maximise the number of compounds correctly classified. To use the results of a discriminant analysis, one simply calculates the appropriate value of the discriminant function, from which the class can be determined.

Neural networks have been proposed as an alternative way to generate quantitative structure–activity relationships [Andrea and Kalayeh 1991]. A commonly used type of neural net contains layers of units with connections between all pairs of units in adjacent layers (Figure 12.38). Each unit is in a state represented by a real value between 0 and 1. The state of a unit is determined by the states of the units in the previous layer to which it is connected and the strengths of the weights on these connections. A neural net must first be trained to perform the desired task To do this, the network is presented with a

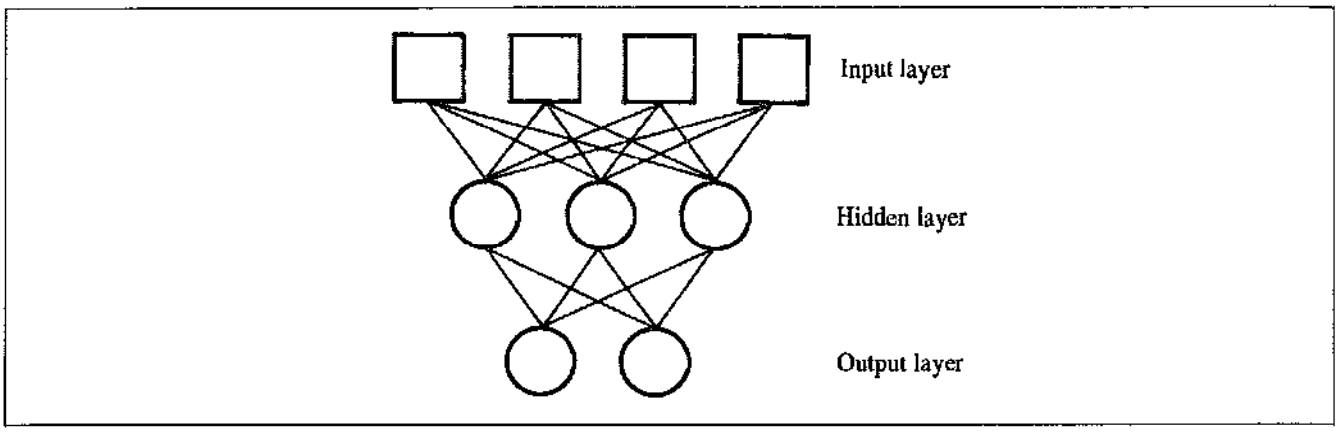


Fig 12 38· Neural network with four input, three hidden and two output nodes

set of sample inputs and outputs. Each input is fed along the connections to the nodes in the next layer, where they are operated upon and the results fed into the next layer, and so on. During the training period, the network adjusts the strengths of the connections using a method called back-propagation [Rumelhart *et al.* 1986] until it finds the set of values giving the best agreement between the input and output. Once trained, the net can then be used in a predictive fashion.

In QSAR, the inputs correspond to the value of the various parameters and the network is trained to reproduce the experimentally determined activities Once trained, the activity of an unknown compound can be predicted by presenting the network with the relevant parameter values. Some encouraging results have been reported using neural networks, which have also been applied to a wide range of problems such as predicting the secondary structure of proteins and interpreting NMR spectra. One of their main advantages is an ability to incorporate non-linearity into the model. However, they do present some problems [Manallack *et al.* 1994]; for example, if there are too few data values then the network may simply 'memorise' the data and have no predictive capability. Moreover, it is difficult to assess the importance of the individual terms, and the networks can require a considerable time to train.

The output from a discriminant analysis or a neural network can often be difficult (if not impossible) to interpret in a manner that easily enables one to identify what features of a molecule give rise to the desired behaviour (or conversely, what features give rise to undesired behaviour!). This contrasts with a loosely associated group of methods that construct 'rules' which can be interpreted in terms of the association between specific molecular features and the activity. Various names are used to denote these methods, including classification trees, decision trees, regression trees, rule induction, machine learning and recursive partitioning. The output from one of these methods can be considered a tree-like structure. At each node there are usually two (but in some methods more than two) branches; which branch is followed for a particular molecule depends upon the rule associated with that node. In QSAR, each rule typically corresponds to the presence or absence of some structural feature of the value of some descriptor An example is given in Figure 12.39 from a study on inotropic compounds (which increase the force of contraction of the heart without increasing its rate) [A-Razzak and Glen 1992]. Each molecule was

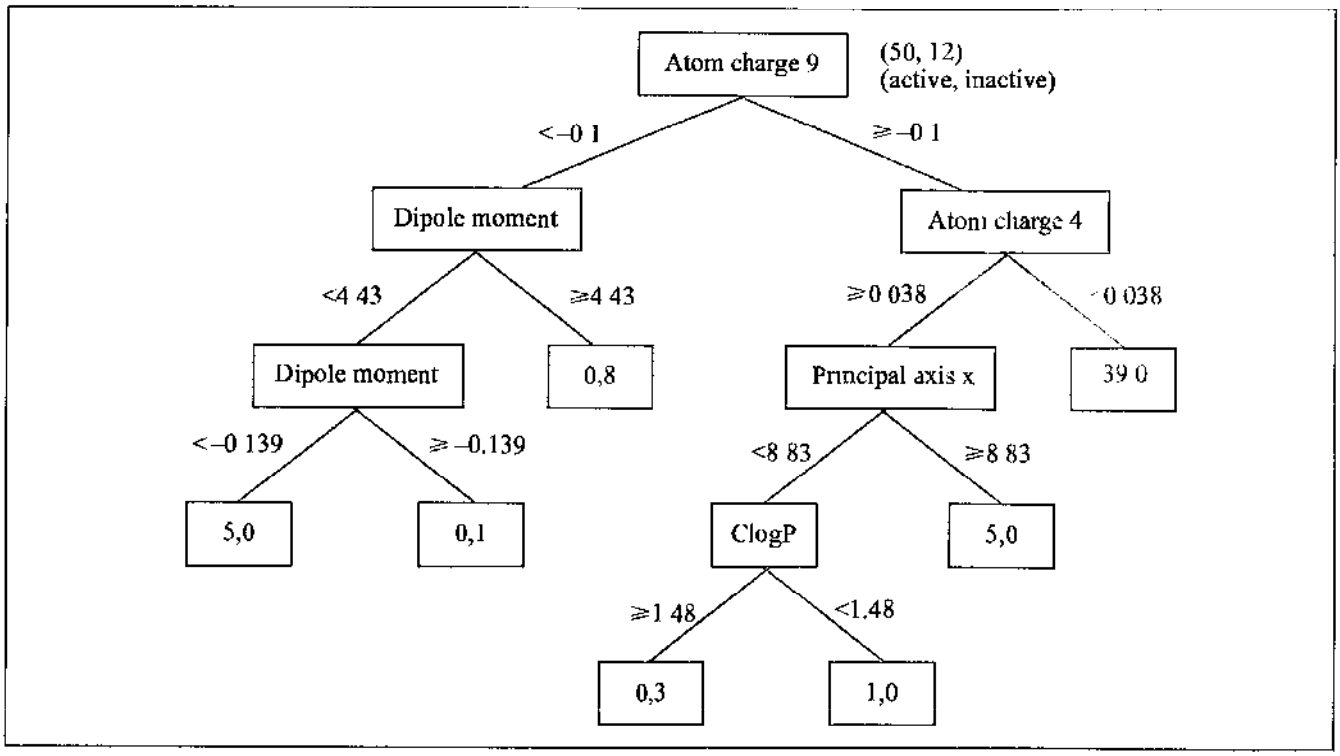


Fig 12 39. Tree describing the 'rules' to differentiate active and inactive inotropic compounds Each of the terminal nodes corresponds to the numbers of active and inactive molecules produced by the application of the preceding rules

classified as active or inactive and was described by 44 descriptors. An algorithm called ID3 was used to decide how to construct the tree from the set of training data (42 compounds, in this case, with 20 compounds for testing). At each stage, the ID3 algorithm chooses to select the property that is 'most informative' from those not yet considered. This is placed on a mathematical footing using a powerful technique called information theory (effectively, it always tries to maximise the entropy gain). Other methods use statistical arguments to decide how to construct the tree. Recursive partitioning uses a statistical test (the *t* statistic) to identify the best descriptor to choose next. In this case, each terminal node in the tree is associated not just with a classification (e.g. active/inactive) but with an actual predicted activity. One particular implementation of recursive partitioning is able to handle large numbers of compounds, each of which is described by an extremely large number of descriptors [Rusinko *et al.* 1999]. This fast implementation is possible because of the binary nature of the descriptors (which are based upon the presence or absence of features such as atom pairs or topological torsions).

A third technique, which uses a somewhat different approach to the problem, is inductive logic programming (ILP) [King *et al.* 1992, 1996]. Initially, a large body of 'facts' about a series of both active and inactive molecules is created. At each stage of the subsequent procedure a machine learning algorithm then takes random pairs of molecules and determines what is common between them to produce a rule which is then evaluated to determine its effectiveness for predicting the remaining molecules. A typical rule can be expressed in a chemically meaningful form, such as 'molecule A is better than molecule B if B has no substitutions at positions 3 and 5 and A has no hydrogen-bond donors at position 3 and A has a π-donor at position 3 and A has a substituent at position 3 with fewer than three rotatable bonds'.

12.12.6 Principal Components Regression

Multiple linear regression cannot deal with data sets where the variables are highly correlated and/or where the number of variables exceeds the number of data values. Two methods are widely used to deal with such situations: principal components regression and partial least squares. In principal components regression, the variables are subjected to a principal components analysis (described in Section 9.13), and then regression analysis is performed using the first few principal components. When a principal components regression is performed using (say) forward-stepping regression then it will be found that the resulting equation is not necessarily expressed using just the lowest principal components. This is because the order of the principal components corresponds to their ability to explain the variance in the independent variables, whereas the regression analysis is concerned with explaining the dependent variable. A general rule of thumb is that only those principal components whose eigenvalues are greater than 1 should be considered for inclusion in a principal components regression. When an eigenvalue falls below 1 then one of the original variables in the set is more effective at explaining the variance than the principal component. Nevertheless, it is often the case that at least the first two principal components often give the best correlation with the dependent variable. Another interesting feature of principal components regression is that, as more principal components are incorporated, the regression coefficients of those already present do not change. This is due to the orthogonal nature of the principal components themselves and because the role of each new principal component is to explain variance not already covered.

12.13 Partial Least Squares

An alternative to principal components regression is to use the technique of *partial least squares* (PLS) [Wold 1982]. The PLS method expresses a dependent variable (y) in terms of linear combinations of the original independent (x) variables as follows:

$$y = b_1 t_1 + b_2 t_2 + b_3 t_3 + \cdots + b_m t_m \tag{12.42}$$

where

$$t_1 = c_{11} x_1 + c_{12} x_2 + \cdots + c_{1p} x_p \tag{12.43}$$

$$t_2 = c_{21} x_1 + c_{22} x_2 + \cdots + c_{2p} x_p \tag{12.44}$$

$$t_m = c_{m1} x_1 + c_{m2} x_2 + \cdots + c_{mp} x_p \tag{12.45}$$

t_1, t_2, etc., are called *latent variables* (or components) and are constructed in such a way that they form an orthogonal set. The use of orthogonal linear combinations of the x values is very similar to principal components analysis. The major difference is that the latent variables in partial least squares are constructed to explain not only the variation in the independent variables x but also to simultaneously explain the variation in the observations, y.

We will illustrate the partial least squares method using a data set published by Dunn *et al.*, which provides the toxicity of a series of halogenated hydrocarbons together with eleven descriptor variables (see Table 12.4).

	Compound	y LD_{25}	x_1 MR	x_2 $\log P$	x_3 BP	x_4 H_{vap}	x_5 MW	x_6 d_{20}	x_7 n_{20}^{Na}	x_8 q_C	x_9 q_{Cl}	x_{10} E ln C	x_{11} E ln Cl
1	CH_2Cl_2	0.96	16 56	1 25	40.0	7.57	85	1.326	1.424	0.097	−0.1083	8 88	9.96
2	$CF_2CHBrCl$	1.31	23.54	2.30	50.0	7.11	197	1.484	1.448	0.1883	−0.1001	9 72	10.04
3	$CHCl_2$	1.45	21.43	1 97	61.7	7.50	119	1.483	1 370	0 1805	−0.0870	9 69	10 16
4	CCl_4	1.53	26.30	2.83	76 5	8 27	154	1 589	1 461	0 2662	−0 0666	10 55	10 36
5	$Cl_2C{=}CHCl$	2.26	26.05	2.29	86.5	8 01	131	1.465	1.456	0.1175	−0.0696	9 90	10.33
6	$Cl_2C{=}CCl_2$	2 26	30.45	2.60	121.0	9.24	166	1.623	1.506	0.1360	−0.0680	10 08	10 34
7	$CHCl_2CHCl_2$	2.42	30 92	2.66	146.0	9.92	168	1.587	1 494	0.1370	−0 1018	9.27	10 02

Table 12 4 Data on halogenated hydrocarbons [Dunn et al. *1984]*

The parameters are as follows: LD_{25}: total toxicity measure; MR: molar refractivity; $\log P$: logarithm of the partition coefficient; BP: boiling point; H_{vap}: latent enthalpy of vaporisation; MW: molecular weight; d_{20}: density at 20°C, n_{20}^{Na}: refractive index at 20°C measured using sodium light; q_C, q_{Cl}: charges on the chlorine-bearing carbon and the chlorine atom, respectively; E ln C, E ln Cl: orbital electronegativities of C, Cl.

The first seven variables ($x_1, \ldots, x_7$) are standard global molecular descriptors, and the final four variables ($x_8, \ldots, x_{11}$) are calculated measures of the electronic character of each molecule. Each of the seven compounds is thus described in terms of eleven variables. However, many of these variables are highly correlated with each other. For example, the charge on the chlorine is perfectly correlated with the electronegativity of the chlorine (a consequence of the way in which these two parameters were calculated, using the Gasteiger and Marsili method). Another strong correlation is that between the molar refractivity and the boiling point (correlation coefficient = 0.92).

A partial least-squares analysis [Malpass 1994] provides the weightings of the original variables in the latent variables. For example, the weightings for the first three latent variables are given in Table 12.5. These results suggest that all of the variables contribute to the first component, with the higher weightings being due to the molar refractivity (x_1), $\log P$ (x_2), boiling point (x_3), latent heat of vaporisation (x_4), d_{20} (x_6) and n_{20}^{Na} (x_7). The first latent variable thus represents a combination of steric, hydrophobic and electronic factors. The highest weightings in the second component are for the charge on the carbon (x_8) and the electronegativity of the carbon (x_{10}) and so this component has a higher contribution from electronic effects. Note that because partial least squares attempts to explain not only the variation in x but also in y, these weightings will differ from those obtained from a principal components analysis on the x variables alone. It is also possible to calculate the

Component	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
1	0.4320	0.3197	0 4428	0.3875	0 1896	0 3265	0.3271	−0.1038	0 2133	0 1219	0 2105
2	−0.0850	0.2172	−0 2863	−0.1765	0.2833	0.2322	0.0479	0.7111	0.0936	0.4147	0.0982
3	0.1273	0.0985	0.0346	−0 3307	−0.1061	−0.1416	−0.5048	−0.2248	0.4841	0.2352	0 4853

Table 12.5. Weightings of the various parameters in the first three latent variables

Component	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	Total
1	97.2	81.5	78 8	67.7	39 8	86.1	66.6	3.6	32 6	30.2	32 1	74.1
2	0.2	14.6	16.4	21 2	12.8	6 8	5.4	84 5	18.4	57 3	19.2	12.8
3	0 3	1.1	0.5	2.2	22.5	0 9	5.7	3.4	45.9	11.4	45.5	4.7

Table 12.6: The degree to which each component explains the variance in each variable, and in the dependent variable (last column)

degree to which each component explains the variance in each variable, and how far each component explains the variation in the dependent variable (Table 12.6).

The results in Table 12.6 reinforce our earlier conclusion that the first component explains most of the steric and hydrophobic effects, with the second component explaining electronic effects. The first component explains 74.1% of the variation in the observed activity, with the first two components explaining a total of 86.9% of the variation.

In this illustration, we have only considered one dependent (y) variable, LD_{25}. In fact, partial least squares can deal with *multivariate* problems, where there is more than one dependent variable, such as different measures of biological activity or different properties. Indeed, for the above set of compounds five different measures of biological activity were reported in the original paper and a partial least-squares analysis performed on the entire data set [Dunn *et al.* 1984]. The algorithm effectively finds pairs of vectors through both the x data and the y data such that the vector pairs are maximally correlated with each other whilst simultaneously explaining as much of the variance in their individual data blocks as possible

12.13.1 Partial Least Squares and Molecular Field Analysis

One of the most popular uses of the partial least-squares method in molecular modelling and drug design is comparative molecular field analysis (CoMFA), first described by Cramer and co-workers [Cramer *et al.* 1988]. The starting point for a CoMFA analysis is a set of conformations, one for each molecule in the set. Each conformation should be the presumed active structure of the molecule. The conformations must be overlaid in the proposed binding mode. The molecular fields surrounding each molecule are then calculated by placing appropriate probe groups at points on a regular lattice that encompasses the molecule, in a manner analogous to that used by the GRID program. The results of this analysis can be represented as a matrix, **S**, in which each row corresponds to one of the molecules and the columns are the energy values at the grid points (Figure 12.40). If there are N points in the grid and P probe groups are used, then there will be $N \times P$ such columns. The table is completed by adding an additional column that contains the activity of the molecule. A correlation between the biological activity and the field values is then determined. The general form of the equation that we desire is:

$$\text{activity} = C + \sum_{i=1}^{N} \sum_{j=1}^{P} c_{ij} S_{ij} \tag{12.46}$$

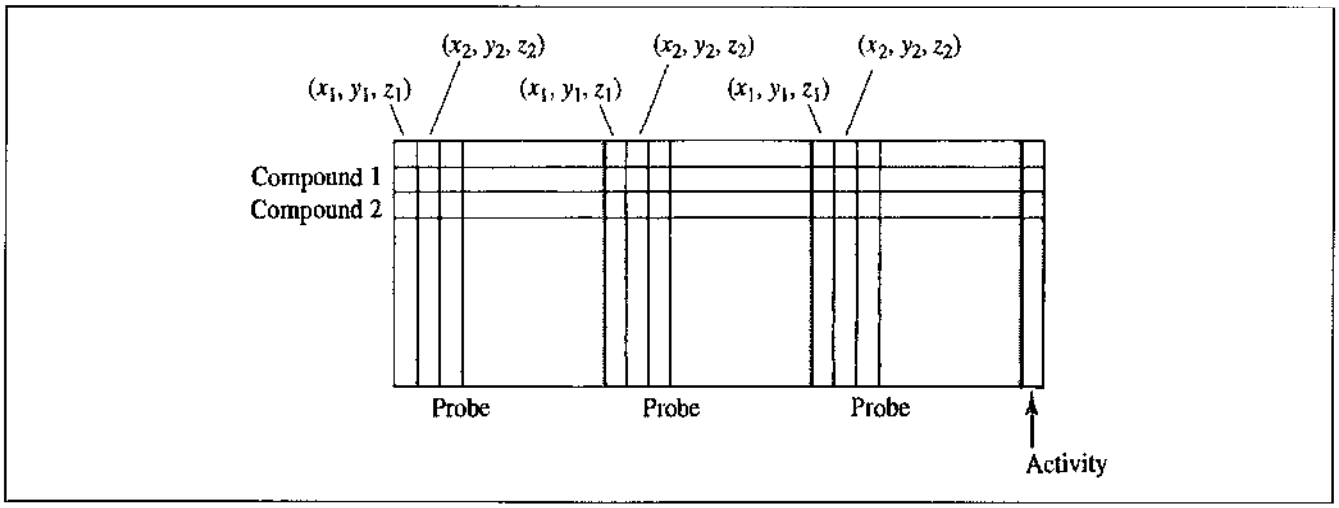


Fig 12 40 The data structure used in a CoMFA analysis

where c_{ij} is the coefficient for the column in the matrix that corresponds to placing probe group j at grid point i. As such, the problem is massively overdetermined as there may be thousands of grid points but often fewer than 30 compounds. Nevertheless, a successful analysis may often be performed using partial least squares.

The maximum number of latent variables is the smaller of the number of x values or the number of molecules. However, there is an optimum number of latent variables in the model beyond which the predictive ability of the model does not increase. A number of methods have been proposed to decide how many latent variables to use. One approach is to use a cross-validation method, which involves adding successive latent variables. Both leave-one-out and the group-based methods can be applied. As the number of latent variables increases, the cross-validated R^2 will first increase and then either reach a plateau or even decrease. Another parameter that can be used to choose the appropriate number of latent variables is the standard deviation of the error of the predictions, s_{PRESS}:

$$s_{\text{PRESS}} = \sqrt{\frac{\text{PRESS}}{N - c - 1}} \tag{12.47}$$

where c is the number of components in the current model. One would generally like to find the smallest number of latent variables which gives a reasonably high Q^2 such that each latent variable gives a fall in the value of s_{PRESS} of at least 5% [Wold *et al.* 1993]. Another measure of the predictive ability is SDEP, which is favoured by some practitioners:

$$\text{SDEP} = \sqrt{\frac{\text{PRESS}}{N}} \tag{12 48}$$

However, SDEP does not penalise an increase in the number of components; should the inclusion of an additional component slightly increase PRESS then the SDEP metric will select the model with more components, whereas s_{PRESS} will not Bootstrapping is another procedure for assessing the stability of a PLS model, which attempts to overcome the all-too-familiar scenario of having fewer data values than might be ideal. In bootstrapping, N random selections are made from the original set several times in order to simulate different samplings from a larger set. Thus in each bootstrapping run some of the data would be

included more than once. This enables one to assess the variation in the different terms in the PLS model and so its stability.

An alternative way to assess the significance of a model is to randomly reassign the activities, thereby associating the 'wrong' activity with each set of grid values. When this is done, the predictive ability of the model should be significantly better for the true data set than for any of the randomised sets. This is a useful technique to check for random correlations when using descriptors that are not easy to interpret.

The CoMFA approach generates a coefficient for each column in the data table. This coefficient indicates the significance of each grid point in explaining the activity. Such data can usefully be represented as a three-dimensional surface that connects points having the same coefficients. These diagrams have been used to identify regions where (for example) changing the steric bulk would increase or decrease binding. An example is shown in Figure 12.41 (colour plate section). These contours can also be very useful for checking that a sensible model has been generated.

Since its introduction, partial least squares has been widely used to calculate such so-called '3D' QSARs. These studies have demonstrated its validity and usefulness but have also highlighted the sensitivity of the approach to several factors [Thibaut *et al.* 1993]. These factors include the selection of the active compounds, the different types of probe group that can be employed, the force-field models to describe the interactions between the probe and each compound, the size and spacing of points in the grid, and indeed the way in which the PLS analysis is performed. One of the main requirements (and indeed limitations) of the CoMFA technique is that it requires the structures of the molecules to be correctly overlaid in what is assumed to be the bioactive conformation (this in turn implies that the compounds have a common binding mode). The first application of CoMFA was to a series of steroid molecules binding to two different targets. human corticosteroid globulins and testosterone-binding globulins. In this case, the steroid nucleus of each molecule was least-squares fitted to the nucleus of the most active steroid. It can be more difficult to determine the appropriate binding mode in other cases, though the pharmacophore identification programs discussed in Section 12.4 may help with this problem. CoMFA can be particularly useful in the design of compounds that are selective for one target over another (related) target; a comparison of the contour maps can highlight regions where the two receptors have different requirements, which can be used to guide subsequent synthesis. It is also worth noting that although it is by far the most well-known approach, CoMFA is by no means the only 3D QSAR technique available [Greco *et al.* 1997].

The vast number of grid-field variables in a typical CoMFA analysis are obviously closely coupled; even the smallest structural change in the compounds will cause changes in not just one variable but in a group of variables that are connected in space. It is thus possible to envisage groups of spatially contiguous grid-field variables which are affected in the same way by structural variations in the compounds. Within such a group, all of the variables contain the same information. The use of such groups in the PLS analysis should give rise to 'better' models (i.e. greater predictive ability and enhanced interpretability). This is the basis of a method termed Smart Region Definition (SRD) [Pastor *et al.* 1997]. The same research group had previously developed automated procedures for selecting

subsets of variables in order to enhance the quality of PLS models [Cruciani *et al.* 1993]. In this latter GOLPE approach (GOLPE stands for Generating Optimal Linear PLS Estimations) multiple combinations of variables are selected using fractional factorial design. For each combination a PLS model is derived and only those variables which significantly influence the predictive ability of the model are retained.

There are three steps in the SRD approach. The first stage involves the selection of a set of grid nodes which have a high importance for the model (i.e. high weights in a 'traditional' CoMFA calculation). Each of these nodes is characterised by a point in 3D space around the molecules and by the particular field (e.g. electrostatic, van der Waals) to which they belong. These nodes acts as seeds, with each of the remaining variables in the data set being assigned to its nearest seed (in a distance sense) or, if the distance to the nearest seed is greater than some cutoff, then the variable is removed from the analysis. It is important to note that the seeds are not uniformly distributed throughout the space; information-rich areas will have many seeds, whilst areas with less information will have fewer seeds. The spatial extent of the regions around the seeds in the information-rich areas will be correspondingly smaller. Those variables which are removed from the analysis usually correspond to areas far away from the compounds or where there is no chemical variation in the compounds. In the third step, the algorithm attempts to merge together neighbouring regions which contain the same information (i.e. are correlated), thus leading to an even greater information reduction. The SRD method was evaluated using a series of glycogen phosphorylase inhibitors, which are particularly relevant for such studies because the structure of each inhibitor bound to the enzyme had been determined by X-ray crystallography As such, the problem of identifying the active conformation of each ligand and producing a molecular overlay disappeared. It was also possible to try to interpret the results of the 3D QSAR analysis in terms of specific interactions with the enzyme. In this case, the energy values were determined using the GRID program with a phenolic hydroxyl probe group (OH). Comparisons with the standard PLS method together with other procedures for grouping variables showed that the SRD algorithm did improve the fit, predictive ability and interpretability of the analysis.

The ability of partial least squares to cope with data sets containing very many x values is considered by its proponents to make it particularly suited to modern-day problems, where it is very easy to compute an extremely large number of descriptors for each compound (as in CoMFA). This contrasts with the traditional situation in QSAR, where it could be time-consuming to measure the required properties or where the analysis was restricted to traditional substituent constants.

12.14 Combinatorial Libraries

Combinatorial chemistry has significantly increased the numbers of molecules that can be synthesised in a modern chemical laboratory. The 'classic' approach to combinatorial synthesis involves the use of a solid support (e.g. polystyrene beads) together with a scheme called 'split-mix'. Solid-phase chemistry is particularly appealing because it permits excess reagent to be used, so ensuring that the reaction proceeds to completion. The excess

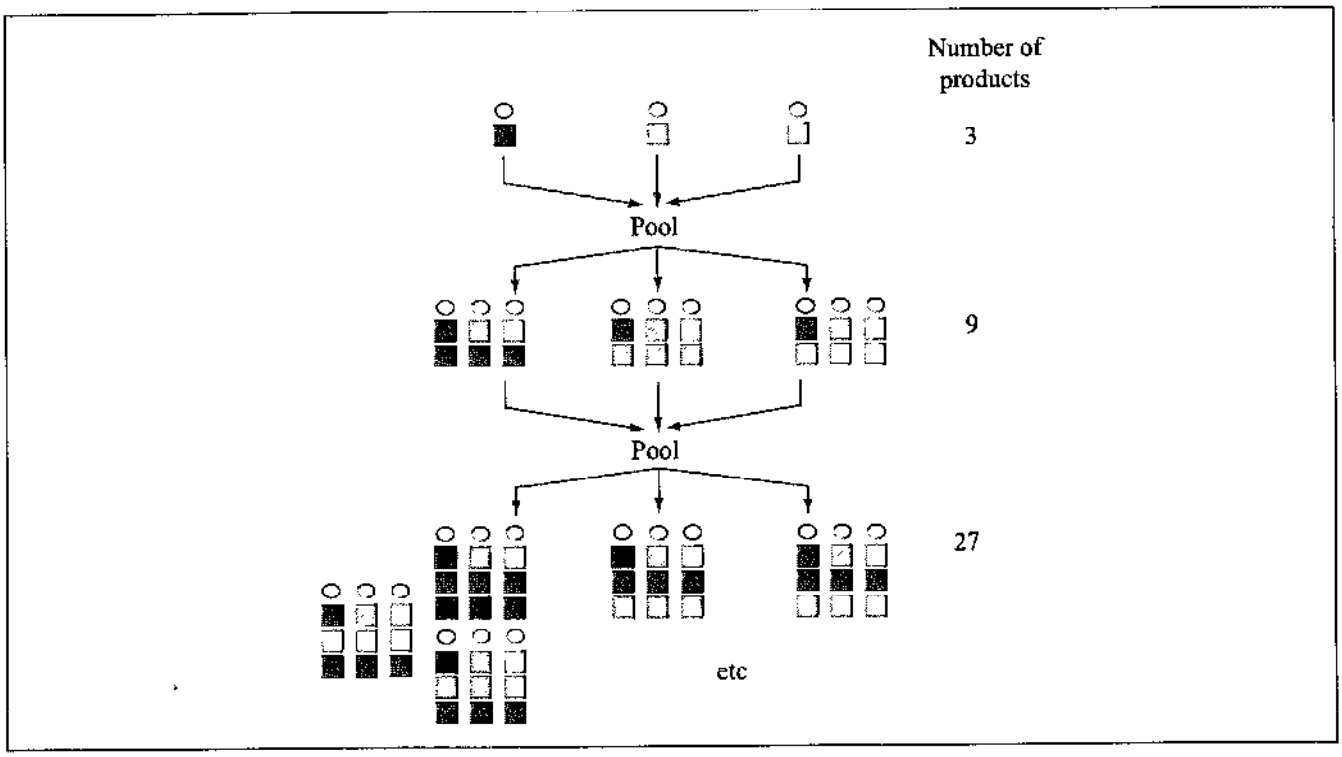


Fig. 12.42. Illustration of the split-mix approach to combinatorial synthesis, using sets containing three monomers

reagent can then be simply washed away. The split-mix approach is illustrated schematically in Figure 12.42; initially, we start with separate pots of the first solid-supported reagent, A (there are n_A of these) These are mixed together and then divided into n_B equal amounts for reaction with the second set of reagents, B The beads are now mixed together again and divided out for reaction with the third set of reagents, C. At this stage there should be $n_A \times n_B \times n_C$ products. The number of products grows exponentially with the number of reagents (hence the term 'combinatorial'). Due to the earlier mixing steps one only knows the identity of the final reagent for a given bead. However, it is important to realise that each bead contains just one discrete compound. The traditional split-mix approach has become less popular because there can still be much work involved in ascertaining the precise identity of any compounds that show activity. The use of 'tags' to encode information about the reagents (henceforth referred to as 'monomers') used to prepare the compound on any particular bead is one alternative.

The initial excitement about combinatorial chemistry was undoubtedly due to the number of molecules that could be synthesised, the assumption being that more molecules would surely lead to more hits in biological assays. However, on the whole this was not observed in practice, with combinatorial libraries often giving rise to fewer leads than historical sets of compounds. In part, this can be ascribed to the fact that as all the molecules in a library will have been synthesised using the same reaction scheme there is an inherent constraint on the amount of structural 'diversity' possible. However, it is also a reflection of the fact that many of the early combinatorial libraries did not (with the benefit of hindsight) contain molecules that were likely to have biological activity let alone be of interest as leads. Two general trends that have emerged recently are the move towards the synthesis of libraries that have been designed for activity against a particular biological target or group of related targets, such

as an enzyme family (focused library design) and towards the synthesis of libraries that contain more 'drug-like' molecules.

As was alluded to above, the early emphasis in combinatorial synthesis was on the generation of large numbers of 'diverse' molecules. Despite the general shift in emphasis to more 'focused' libraries, combinatorial methods do still offer a very powerful way to explore the range of chemical diversity. A useful way to reconcile this apparent conflict is to align the degree of diversity with the amount of knowledge about the molecular target for which the library is being synthesised. Thus, when one has a lot of knowledge about the target (for example, when an X-ray structure is available) then rather less diversity would be required than is the case when (for example) one only knows what general class the target belongs to on the basis of a sequence analysis. The difficulty is in quantifying these factors and the balance between them. An early attempt was described by Martin and colleagues, who used experimental design techniques to select monomers for peptoid libraries (peptoids are synthetic oligomers with a peptide backbone but with the side chain attached to the nitrogen atom rather than the alpha-carbon atom) [Martin *et al.* 1995]. A variety of descriptors were selected to represent features such as lipophilicity, shape and chemical functionality. These properties could then be displayed graphically for each monomer in a very intuitive fashion that greatly facilitated comparison.

12.14.1 The Design of 'Drug-like' Libraries

Solid-phase combinatorial synthesis has its roots in the work of Merrifield on the synthesis of peptides and so it is not surprising that many of the early libraries were made using this type of chemistry. However, peptides do not generally make very good drugs; not only are they readily broken down *in vivo* but they also tend to be rather large, high molecular weight compounds with many rotatable bonds when compared with typical drug molecules. Figure 12.43 shows the distribution of molecular weight, the number of rotatable bonds (a simple measure of conformational flexibility) and the calculated $\log P$ for a number of combinatorial libraries. As can be seen, for some libraries the distributions are quite close to those for the drug molecules, but for others there is a significant difference [Leach and Hann 2000]. What makes a molecule 'drug-like'? This is clearly an almost impossible question to answer, given the vast range of drug structures. Nevertheless, several attempts have been made to try to quantify 'drug-likeness'. The expectation is that libraries that are more drug-like will be more likely to contain molecules with biological activity and that any hits from such a library will represent more attractive starting points for lead optimisation. Of course, the need for drug-likeness is not restricted to combinatorial libraries but indeed can include any molecule that might be put through an assay, and so the techniques can also be applied to the selection of in-house or external compounds for screening.

Most practical implementations of drug-likeness use a computational model which takes as input the molecular structure, together with various properties, and predicts whether the molecule is drug-like or not. Some of these models may be very simple, such as a series of substructural filters. Only those molecules which pass all of these filters are output. Such filters can be used to eliminate molecules that contain inappropriate functionality.

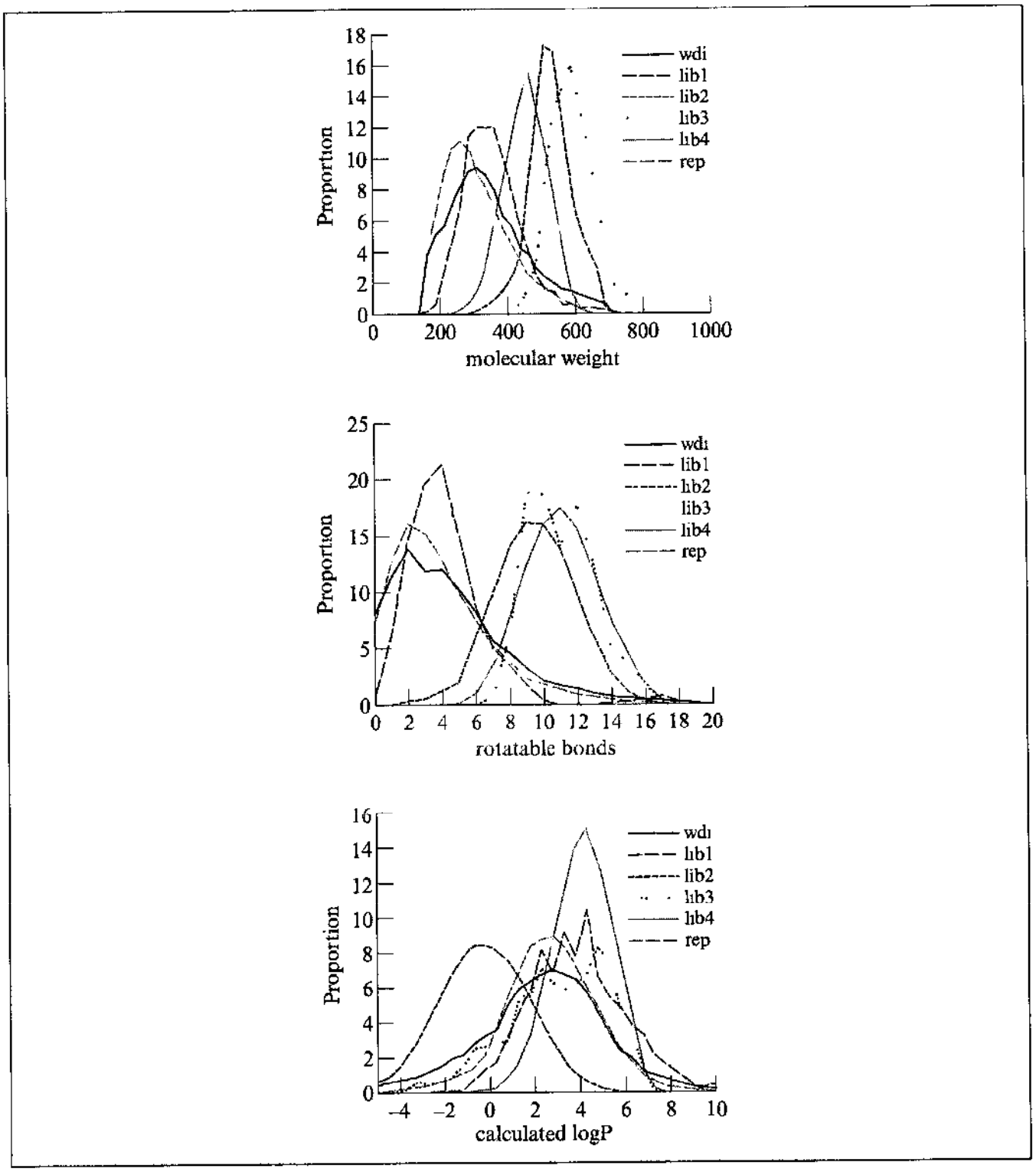


Fig 12 43. Distribution of molecular weight, number of rotatable bonds and calculated partition coefficient for a number of early libraries synthesised at Glaxo Wellcome (lib1–lib4), together with the corresponding distributions for the World Drug Index (wdi) and a set of representative 'historical' compounds (rep)

For example, certain types of reactive group (e.g. alkyl halides, acid chlorides) will almost always give a positive reading in any biological assay. Similar filters can be used to eliminate molecules that have a high molecular weight or are very flexible. The 'rule of 5' is a set of empirical filters that can be used to suggest whether or not a molecule is likely to be

poorly absorbed [Lipinski *et al.* 1997]. Any molecule with a molecular weight greater than 500, a calculated log *P* greater than 5, more than 5 hydrogen-bond donors (defined as the sum of OH and NH) or more than 10 hydrogen-bond acceptors (defined as the sum of nitrogen and oxygen atoms in the molecule) is predicted to be poorly absorbed.

More sophisticated models are also possible. Such models may use neural networks [Sadowski and Kubinyi 1998; Ajay *et al.* 1998] or a regression-type equation with coefficients derived using a genetic algorithm [Gillet *et al.* 1998] to predict drug-likeness from a set of molecular properties. To train such models it is usually necessary to have a set of molecules that are considered drug-like and a set that is considered non-drug-like. The model is then optimised to achieve the optimal discrimination between the drug and non-drug sets.

Filters and drug-likeness models can be extremely valuable in library design and compound selection. They typically require just a 2D representation (such as a SMILES string) and so can be used to rapidly eliminate molecules of no interest and to score or rank the remainder (an example of *virtual screening*). However, it should always be remembered that such approaches are usually very general in nature and that any specific target may require molecules that violate one or more of these more general criteria.

12.14.2 Library Enumeration

The term 'enumeration' when applied to a combinatorial library refers to the process by which the connection tables for the product structures in a real or virtual library are produced. It should be noted that a single compound can be considered as a library of one and so enumeration can equally well be applied in this case. However, whereas it is considered reasonable for a chemist to draw the structure of a single compound manually (which may have taken days, if not months or years, to synthesise), it is clearly not practical to do so even for small combinatorial libraries. Hence the need for automated tools to perform this procedure.

Generally speaking, there are two different approaches to the enumeration problem The first of these is often referred to as 'fragment marking'. In this method, a central core template, common to all product structures, is identified. The template will contain one or more points of variation where different substituents (often termed R groups) can be placed. By varying the R groups at the points of substitution, different product structures can be generated. In order to enumerate a combinatorial library, it is first necessary to construct sets of R group substituents from the relevant monomer sets. In the simplest cases, this is done by replacing the reactive functional group in the monomer with a 'free valence'. By creating a bond between the template and the required R groups the connection table for the product molecule can be generated. Enumeration of the full library corresponds to systematically generating all possible combinations of R-group substituents at the different points of variation.

The alternative approach is to use the computational equivalent of a chemical reaction, or *reaction transform*. Here, one does not need to define a common template or to generate sets of 'clipped' reagents. Rather, the library can be enumerated using as input the initial reagent structures and the chemical transforms required to operate upon them. In this

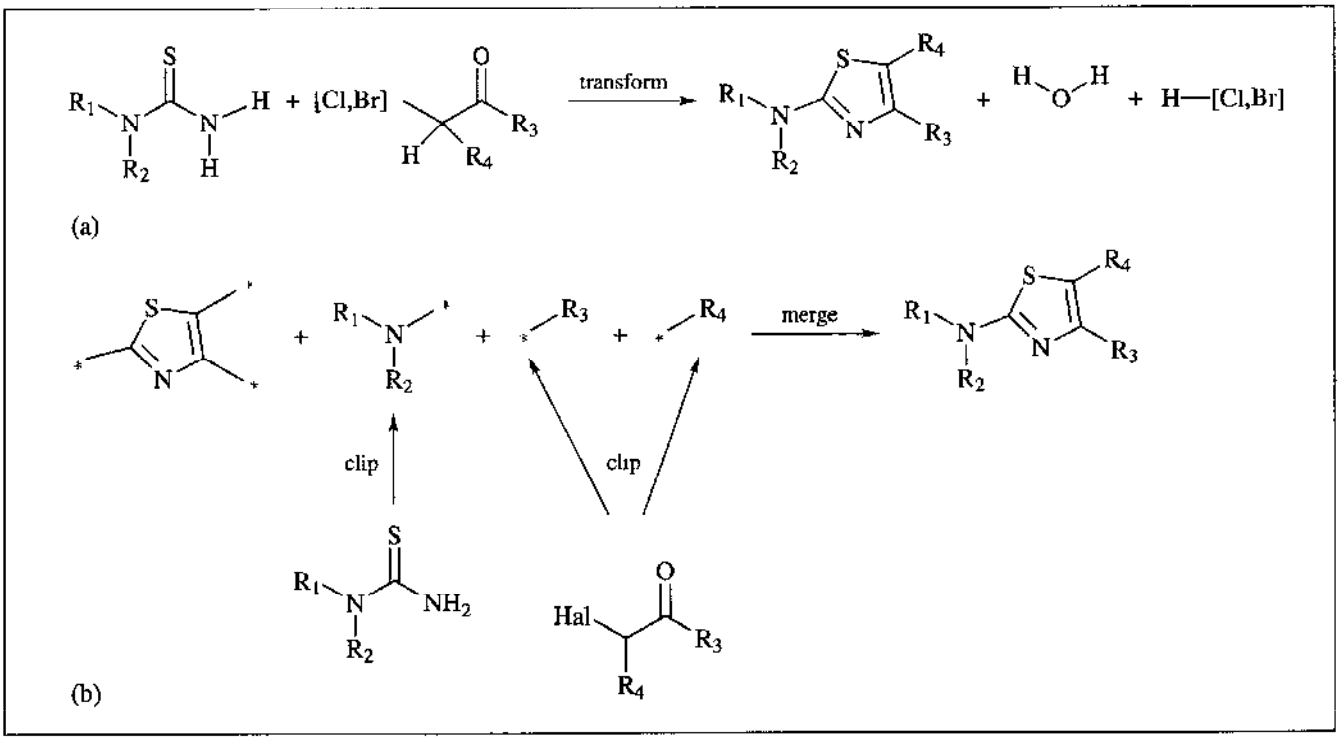


Fig 12 44 Comparison of the reaction transform (a) and fragment-marking (b) approaches to the enumeration of aminothiazoles.

way, it more closely replicates the stages involved in the actual synthesis, wherein reagents react together according to the rules of synthetic chemistry (at least, when the chemistry works as planned!).

The key elements of the fragment-marking and transform approach can be illustrated using as an example the synthesis of aminothiazaoles from thioureas and alpha halo ketones (Figure 12.44). With the reaction transform approach (Figure 12.44(a)), one would simply define an appropriate transform. The enumeration engine then applies this transform to the initial starting materials (i.e. the thiourea and the alpha halo ketone) to produce the aminothiazole (with water and the hydrogen halide as byproducts). Using the fragment-marking approach (Figure 12.44(b)), one would construct three sets of 'clipped' fragments (two from each alpha halo ketone and one from each thiourea), which would then be grafted on to give the central thiazole core to give the appropriate products.

Both the marking-up and reaction transform approach have advantages and disadvantages. In favour of the marking-up approach is the fact that for some kinds of library (i.e. those that most obviously fit the 'core plus R group' definition) it can be the fastest way to enumerate the library. This is because the fragment-marking approach involves only some rather elementary connection table operations once the R groups have been generated. Although most systems offer automated ways to generate the R groups (i.e. 'clipping algorithms') problems almost invariably arise which need to be corrected by hand. This can make the fragment-marking approach time-consuming to perform for a non-expert unless sets of predefined R groups are already available. In addition, there are certain reactions which are not properly handled by the fragment-marking approach, one well-known example being the Diels–Alder reaction, where a simple fragment-marking approach would generate a

number of extraneous and incorrect products. Moreover, in some cases there is no clear core structure. The advantages of the reaction method include the ability to enumerate directly from the reagents without having to perform any pre-processing and the ability to reuse the same transforms many times (once they have been defined). However, this method requires more computational steps and so is typically slower Perhaps the key advantage, however, is that this approach models the actual chemical steps involved in the experiment, so bringing the experimental and computational systems closer together (and thus easier for the bench scientist to appreciate and to do themselves).

Enumeration of a library by either the fragment-marking or the reaction transform approach typically involves just one molecule or reaction scheme being considered at a time. However, the nature of most combinatorial libraries is that there is often much in common between the molecules. A common 'core' can usually be identified (else the fragment-marking method would not work) but in addition some subsets of the products may also have parts in common For example, a fraction of the products may have a phenyl ring at a particular position. By recognising these relationships (using Markush structures, which were originally developed for the computer representation of patents) enumeration can be performed much more efficiently [Downs and Barnard 1997].

12.14.3 Combinatorial Subset Selection

For any synthetic scheme, the key issue in combinatorial library design is monomer selection, the objective of which is to identify those monomers which when combined together provide the 'optimal' combinatorial library. By 'optimal' we mean that library which best meets the prescribed objectives: it might be the most diverse, have the maximum number of molecules that could fit a 3D pharmacophore or a protein binding site, best match a particular distribution of some physicochemical property, or some combination of these or other criteria. An important consideration when designing a combinatorial library is the *subset selection constraint*. In a 'true' combinatorial library of the form $A \times B \times C$, every molecule from the set of reagents A reacts with every molecule from B and every molecule from C to generate $n_A \times n_B \times n_C$ product structures, where n_A, n_B, n_C are the numbers of reagent molecules A, B and C. Typically, there will be many more possible reagents A, B, C available to us than we can actually incorporate into the library, hence the need to select the subset of monomers which give rise to the 'optimal' library. Suppose the number of possible reagents A is N_A, etc. The size of the so-called *virtual library* is thus $N_A \times N_B \times N_C$. The number of ways of selecting n objects from N is ${}^{N}C_{n}$, and so the number of different combinatorial libraries of size $n_A \times n_B \times n_C$ that could be made for this three-component library is ${}^{N_A}C_{n_A} \times {}^{N_B}C_{n_B} \times {}^{N_C}C_{n_C}$. If we have available 100 reagents for each of A, B and C and we wish to make a $10 \times 10 \times 10$ library then the number of possible libraries that we could make is approximately 10^{40}. Identifying the one 'optimal' library from this extremely large number of possible libraries is clearly a difficult problem, which cannot be solved by a systematic examination of every possible solution.

As might be expected, established optimisation techniques such as simulated annealing and genetic algorithms have been used to tackle the subset selection problem These methods

gradually evolve possible solutions until either no better solution can be found or until the predetermined number of iterations is exceeded. In the genetic algorithm approach, the chromosome encodes for a particular set of monomers, which, when combined together in a combinatorial fashion, would give a particular set of products. As we alluded earlier, the initial efforts were directed towards 'diverse' libraries and so the optimisation functions were formulated accordingly. The subsequent emphasis on more focused libraries has required alternative functions that aim to optimise the number of molecules with a particular property. More generic are those approaches which are able to simultaneously optimise for both diversity and target constraints [Gillet *et al.* 1999].

This type of library design is often known as *product-based monomer selection* as it is the properties of the product molecules that determine the ultimate monomer selections. Enumeration is clearly key to this approach, as it requires product structures to be generated. The alternative is monomer-based selection, where one only considers the properties of the individual monomers and not the properties of the product molecules. The main advantage to monomer-based selection is that the size of the search space is much smaller; in product-based selection one has to directly or indirectly consider the $N_A \times N_B \times N_C$ potential product molecules, whereas in monomer-based selection one only need consider the $N_A + N_B + N_C$ monomers. It has been shown that product-based selection gives superior results (as one would expect) [Gillet *et al.* 1997], but it is also clear that in some cases the virtual libraries will be so large that full enumeration may be impossible, and some combination of the two approaches may be required. Moreover, the synthetic strategy may not require the library to be fully combinatorial so enabling this constraint to be relaxed somewhat

12.14.4 The Future

The techniques of combinatorial chemistry and high-throughput screening have developed extremely rapidly since their potential application to drug discovery was recognised [Leach and Hann 2000]. Most large pharmaceutical companies have a significant investment in these areas, and many smaller companies have been founded specifically to exploit these new techniques. In addition, combinatorial techniques are starting to be applied to other areas such as materials science. Nevertheless, there remain some important issues in the way that combinatorial chemistry is practically applied in drug discovery and the role of computational methods in supporting that process. Many of these issues are more concerned with practice than theory, such as the need for adequate supplies of appropriate monomers, the development of new solid-phase chemistries and the alignment of synthesis and screening resources. On the theoretical side, we have seen a gradual shift in emphasis from 'diversity' as the sole factor in library design towards 'biased' or 'focused' libraries that also try to take into account relevant knowledge about the biological target(s) for which the library is intended [Hann and Green 1999]. In addition, some of the notions concerning 'drug-likeness' and the type of molecule that should be in a library are being scrutinised. For example, as the chance of finding the 'drug molecule' in one step is so unlikely it is perhaps more appropriate to expect that molecules with only modest affinity will be found in the early stages. Thus exploratory libraries should contain smaller, less complex molecules than typical drugs [Teague *et al.* 1999]. As with so many other areas of molecular

modelling, progress requires not only good algorithms and faster computers but also a close integration with experiment and a better understanding of the underlying chemical and physical principals involved.

Further Reading

Agrafiotis D K, J C Myslik and F R Salemme 1999. Advances in Diversity Profiling and Combinatorial Series Design *Molecular Diversity* 4 1–22.

Charifson P S (Editor) 1997 *Practical Application of Computer-Aided Drug Design* New York, Dekker

Clark D E, C W Murray and J Li 1997 Current Issues in *De Novo* Molecular Design In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 11. New York, VCH Publishers, pp. 67–125.

Dean P M (Editor) 1995. *Molecular Similarity in Drug Design.* London, Blackie Academic and Professional

Downs G M and Peter Willett 1995. Similarity Searching in Databases of Chemical Structures In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 7 New York, VCH Publishers, pp 1–66

Drewry D H and S S Young 1999 Approaches to the Design of Combinatorial Libraries *Chemometrics in Intelligent Laboratory Systems* 48·1–20

Good A C and J S Mason 1995. Three-Dimensional Structure Database Searches. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 7. New York, VCH Publishers, pp. 67–117.

Graham R C 1993. *Data Analysis for the Chemical Sciences. A Guide to Statistical Techniques* New York, VCH Publishers

Guner O F (Editor) 2000 *Pharmacophore Perception, Development, and Use in Drug Design.* International University Line Biotechnology Series, 2

Jurs P C 1990 Chemometrics and Multivariate Analysis in Analytical Chemistry In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 1. New York, VCH Publishers, pp. 169–212.

Kubinyi H (Editor) 1993. *3D QSAR in Drug Design, Theory, Methods and Applications.* Leiden, ESCOM.

Kubinyi H 1995. The Quantitative Analysis of Structure–Activity Relationships. In Wolff M E (Editor) *Burger's Medicinal Chemistry and Drug Discovery.* 5th Edition, Volume 1. New York, John Wiley & Sons, pp 497–571.

Livingstone, D 1995 *Data Analysis for Chemists* Oxford, Oxford University Press.

Livingstone, D 2000. The Characterisation of Chemical Structures Using Molecular Properties A Survey *Journal of Chemical Information and Computer Science* 40·195–209

Marshall G R 1955 Molecular Modeling in Drug Design. In Wolff M E (Editor) *Burger's Medicinal Chemistry and Drug Discovery.* 5th Edition, Volume 1 New York, John Wiley & Sons, pp 573–659.

Martin E J, D C Spellmeyer, R E Critchlow Jr and J M Blaney 1997 Does Combinatorial Chemistry Obviate Computer-Aided Drug Design? In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 10. New York, VCH Publishers, pp 75–100

Martin Y C 1978. *Quantitative Drug Design A Critical Introduction* New York, Marcel Dekker

Martin Y C, M G Bures and P Willett 1990 Searching Databases of Three-Dimensional Structures. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 1. New York, VCH Publishers, pp 213–263.

Montgomery D C and A A Peck 1992 *Introduction to Linear Regression Analysis* New York, John Wiley & Sons

Murcko M A 1997. Recent Advances in Ligand Design Methods In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 11. New York, VCH Publishers, pp. 1–66.

Oprea T I and C L Waller 1997. Theoretical and Practical Aspects of Three-Dimensional Quantitative Structure–Activity Relationships. In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 11. New York, VCH Publishers, pp 127–182.

Otto M. *Chemometrics. Statistics and Computer Application in Analytical Chemistry.* New York, Wiley-VCH.

Spellmeyer D C and P D J Grootenhuis 1999. Recent Developments in Molecular Diversity: Computational Approaches to Combinatorial Chemistry. *Annual Reports in Medicinal Chemistry* 34·287–296

Tute M S 1990. History and Objectives of Quantitative Drug Design In Hansch C, P G Sammes and J B Taylor (Editors) *Comprehensive Medicinal Chemistry* Volume 4 Oxford, Pergamon Press, pp. 1–31.

Waterbeemd H van de 1995 *Chemometric Methods in Molecular Design* Weinheim, VCH Publishers.

Willett P (Editor) 1997 *Computational Methods for the Analysis of Molecular Diversity Perspectives in Drug Discovery and Design* Volumes 7/8. Dordrecht, Kluwer.

References

Agrafiotis D K 1997 Stochastic Algorithms for Maximising Molecular Diversity. *Journal of Chemical Information and Computer Science* **37**:841–851.

Ajay A and M A Murcko 1995 Computational Methods to Predict Binding Free Energy in Ligand–Receptor Complexes *Journal of Medicinal Chemistry* **38**:4951–4967.

Ajay A, W P Walters and M A Murcko 1998. Can We Learn to Distinguish Between 'Drug-like' and 'Nondrug-like' Molecules? *Journal of Medicinal Chemistry* **41** 3314–3324

Andrea T A and H Kalayeh 1991 Applications of Neural Networks in Quantitative Structure–Activity Relationships of Dihydrofolate Reductase Inhibitors *Journal of Medicinal Chemistry* **34**:2824–2836.

A-Razzak M and R C Glen 1992. Applications of Rule-induction in the Derivation of Quantitative Structure–Activity Relationships *Journal of Computer-Aided Molecular Design* **6**:349–383

Babine R E and S L Bender 1997 Recognition of Protein–Ligand Complexes: Applications to Drug Design. *Chemical Reviews* **97** 1359–1472.

Barnum D, J Greene, A Smellie and P Sprague 1996 Identification of Common Functional Configurations among Molecules *Journal of Chemical Information and Computer Science* **36**:563–571

Baxter C A, C W Murray, D E Clark, D R Westhead and M D Eldridge 1998 Flexible Docking using Tabu Search and an Empirical Estimate of Binding Affinity. *Proteins Structure, Function and Genetics* **33**:367–382

Blaney J M and J S Dixon 1993 A Good Ligand Is Hard to Find: Automated Docking Methods. *Perspectives in Drug Discovery and Design* **1**:301–319.

Böhm H-J 1992. LUDI – Rule-Based Automatic Design of New Substituents for Enzyme Inhibitor Leads. *Journal of Computer-Aided Molecular Design* **6**·593–606

Böhm H-J 1994 The Development of a Simple Empirical Scoring Function to Estimate the Binding Constant for a Protein-ligand Complex of Known Three-Dimensional Structure *Journal of Computer-Aided Molecular Design* **8**·243–256.

Böhm H-J 1998. Prediction of Binding Constants of Protein Ligands: A Fast Method for the Prioritisation of Hits Obtained from *De Novo* Design or 3D Database Search Programs *Journal of Computer-Aided Molecular Design* **12**:309–323

Böhm H-J and G Klebe 1996 What Can We Learn From Molecular Recognition in Protein–Ligand Complexes for the Design of New Drugs? *Angewandte Chemie International Edition in English* **35**·2588–2614

Boström J, P-O Norrby and T Liljefors 1998. Conformational Energy Penalties of Protein-bound Ligands. *Journal of Computer-Aided Molecular Design* **12**:383–396.

Bradshaw J 1997. Introduction to Tversky Similarity Measure. At http //www.daylight.com/meetings/mug97/Bradshaw/MUG97/tv_tversky.html.

Bron C and J Kerbosch 1973 Algorithm 475. Finding All Cliques of an Undirected Graph *Communications of the ACM* **16**:575–577.

Brown R D and Y C Martin 1996 Use of Structure–Activity Data to Compare Structure-Based Clustering Methods and Descriptors for Use in Compound Selection *Journal of Chemical Information and Computer Science* 36:572–583.

Carbo R, L Leyda and M Arnau 1980. An Electron Density Measure of the Similarity Between Two Compounds. *International Journal of Quantum Chemistry* **17**·1185–1189.

Carhart R E, D H Smith and R Venkataraghavan 1985 Atom Pairs as Molecular Features in Structure-Activity Studies· Definition and Applications *Journal of Chemical Information and Computer Science* **25**:64–73.

Charifson P S, J J Corkery, M A Murcko and W P Walters 1999 Consensus Scoring: A Method for Obtaining Improved Hit Rates from Docking Databases of Three-Dimensional Structures into Proteins. *Journal of Medicinal Chemistry* 42 5100–5109.

Cramer R D III, D E Patterson and J D Bunce 1988 Comparative Molecular Field Analysis (CoMFA). 1. Effect of Shape on Binding of Steroids to Carrier Proteins *Journal of the American Chemical Society* **110**:5959–5967

Cruciani G, S Clementi and M Baroni 1993 Variable Selection in PLS Analysis. In Kubinyi H (Editor) *3D QSAR in Drug Design* Leiden, ESCOM, pp 551–564

Cummins D J, C W Andrews, J A Bentley and M Cory 1996. Molecular Diversity in Chemical Databases: Comparison of Medicinal Chemistry Knowledge Bases and Databases of Commercially Available Compounds *Journal of Chemical Information and Computer Science* **36**:750–763.

Dalby A, J G Nourse, W D Hounshell, A K I Gushurst, D L Grier, B A Leland and J Laufer 1992 Description of Several Chemical Structure File Formats Used by Computer Programs Developed at Molecular Design Limited *Journal of Chemical Information and Computer Science* **32**·244–255

Dammkoehler R A, S F Karasek, E F B Shands and G R Marshall 1989. Constrained Search of Conformational Hyperspace *Journal of Computer-Aided Molecular Design* **3**:3–21.

Desjarlais R L, R P Sheridan, G L Seibel, J S Dixon, I D Kuntz and R Venkataraghavan 1988 Using Shape Complementarity as an Initial Screen in Designing Ligands for a Receptor Binding Site of Known Three-Dimensional Structure *Journal of Medicinal Chemistry* **31**:722–729

Downs G M and J M Barnard 1997. Techniques for Generating Descriptive Fingerprints in Combinatorial Libraries *Journal of Chemical Information and Computer Science* **37**:59–61

Downs G M, P Willett amd W Fisanick 1994 Similarity Searching and Clustering of Chemical-Structure Databases Using Molecular Property Data. *Journal of Chemical Information and Computer Science* **34**:1094–1102

Dunn W J III, S Wold, U Edlund, S Hellberg and J Gasteiger 1984 Multivariate Structure–Activity Relationships Between Data from a Battery of Biological Tests and an Ensemble of Structure Descriptors: The PLS Method. *Quantitative Structure–Activity Relationships* **3** 131–137

Eldridge M D, C W Murray, T R Auton, G V Paoliniand R P Mee 1997 Empirical Scoring Functions· I. The Development of a Fast Empirical Scoring Function to Estimate the Binding Affinity of Ligands in Receptor Complexes *Journal of Computer-Aided Molecular Design* **11**.425–445

Gasteiger J, C Rudolph and J Sadowski 1990. Automatic Generation of 3D Atomic Coordinates for Organic Molecules *Tetrahedron Computer Methodology* **3** 537–547

Gelhaar D K, G M Verkhivker, P A Rejto, C J Sherman, D B Fogel, L J Fogel and S T Freer 1995 Molecular Recognition of the Inhibitor AG-1343 by HIV-1 Protease: Conformationally Flexible Docking by Evolutionary Programming. *Chemistry and Biology* **2**.317–324

Ghose A K and G M Crippen 1986. Atomic Physicochemical Parameters for Three-dimensional Structure-directed Quantitative Structure–Activity Relationships. I. Partition Coefficients as a Measure of Hydrophobicity. *Journal of Computational Chemistry* 7·565–577.

Ghose A K, V N Viswanadhan and J J Wendoloski 1998 Prediction of Hydrophobic (Lipophilic) Properties of Small Organic Molecules Using Fragmental Methods: An Analysis of ALOGP and CLOGP Methods *Journal of Physical Chemistry* **102**:3762–3772.

Gillet V J, A P Johnson, P Mata, S Sik and P Williams 1993. SPROUT – A Program for Structure Generation. *Journal of Computer-Aided Molecular Design* **7** 127–153.

Gillet V J, P Willett and J Bradshaw 1997. The Effectiveness of Reactant Pools for Generating Structurally Diverse Combinatorial Libraries *Journal of Chemical Information and Computer Science* **37**·731–740

Gillet V J, P Willett and J Bradshaw 1998. Identification Of Biological Activity Profiles Using Substructural Analysis And Genetic Algorithms. *Journal of Chemical Information and Computer Science* **38**.165–179.

Gillet V J, P Willett, J Bradshaw and D V S Green 1999. Selecting Combinatorial Libraries to Optimize Diversity and Physical Properties. *Journal of Chemical Information and Computer Science* **39**.169–177.

Glen R C and A W R Payne 1995 A Genetic Algorithm for the Automated Generation of Molecules within Constraints. *Journal of Computer-Aided Molecular Design* **9**.181–202.

Good A C, E E Hodgkin and Richards W G 1993 The Utilisation of Gaussian Functions for the Rapid Evaluation of Molecular Similarity *Journal of Chemical Information and Computer Science* **32**.188–192.

Good A C and I D Kuntz 1995 Investigating the Extension of Pairwise Distance Pharmacophore Measures to Triplet-based Descriptors *Journal of Computer-Aided Molecular Design* **9**·373–379.

Goodford P J 1985 A Computational Procedure for Determining Energetically Favorable Binding Sites on Biologically Important Macromolecules *Journal of Medicinal Chemistry* **28**:849–857.

Goodsell D S and A J Olson 1990 Automated Docking of Substrates to Proteins by Simulated Annealing. *Proteins Structure, Function and Genetics* **8** 195–202

Greco G, E Novellino and Y C Martin 1997 Approaches to Three-dimensional Quantitative Structure–Activity Relationships In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 11. New York, VCH Publishers, pp. 183–240

Greene J, S Kahn, H Savoj, P Sprague and S Teig 1994 Chemical Function Queries for 3D Database Search *Journal of Chemical Information and Computer Science* **34**·1297–1308

Hall L H and L B Kier 1991 The Molecular Connectivity Chi Indexes and Kappa Shape Indexes in Structure–Property Modeling In Lipkowitz K B and D B Boyd (Editors) *Reviews in Computational Chemistry* Volume 2 New York, VCH Publishers, pp. 367–422

Hall L H, B Mohney and L B Kier 1991. The Electrotopological State: An Atom Index for QSAR *Quantitative Structure–Activity Relationships* **10**:43–51.

Hann, M and R Green 1999 Chemoinformatics – A New Name for an Old Problem? *Current Opinion in Chemistry and Biology* **3** 379–383.

Hansch C 1969. A Quantitative Approach to Biochemical Structure–Activity Relationships *Accounts of Chemical Research* **2**.232–239.

Hansch C and T E Klein 1986 Molecular Graphics and QSAR in the Study of Enzyme–Ligand Interactions. On the Definition of Bioreceptors. *Accounts of Chemical Research* **19**·392–400

Hansch C, J McClarin, T Klein and R Langridge 1985 A Quantitative Structure–Activity Relationship and Molecular Graphics Study of Carbonic Anhydrase Inhibitors *Molecular Pharmacology* **27**·493–498.

Hassan M, J P Bielawski, J C Hempel and M Waldman 1996 Optimisation and Visualisation of Molecular Diversity of Combinatorial Libraries *Molecular Diversity* **2**.64–74

Head R D, M L Smythe, T I Oprea, C L Waller, S M Green and G R Marshall 1996 VALIDATE: A New Method for the Receptor-Based Prediction of Binding Affinities of Novel Ligands *Journal of the American Chemical Society* **118** 3959–3969

Hodgkin E E and W G Richards 1987. Molecular Similarity Based on Electrostatic Potential and Electric Field. *International Journal of Quantum Chemistry. Quantum Biology Symposia* **14** 105–110

Holiday J D, S R Ranade and P Willett 1995. A Fast Algorithm For Selecting Sets Of Dissimilar Molecules From Large Chemical Databases. *Quantitative Structure–Activity Relationships* **14**:501–506

Holloway M K, J M Wai, T A Halgren, P M D Fitzgerald, J P Vacca, B D Dorsey, R B Levin, W J Thompson, J Chen, S J deSolms, N Gaffin, A K Ghosh, E A Giuliani, S L Graham, J P Guare, R W Hungate, T A Lyle, W M Sanders, T J Tucker, M Wiggins, C M Wiscount, O W Woltersdorf, S D Young, P L Darke and J A Zugay 1995. A Priori Prediction of Activity for HIV-1 Protease Inhibitors Employing Energy Minimisation in the Active Site. *Journal of Medicinal Chemistry* **38**:305–317

Hudson B D, R M Hyde, E Rahr, J Wood and J Osman 1996 Parameter Based Methods for Compound Selection from Chemical Databases. *Quantitative Structure–Activity Relationships* **15** 285–289

Jones G, P Willett and R C Glen 1995a. A Genetic Algorithm for Flexible Molecular Overlay and Pharmacophore Elucidation. *Journal of Computer-Aided Molecular Design* **9**:532–549.

Jones G, P Willett and R C Glen 1995b Molecular Recognition of Receptor Sites Using a Genetic Algorithm with a Description of Desolvation. *Journal of Molecular Biology* **245**:43–53

Jones G, P Willett, R C Glen, A R Leach and R Taylor 1997. Development and Validation of a Genetic Algorithm for Flexible Docking. *Journal of Molecular Biology* **267**.727–748

Judson R S, E P Jaeger and A M Treasurywala 1994. A Genetic Algorithm-Based Method for Docking Flexible Molecules *Journal of Molecular Structure Theochem* **114**:191–206.

Kennard R W and L A Stone 1969 Computer Aided Design of Experiments *Technometrics* **11**:137–148

King R D, S Muggleton, R A Lewis and M J E Sternberg 1992 Drug Design by Machine Learning: The Use of Inductive Logic Programming to Model the Structure–Activity Relationships of Trimethoprim Analogues Binding to Dihydrofolate Reductase *Proceedings of the National Academy of Sciences USA* **89** 11322–11326

King R D, S H Muggleton, A Srinivasan and M J E Sternbery 1996. Structure–Activity Relationships Derived by Machine Learning: The Use of Atoms and Their Bond Connectivities to Predict Mutagenicity by Inductive Logic Programming. *Proceedings of the National Academy of Sciences USA* **93**:438–442

Klopman G, S Wang and D M Balthasar 1992. Estimation of Aqueous Solubility of Organic Molecules by the Group Contribution Approach Application to the Study of Biodegradation. *Journal of Chemical Information and Computer Science* **32**:474–482.

Kramer B, M Rarey and T Lengauer 1999. Evaluation of the FLEXX Incremental Construction Algorithm for Protein–Ligand Docking. *Proteins: Structure, Function and Genetics* **37**:228–241

Kubinyi H 1998. Structure-based Design of Enzyme Inhibitors and Receptor Ligands. *Current Opinion in Drug Discovery and Development* **1**.5–15

Kuntz I D 1992 Structure-Based Strategies for Drug Design and Discovery. *Science* **257**.1078–1082.

Kuntz I D, J M Blaney, S J Oatley, R Langridge and T E Ferrin 1982. A Geometric Approach to Macromolecule–Ligand Interactions. *Journal of Molecular Biology* **161**:269–288

Kuntz I D, E C Meng and B K Shoichet 1994 Structure-Based Molecular Design *Accounts of Chemical Research* **27**:117–123

Lam P Y S, P K Jadhav, C E Eyermann, C N Hodge, Y Ru, L T Bachelor, J L Meek, M J Otto, M M Rayner, Y N Wong, C-H Chang, P C Weber, D A Jackson, T R Sharpe and S Erickson-Viitanen 1994. Rational Design of Potent, Bioavailable, Nonpeptide Cyclic Ureas as HIV Protease Inhibitors. *Science* **263**:380–384.

Lauri G and P A Bartlett 1994 CAVEAT – A Program to Facilitate the Design of Organic Molecules *Journal of Computer-Aided Molecular Design* **8**:51–66.

Leach A R 1994 Ligand Docking to Proteins With Discrete Side-chain Flexibility. *Journal Of Molecular Biology* **235**:345–356.

Leach A R and M M Hann 2000. The In Silico World of Virtual Libraries. *Drug Discovery Today* **5**:326–336.

Leach A R and I D Kuntz 1990. Conformational Analysis of Flexible Ligands in Macromolecular Receptor Sites. *Journal of Computational Chemistry* **13** 730–748.

Lemmen C and T Lengauer 2000. Computational Methods for the Structural Alignment of Molecules. *Journal of Computer-Aided Molecular Design* **14**:215–232.

Leo A and Weininger A 1995 CMR3 Reference Manual. At http://www.daylight com/dayhtml/doc/cmr/cmrref.html.

Leo A J 1993 Calculating log P_{oct} from Structures. *Chemical Reviews* **93**:1281–1306.

Lewis D W, D J Willock, C R A Catlow, J M Thomas and G J Hutchings 1996 *De Novo* Design of Structure-directing Agents for the Synthesis of Microporous Solids. *Nature* **382**:604–606.

Lewis R A, J S Mason and I M McLay 1997 Similarity Measures for Rational Set Selection and Analysis of Combinatorial Libraries: The Diverse Property-Derived (DPD) Approach *Journal of Chemical Information and Computer Science* **37** 599–614.

Lewis R M and A R Leach 1994. Current Methods for Site-Directed Structure Generation *Journal of Computer-Aided Molecular Design* **8**:467–475.

Lipinski C A, F Lombardo, B W Dominy and P J Feeney 1997. Experimental and Computational Approaches to Estimate Solubility and Permeability in Drug Discovery and Development Settings. *Advanced Drug Delivery Reviews* **23**:3–25.

Malpass J A 1994 Continuum Regression Optimised Prediction of Biological Activity PhD thesis, University of Portsmouth, UK

Manallack D T, D D Ellis and D J Livingstone 1994. Analysis of Linear and Nonlinear QSAR Data Using Neural Networks *Journal of Computer-Aided Molecular Design* **37**:3758–3767.

Marriott D P, I G Dougall, P Meghani, Y-J Liu and D R Flower 1999 Lead Generation Using Pharmacophore Mapping and Three-Dimensional Database Searching· Application to Muscarinic M_3 Receptor Antagonists *Journal of Medicinal Chemistry* **42**:3210–3216

Martin E J, J M Blaney, M A Siani, D C Spellmeyer, A K Wong and W H Moos 1995 Measuring Diversity Experimental Design of Combinatorial Libraries for Drug Discovery. *Journal of Medicinal Chemistry* **38** 1431–1436

Martin Y C, M G Bures, A A Danaher, J DeLazzer, I Lico and P A Pavlik 1993. A Fast New Approach to Pharmacophore Mapping and its Application to Dopaminergic and Benzodiazepine Agonists. *Journal of Computer-Aided Molecular Design* **7**·83–102.

Mason J S, I Morize, P R Menard, D L Cheney, C Hulme and R F Labaudiniere 1999. New 4-Point Pharmacophore Method for Molecular Similarity and Diversity Applications· Overview of the Method and Applications, Including a Novel Approach to the Design of Combinatorial Libraries Containing Privileged Substructures. *Journal of Medicinal Chemistry* **42**:3251–3264.

Meng E C, B K Shoichet and I D Kuntz 1992 Automated Docking with Grid-Based Energy Evaluation. *Journal of Computational Chemistry* **13**:505–524.

Miranker A and M Karplus 1991. Functionality Maps of Binding Sites – A Multiple Copy Simultaneous Search Method. *Proteins Structure, Function and Genetics* **11**:29–34.

Moon J B and W J Howe 1991. Computer Design of Bioactive Molecules – A Method for Receptor-Based *De Novo* Ligand Design. *Proteins: Structure, Function and Genetics* **11**:314–328

Morgan H L 1965. The Generation of a Unique Machine Description for Chemical Structures – A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation* **5**:107–113.

Myatt G 1995. Computer-aided Estimation of Synthetic Accessibility. PhD thesis, University of Leeds.

Nilakantan R, N Bauman, J S Dixon and R Venkataraghavan 1987 Topological Torsion: A New Molecular Descriptor for SAR Applications Comparison with Other Descriptors *Journal of Chemical Information and Computer Science* **27** 82–85.

Oshiro C M, I D Kuntz and J S Dixon 1995 Flexible Ligand Docking Using a Genetic Algorithm. *Journal of Computer-Aided Molecular Design* **9**·113–130.

Pastor M, G Cruciani and S Clementi 1997. Smart Region Definition: A New Way to Improve the Predictive Ability and Interpretability of Three-Dimensional Quantitative Structure-Activity Relationships. *Journal of Medicinal Chemistry* **40**:1455–1464.

Patani G A and E J LaVoie 1996. Bioisosterism: A Rational Approach in Drug Design. *Chemical Reviews* **96**:3147–3176

Pearlman R S and K M Smith 1998. Novel Software Tools for Chemical Diversity. *Perspectives in Drug Discovery and Design* vols 9/10/11(3D QSAR in Drug Design: Ligand/Protein Interactions and Molecular Similarity), pp 339–353

Pickett S D, J S Mason and I M McLay 1996 Diversity Profiling and Design Using 3D Pharmacophores: Pharmacophore-Derived Queries (PDQ). *Journal of Chemical Information and Computer Science* **36**:1214–1223.

Poso A, R Juvonen and J Gynther 1995 Comparative Molecular Field Analysis of Compounds with CYP2A5 Binding Affinity. *Quantitative Structure-Activity Relationships* **14** 507–511

Priestle J P, A Fassler, J Rosel, M Tintelnog-Blomley, P Strop and M G Gruetter 1995 Comparative Analysis of The X-Ray Structures of HIV-1 and HIV-2 Proteases in Complex with a Novel Pseudosymmetric Inhibitor. *Structure (London)* **3**:381–389

Rarey M, B Kramer, T Lengauer and G Klebe 1996. A Fast Flexible Docking Method Using an Incremental Construction Algorithm *Journal of Molecular Biology* **261**:470–489

Rhyu K-B, H C Patel and A J Hopfinger 1995. A 3D-QSAR Study of Anticoccidal Triazines Using Molecular Shape Analysis. *Journal of Chemical Information and Computer Science* **35**:771–778.

Rogers D and A J Hopfinger 1994 Application of Genetic Function Approximation to Quantitative Structure-Activity Relationships and Quantitative Structure-Property Relationships. *Journal of Chemical Information and Computer Science* **34**:854–866.

Rumelhart D E, G W Hinton and R J Williams 1986. Learning Representations by Back-propagating Errors. *Nature* **323**:533–536.

Rusinko A III, M W Farmen, C G Lambert, P L Brown and S S Young 1999 Analysis of a Large Structure/Biological Activity Data Set Using Recursive Partitioning *Journal of Chemical Information and Computer Science* **39**:1017–1026

Rusinko A III, J M Skell, R Balducci, C M McGarity and R S Pearlman 1988. *CONCORD: A Program for the Rapid Generation of High Quality 3D Molecular Structures* St Louis, Missouri, The University of Texas at Austin and Tripos Associates.

Sadowski J and H Kubinyi 1998. A Scoring Scheme for Discriminating Between Drugs and Nondrugs. *Journal of Medicinal Chemistry* 41 3325–3329.

Sheridan R P, R Nilakantan, J S Dixon and R Venkataraghavan 1986. The Ensemble Approach to Distance Geometry. Application to the Nicotinic Pharmacophore. *Journal of Medicinal Chemistry* **29**:899–906.

Shuker S B, P J Hadjuk, R P Meadows and R P Fesik 1996. Discovering High-affinity Ligands for Proteins: SAR by NMR. *Science* **274**:1531–1534

Snarey M, N K Terrett, P Willett and D J Wilton 1997. Comparison of Algorithms for Dissimilarity-based Compound Selection. *Journal of Molecular Graphics and Modelling* **15** 372–385.

Swain C G and E C Lupton 1968. Field and Resonance Components of Substituent Effects *Journal of the American Chemical Society* **90**:4328–4337

Swain C G, S H Unger, N R Rosenquist and M S Swain 1983. Substituent Effects on Chemical Reactivity. Improved Evaluation of Field and Resonance Components *Journal of the American Chemical Society* **105**:492–502

Teague S J, A M Davis, P D Leeson and T Oprea 1999 The Design of Leadlike Combinatorial Libraries. *Angewandte Chemie International Edition in English* **38**:3743–3748.

Thibaut U, G Folkers, G Klebe, H Kubinyi, A Merz and D Rognan 1993 Recommendations for CoMFA Studies and 3D QSAR Publications. In Kubinyi H (Editor) *3D QSAR in Drug Design*. Leiden, ESCOM, pp. 711–728

Thornber C W 1979. Isosterism and Molecular Modification in Drug Design. *Chemical Society Reviews* 8:563–580

Tversky A 1977. Features of Similarity *Psychological Reviews* **84**:327–352.

Ullmann J R 1976. An Algorithm for Subgraph Isomorphism. *Journal of the Association for Computing Machinery* 23:31–42.

Von Itzstein M, W Y Wu, G B Kok, M S Pegg, J C Dyason, B Jin, T V Phan, M L Smythe, H F Whites, S W Oliver, P M Colman, J N Varghese, D M Ryan, J M Woods, R C Bethell, V J Hotham, J M Cameron and C R Penn 1993. Rational Design of Potent Sialidase-Based Inhibitors of Influenza Virus Replication *Nature* **363**:418–423

Wang R, Y Fu and L Lai 1997. A New Atom-Additive Method for Calculating Partition Coefficients. *Journal of Chemical Information and Computer Science* **37**:615–621.

Weininger D 1988 SMILES, A Chemical Language and Information System 1 Introduction to Methodology and Encoding Rules. *Journal of Chemical Information and Computer Science* **28**:31–36.

Weininger D, A Weininger and J L Weininger 1989 SMILES 2 Algorithm for Generation of Unique SMILES Notation *Journal of Chemical Information and Computer Science* **29**:97–101.

Welch W, J Ruppert and A N Jain 1996 Hammerhead: Fast, Fully Automated Docking of Flexible Ligands to Protein Binding Sites. *Chemistry and Biology* **3**:449–462.

Wildman S A and G M Crippen 1999 Prediction of Physicochemical Parameters by Atomic Contributions *Journal of Chemical Information and Computer Science* **39**:868–873

Willett P, J M Barnard and G M Downs 1998 Chemical Similarity Searching. *Journal of Chemical Information and Computer Science* **38**:983–996

Willock D J, D W Lewis, C R A Catlow, G J Hutchings and J M Thomas 1997 Designing Templates for the Synthesis of Microporous Solids Using *De Novo* Molecular Design Methods *Journal of Molecular Catalysis A. Chemical* **119** 415–424

Wiswesser W J 1954 *A Line-Formula Chemical Notation.* New York, Crowell Co

Wold H 1982 Soft Modeling. The Basic Design and Some Extensions In Joreskog K-G and H Wold (Editors) *Systems under Indirect Observation* Volume II Amsterdam, North-Holland

Wold S, E Johansson and M Cocchi 1993. PLS - Partial Least-squares Projections to Latent Structures In Kubinyi H (Editor) *3D QSAR in Drug Design* Leiden, ESCOM, pp 523–550

Index

Note: **emboldened** page references indicate chapters

ab initio defined 65
ab initio molecular dynamics 616–22
ab initio potentials for water 216–18
ab initio quantum mechanics, calculating properties using 74–86
  see also advanced *ab initio* methods
absolute free energies 573–4
accessible surface 7
ACE (angiotension converting enzyme) 649–51
acetaldehyde 180, 578
acetamide 573
acetic acid 504–5, 573, 643
  SMILES notation 644, 645
4-acetamido benzoic acid 661
acetonitrile 597
acetylcholine 678
acronyms and abbreviations 104–5, 553–4
adenine 227
adiabatic mapping 286
adjacency matrix 647
adjoint matrix 15
adsorption processes, Monte Carlo simulations of 441–2
advanced *ab initio* methods **108–64**
  density functional theory 126–37
  electron correlation 110–17
  energy component analysis 122–4
  open-shell systems 108–10
  practical considerations 117–22
  solid state quantum mechanics 138–60
  valence bond theories 124–6
agglomerative cluster analysis methods 493–4
agonists 640
AINT function 350
alanines 169, 459, 511, 525, 542, 546, 556–7
  energy minimisation methods 277, 280, 286
  free energy calculations 583–4
aldehyde 610–11
aldol reactions 610–12
aliovalent substitution 623
alkaline earth oxides 147
alkanes 449–50
α-helix 513–15
AM1 86, 97–8, 102–3, 230
AMBER force field 169–70, 175–6, 191, 211, 230–2
amino acids 511, 549, 602
  computer simulation 329–30
  conformational analysis 459, 487
  energy minimisation methods 277, 280, 286
  force fields 169–70, 221
  free energy calculations 572, 583–4
  motifs 522
  PAM matrices 524–6, 531, 556–7
  'threading' 546
  torsion angles 515
  see also peptides; proteins
aminothiazoles 716
AMPAC program 8, 99
amphiphiles, molecular dynamics simulation of 394–404
angiotension converting enzyme 649–51
angle bending 166, 173
annealing, simulated 483–9, 504–5, 519, 691
annotations 513
antagonists 640
antisymmetry principle 35
arbitrary step energy minimisation 264
Argand diagram 17
arginine 329–30, 510, 525, 556–7
argon 253, 323
  force fields 205, 214
  J-walking 434–5
  time-steps 361–2
  velocity autocorrelation 377
arithmetic mean 20
aromatic systems and charge schemes 197–9
asparagine 330, 510, 525, 546, 556–7
aspartic acid 510, 525, 556–7
atoms/atomic
  charges 157–9, 181, 192–5
  marker 329–30
  one-electron 30–4
  orbitals 41–2, 56, 100, 241
  polyelectronic 34–41
  type 169
  units 29
atoms in molecules theory 80–1
Aufbau principle 35

Austin Model 1 (AM1) 86, 97–8, 102–3
autocorrelation function 376–8
automated protein modelling 548–9
autoscaling 681
availability 300
Axilrod–Teller term (triple-dipole) 213–15, 239
Azimuthal quantum number 31

backtracking 462
backward sampling 567
band theory 141–2
  orbital-based approach 142–6
Barker–Fisher–Watts potential 214
Basic Local Alignment Search Tool *see* BLAST
basis sets/functions 56, 85, 123
  computational quantum mechanics 65–74
  Gaussian functions 65–73 *passim*, 120, 137, 195
  superposition error 121–2
  see also STOs
BCUT method 686–7
bead model of polymers 428
Becke *see* BLYP
Beeman's algorithm 357
bending 166, 173, 176–8
benperidol 675
benzamidine 586–8
benzene 123
  force fields 170, 174, 178, 186, 197–8
  Hückel theory 99, 100
  ring 81, 170, 178
  SMILES notation 644
  spin-coupled valence bond theory 126
benzyl bromide 670
beryllium 40, 113
β-strand structures 513–14
β-turns 513
BFGS (Broyden–Fletcher–Goldfarb–Shanno) method 269–70
bilinear model 702
binding site 662, 689–91
binomial expansion 11
bioinformatics 513
  see also amino acids; DNA; proteins
bioisosteres 648
biotin 576, 641
bitstring 645–7
BLAST (Basic Local Alignment Search Tool) 521, 524, 531–4, 548
Bloch's theorem/function 142–5, 146, 148, 161
block-diagonal Newton–Raphson minimisation 268
BLOSUM matrices 526
BLYP (Becke gradient-exchange correction and Lee–Yang–Parr correlation functional) 135, 136, 137
B3LYP density function 615
Bohr radius 31
Boltzmann distribution 192, 214, 274, 483, 611
  computer simulation 306–7, 347
  conformational analysis 457
  Monte Carlo simulation 415–16, 433, 435–6, 445–6
Boltzmann factor 306–7, 413
Boltzmann weighted average 576, 581
bond/bonding 644
  inorganic molecules 234–5
  lack of *see* non-bonded interactions
  orders 81–3
  stretching 166, 170–3
  valence 124–6
  see also under carbon; hydrogen
bond fluctuation model 424–5
Born equation/model 238, 593–4, 598–601, 609
Born–Oppenheimer approximation 4, 35–6, 50
boundary
  computer simulation 317–21
  element method 598
Bravais lattices 138–9
Brillouin's theorem 112–13, 115
Brillouin zone 140–1, 145–6, 150–1, 157–8, 298–9
bromine 571
Broyden–Fletcher–Goldfarb–Shanno method 269–70
BSSE (basis set superposition error) 121–2
Buckingham potential 209, 238
build-up approach 517
butadiene 233, 294–5
butane 85–6, 186, 449–50, 582
butanone 611–12

cage structure of liquids 377
calcium 589, 626
calix[4]arene 291–2
Cambridge Structural Database *see* CSD
canonical ensemble 563, 569
canonical genetic algorithms 479–80
canonical representation 644
canonical structures 541
captopril 649
Carbo index 678–9
carbon
  bonds 4–5, 98, 362, 612, 652
    energy minimisation methods 253, 280–1
    force fields 167, 180, 211, 233, 236
  five-carbon fragment 472
  force fields 167, 180, 211, 233, 236, 244
  valence electron density 160
carbon dioxide 181, 316, 616–17
carbonic anhydrase 616
carboxylic acid 660
Car–Parrinello scheme 610, 617–19

Cartesian coordinates 2–4
  conformational analysis 466–8
  energy minimisation methods 255, 257–8, 275, 290
  molecular dynamics simulation 370, 372, 379, 393
  Monte Carlo simulation 417, 420–1, 423
  vectors 11–12
CASP3 548
CASSCF (complete active-space SCF) 113, 295
CAVEAT program 689
CBMC (configurational bias Monte Carlo) simulation 443–50
cell
  cubic 315–19
  index method 326
  multipole method 341–3
  unit 138
  Wigner–Seitz 140, 350
Central Dogma 509, 512
central multipole expansion 181–7
CFF (consistent force field) 231
chain amphiphiles and molecular dynamics 394–404
charge 187
  atomic 157–9, 181, 191–5
  density matrix 58–9
  image simulation 340–1
  oscillating 201–2
  schemes 197–9
CHELP procedure 191
chemical potential calculation in Monte Carlo simulation 442–3
chemical reactions 610–22
  molecular dynamics 616–22
  potential of mean force of 612–14
  quantum and molecular mechanics combined 614–16
  simulation empirically 610–12
chemokine 475
chi molecular connectivity indices 672
chi-squared test 344–5
chiral constraints 473–4
chlorides 238, 612–13, 614, 676–7
chlorine 181, 231, 280–1, 571, 612, 620–1
chloroform 227, 573
chlorpromazine 678
chymosin 545
chymotrypsin 522–3
CI (configuration interaction) 111–13, 120
CID (configuration interaction doubles) 112, 113
CISD (configuration interaction singles and doubles) 112, 113
city block *see* Hamming
Clausius, virial theorem of 309
Clausius–Mosotti relationship 238–9
clique detection 653–6
CLOGP program 669–70
closed-shell systems 51, 56–9, 86–8, 109
cluster analysis 494, 534, 682–3
clustering algorithms 491–7
CMR program 671
CNDO (complete neglect of differential overlap) 86, 89–92, 93, 94, 95
coefficients 148–9, 374
  new molecules 676–8, 680–1, 685
  partition 572–3, 668–71
cofactor 14–15
combination generator 420, 453–4
combinatorial explosion 460–1
combinatorial libraries 711–19
CoMFA (comparative molecular field analysis) 679, 708–11
comparative modelling of proteins 539–45
comparative molecular field analysis 679, 708–11
complete active-space SCF 113, 295
complete neglect of differential overlap 86, 89–92, 93, 94, 95
complete-linkage (furthest-neighbour) cluster algorithm 493–4
complex numbers 16–18
computational quantum mechanics **26–107**
  acronyms used in 104–5
  approximate orbital theories 86
  atomic units 29
  basis sets 65–74
  calculating properties 74–86
  Hückel theory 99–102
  one-electron atoms 30–4
  operators 28–9
  polyelectronic atoms and molecules 34–41
  semi-empirical methods 65, 86–99, 102–3
  see also calculations *under* orbital; Hartree–Fock equations
computer simulation **303–52**
  boundaries 317–21
  equilibration, monitoring 321–3
  free energy calculation difficulties 563–4
  long-range forces 334–43
  molecular dynamics 305–6, 307
  phase space 312–15
  practical aspects 315–16
  real gas contribution to virial 309, 349–50
  results and errors 343–7
  statistical mechanics 347–8
  thermodynamic properties, simple 307–12
  time and ensemble averages 303–5
  translating particle back into control box 350
  truncating potential and minimum image convention 324–34
  see also molecular dynamics simulation, Monte Carlo simulation

computers
hardware 8–9
Internet and World Wide Web 9–10, 548, 553
software 8–9, 99
see also databases
concepts **1–25**
see also coordinates; mathematical concepts
CONCORD program 659
conditionally convergent series 336
conduction band 142
conductor-like screening model 597
configuration interaction *see* CI; CID; CISD
configurational bias *see* CBMC
conformational analysis **457–508**
choice of method 476–7
clustering algorithms and pattern recognition 491–7
conformational search 457, 662
random 465–7, 476
systematic 458–64, 476, 505
crystal structures predicted 501–5
dimensionality of data set reduced 497–9
distance geometry 467–75, 476
fitting, molecular 490–1
global energy minimum 458, 479–83
model-building 464–5 and NMR/x-ray crystallography 468, 474–5, 483–9
poling 499–501
structural databases 482, 489–90, 493–4, 499
variations on standard methods 477–9
conformational changes in molecular dynamics simulation 392–3
conformationally flexible docking 662–3
CONGEN program 542
conjugate gradients 262, 264–7, 473
conjugate peak refinement 290–1
connection table 643
consistent force field (CFF) 231
constraints
chiral 473–4
constraints, holonomic versus non-holomonic 370
in molecular dynamics 368–74
simulation 368–74
and restraints, difference between 369–70
subset selection 717
systematic search 649–51
contact surface 7
'continuous' models of polymers 428–31
continuum models and solvation, free energy of 592–3, 598–601
contraction, basis set 69
convergence sphere 186
coordinates 2–4
internal 2–4, 257
intrinsic reaction 288–9
mass-weighted 274–5
scaled 438–9
see also Cartesian coordinates
Corey–Pauling–Koltun (CPK) models 5–6
CORINA program 659
correlation
BLYP 135, 136, 137
coefficients 374, 681
electron 110–17
exchange-correlation functional 129–34
functions and molecular dynamics simulation 374–80
spectroscopy (COSY) 474–5, 486
Cosine coefficient 676, 685
COSMO (conductor-like screening model) 597
COSY (correlated spectroscopy) 474–5, 486
Coulomb interaction/integral 598–9
advanced *ab initio* methods 122, 127, 128, 132, 133–4, 146–7
computational quantum mechanics 30, 42, 45, 49–53, 58, 60, 85, 100
force fields 184–5, 187, 194, 202, 238
Coulomb potential 167, 244, 338, 341–2
Coulomb's law 95, 194, 202, 212, 596, 603–4, 607
counterpoise correction 121–2
coupling parameter 567
Craig plot 681
cross-correlation function 376
crossover operator 480–1
crystal momentum 148
crystal structures, predicted 501–5
CSD (Cambridge Structural Database)
conformational analysis 482, 489, 493–4, 499
new molecules 659, 691, 693
Cu–Zn superoxide dismutase 607
cut-offs in computer simulation 324–7
group-based 327–30
problems with 330–4
cyclic urea, HIV protease inhibitor 691–3
cyclobutane 176–7
cyclobutanone 176
cyclobutene 117
cycloheptadecane 476
cyclohexane 286, 463, 465, 497, 597, 644
cyclopropene 117
cyclosporin 196–7, 391
cysteine 511, 525, 556–7
cytosine 84, 227

databases 489, 537, 539
3D 659–61, 679
see also CSD; structural databases
Davidon–Fletcher–Powell method 269–70
de Broglie thermal wavelength 411, 440–1
de novo ligand design 687–94

degrees of freedom 699–700
delocalised π-systems, force fields for 233–4
DelPhi program 604–5
density
  charge density matrix 58–9
  electron 77–9, 80–2, 160
  functional theory 126–37, 156, 619
    of levels 154
  spin 129–31
    of states and Fermi surface 153–5
depth-first search 462, 663
derivative, energy
  calculating 120–1
  function 225–6
  minimisation 257–8, 261–2, 268–9
descriptors and new molecules 668–79
determinant of matrix 13–14
DFP (Davidon–Fletcher–Powell) method 269–70
DFT (density functional theory) 126–37, 156, 619
DHFR (dihydrofolate reductase) 278–9, 320, 460
diagonalisation of matrix 16
diatomic overlap *see* MNDO
1,2-dichloroethane 387–8
Dick–Overhauser shell model 239
dielectric constant 297
dielectric models 202–4
Diels–Alder reaction 294–5, 615, 716–17
differences, free energy 564–74
  applications of methods 569–74
  formula for 568, 630–1
  methods for calculating 564–9
differential overlap
  neglect of 86, 89–96
  zero (ZDU) 88–9, 91
dihedral angle, definition 4
dihydrofolate reductase 278–9, 320, 460
DIIS (direct inversion of iterative subspace) 118
dimensionality reduction 497–9
dimethyl formamide 613–14
dimethyl thioether 676–7
N,N-dimethyl-ketopropanamide 230
dipole 75–7
  force fields 181, 182–5, 189, 199–201, 219, 246
  models of solvation, free energy of 593–5, 601–3
  moment, net 378–9
  triple 213–15, 239
dipole correlation time 378–8
direct inversion of iterative subspace 118
direct SCF method 118–20
director 396
discriminant analysis and QSAR 703–5
dispersion curve 298–9
dispersive interactions 204–6
displacements 322–3, 624
dissimilarity-based methods 683–5
dissipative particle dynamics 402–4
distance
  bounds 468
  geometry 467–75, 476, 651–3, 663
  Hamming (city block) 492, 676–8
  map 651
  matrix 652
  Soergel 676–8
distance-dependent dielectric 203
distributed multipole analysis 195–7
diverse sets of compounds, selecting 680–7
DMA (distributed multipole analysis) 195–7
DMF (dimethyl formamide) 613–14
DNA 197, 227, 452, 489, 604
  computer simulation 319, 338–9
  Human Genome Project 512, 548–9
  inhibitor 270–1
  new molecules 662
  proteins 509, 512, 549
DOCK program/algorithm 662–3, 665, 667
docking 661–8, 689
domain 515
D-optimal design 697–8
double dynamic programming 537–9
double zeta basis sets 70
double-wide sampling 567–8
DPD (dissipative article dynamics) 402–4
Dreiding models 5–6
Drude molecules 205–6
  interaction between 246–7
drugs, new *see* new molecules
dual topology 578
dummy atoms, in Z-matrix 271–2
Dunning basis sets 73
dynamic/dynamics 353–409
dynamically modified windows 578
  programming and protein prediction 526–9
  and statics in energy minimisation 295–300
  see also molecular dynamics

EA (evolutionary algorithms) 479–83
edges of search trees 461
effective medium theory 243–4
effective pair potentials 214–15
eigenvalues and eigenvectors 15–16, 114
  conformational analysis 469, 471, 479, 498
  energy minimisation methods 272, 282–5
Einstein relationships 381, 627–8
Eisenberg's 3D profiles 543–4
elastic constants 240, 296–7
electric multipoles, calculation of 75–7
electron
  affinity (EA) 194
  correlation 110–17
  density 77–9, 80–2, 160

electron (*cont*)
gas theory 240
integrals, one- and two- 50–1
nearly free-electron approximation 147–53
polyelectronic atoms and molecules 34–41
spin 34–5, 38
electronegativity 192–3
electrostatic interactions
force fields 166, 181–204, 221, 237
free energy calculations 566–7, 576, 580, 588, 613
potentials 83–5, 188, 189–91
solvation free energy calculations 593–608
electrotopological state index 674
embedded-atom model 241, 243–4
embedding 469
empirical bond-order potential *see* Tersoff
endothiapepsin 589–91
energy
calculation from wavefunction 41–6
of closed-shell system 51
component analysis 122–4
computer simulation 308, 348–9
conservation in molecular dynamics simulation 359, 405–6
derivatives, calculating 120–1
force field 240
function, derivatives of 225–6
of general polyelectronic system 46–50
global minimum 253, 458, 479–83, 551–2
Koopman's theorem and ionisation potentials 74–5
lower-energy regions 564
minimum, global 253, 458, 479–83, 551–2
potential 4–5, 238, 253
strain 226–7, 627
surface (hypersurface) 4–5, 253, 475
units of 9
see also derivative; energy minimisation; free energy; quantum mechanics
energy minimisation methods **253–302**, 623
applications of 273–9
choice of 270–3
derivative 257–8, 261–2, 268–9
first-order 262–7
Newton–Raphson 267–8, 270, 288
non-derivative 258–61
quasi-Newton 268–9
solid-state systems 295–300
statement of problem 255–7
transition structures and reaction pathways 279–95
enol boroate/aldehyde reaction 610–11
ensemble
averages 303–5
distance geometry 651–3
molecular dynamics 653
Monte Carlo simulation 438–42, 450–1
enthalpy 159, 574
entropy 574
enumeration of libraries 715–17
equilibration monitoring and computer simulation 321–3
equilibria phases in computer simulation 315, 450–1
ergodicity 304, 313
quasi ergodicity 433–8
error, estimating in a simulation 343–7
ESS (explained sum of squares) 699–700
ethane
carbon–carbon bond 4–5
force fields 174
Monte Carlo simulation 441–2
SMILES notation 644
thiol 564–9
torsion angles 286–7
Z-matrix 2–3, 9
ethanol 564–9, 611–12
ethene 83, 236, 293–5
ethylene 621
ethyne 83
Euclidean distance measure 492
Euler angles 421–2
even-tempered basis set 71–2
evolutionary algorithms 479–83
evolutionary design 694
evolutionary planning (EP) and strategies (ES) 479–80, 482
Ewald summation 238, 334–9, 342, 402, 625–6
exchange
-correlation functional 129–34
forces *see* repulsive forces
gradient 135, 136, 137
integral 50, 52–3, 58, 60
interaction 46
exclusion spheres 658–9
exons 512
explained sum of squares 699–700
extended Hückel theory (EHT) 101–2
extended system method 384
extreme value distribution 532

fabric softeners 401–2
face-centred cubic lattice 139, 316
factor analysis 681–2, 686
factors (variables) 697
family (proteins) 539
fast Fourier transform 24, 338–9, 342
fast multipole method 341–3, 364
FASTA 524, 531
FDPB (finite difference Poisson–Boltzmann method) 604–8

Fermi surface and energy 153–5
ferrocene 234
FFT (fast Fourier transform) 24, 338–9, 342
Fick's laws 380–1
finite difference methods 355–8, 604–8
Finnis–Sinclair potential 241–5 *passim*
first principles method for predicting proteins 517–22
first-order energy minimisation 262–7
fitting, molecular 490–1
flexible fitting 491
flexible molecules 423, 582–5
FlexX program 667
Fock matrix 100
  Hartree–Fock equations 57–9, 61, 63–4
  open-shell systems 108–9
  semi-empirical methods 89–90, 94, 95, 96
  solid state quantum mechanics 146
Fock operator 53, 57, 114
focusing 606
folding *see under* proteins
force field models, empirical **165–252**, 610
  angle bending 166, 173
  bond stretching 166, 170–3
  Class 1, 2 and 3 178–80
  cross terms 178–80
  derivatives of molecular mechanics energy function 225–6
  Drude molecules, interaction between 246–7
  effective pair potentials 214–15
  general features 168–70
  hydrogen bonding 215–16
  improper torsions and out-of-plane bending 176–8
  inorganic molecules 234–6
  many-body effects in empirical potentials 212–14
  metals and semiconductors 240–5
  parameters 221, 224–5, 228–32
  π systems, delocalised 233–4
  simple 165–6
  solid-state systems 236–40
  thermodynamic properties calculated using 226–8
  torsional terms 173–6
  united atom, reduced representations and 221–5
  water simulation 216–20
  see also non-bonded interactions
force-bias Monte Carlo method 432–3
formaldehyde 76–7, 236
formamide
  electron density around 78–9, 81–2
  gradient vector path 81
  HOMO and LUMO for 79
forward sampling 567
Fourier
  analysis 379, 392–3
  coefficient 148–9
  series 21–3, 155, 235, 237, 392
  transform 22–4, 392
fractional factorial design 697
fragments 472
  binding 689–91
  conformational analysis 464–5, 472
  locating 687–9
  marking 715–17
free energy calculations **563–639**
  chemical reactions 610–22
  computer difficulties 563–4
  enthalpy and entropy differences 574
  linear response method 631–2
  partitioning 574–6
  pitfalls 577–70
  potentials of mean force 580–5
  rapid methods, approximate 585–92
  solid-state defects 622–30
  see also differences, free energy, Helmholtz free energy; solvation
freely rotating chain model 428–9
Frenkel defect 623, 626
friction coefficient 388–9
frontier orbitals 293
full configuration interaction 112
full factorial design 697
fullerenes 101
functional genomics 512
future 160–1, 718–19

GA *see* genetic algorithms
gap penalties 526–8
Gasteiger–Marsili approach 192–3
Gaussian functions/distribution 20–1
  basis sets 65–73 *passim*, 120, 137, 195
  computer simulation 336–7, 339
  conformational analysis 481–2
  density functional theory 131–2
  force fields 195–6
  Gaussian-3 (G3) theory 116–17
  many-body perturbations 116–17
  molecular dynamics simulation 365, 381, 384, 389–90
  new molecules 679, 703
  proteins 551
  SCF 119
  semi-empirical methods 92–8
  solid state quantum mechanics 146
Gay–Berne potential 222–5
GB (generalised Born equation) 598–601
  surface area model (GB/SA) 609
Gear algorithm 358–9
general polyelectronic systems 38–41, 46–50

generalised coordination 370
generalised valence bond 125
Generating Optimal Linear PLS Estimations 711
generator matrices 429
genetic algorithms
  conformational analysis 479–82
  new molecules 653, 663, 691, 701
genomics 512, 548–9
geometry 658–9
  distance 467–75, 476, 651–3, 663
germanium 159–60, 244
Gibbs ensemble Monte Carlo method 439, 450–1
Gibbs free energy 563, 569
global energy minimum 253, 458, 479–83, 551–2
D-glucose 575
glutamic acid 510, 525, 556–7
glutamine 510, 525, 556–7
glycine 221, 459, 511, 525, 556–7
Go–Scheraga chain closure algorithm 541–2
goal nodes 461
GOLD program 667
GOLPE (Generating Optimal Linear PLS Estimations) 711
gradient
  -corrected functional 134–5
  exchange 135, 136, 137
  vector path 80–1
grand canonical Monte Carlo simulations 440–2
graphics, molecular 5–6
graphite, adsorption 441–2
graphs 642–3, 654
Green–Kubo formula 382
GRID program 215, 687–8, 708, 711
grid search 459, 505
GROMOS program 330
Grotthuss mechanism 620
Group 14 elements 244
  solid state quantum mechanics applied to 158–60
  see also carbon; germanium; silicon
group average 493–4
group-based cut-offs 327–30
G3 theory 116–17
guanine 227
GVB (generalised valence bond) 125

haemagglutinin 667
haemerythrin 544
halides 237, 239, 504, 571
halogenated hydrocarbons 707
Hamiltonian operator
  advanced *ab initio* methods 114–15, 120–1
  computational quantum mechanics 27–9, 30, 32, 36, 42, 46, 53, 90–2
  computer simulation 312, 313, 410
  force fields 246
  free energy calculations 565, 567–9, 574, 577–9, 586, 595–6, 614
Hammett substituent parameter 695–7
Hamming (city block) distance 492, 676–8
hard-sphere model 353–4
harmonic approximation 278
harmonic potential *see* Hooke's law
Hartree atomic unit 29
Hartree product 38–9
Hartree–Fock equations/theory 51–65, 85
  application 65
  closed-shell 109
  configuration interaction 112–13
  density functional theory 126, 128, 129, 135–7
  free energy calculations 615–16
  LCAO 56
  many-body perturbation 114–15, 116
  RHF 108–10
  SCF 75, 87, 119
  Slater's Rules 54–6
  solid state quantum mechanics 146–7
  two-electron integrals 19
  see also Fock; Roothaan–Hall equations; UHF
hashed fingerprint 645–6, 677, 678, 685
heat bath, molecular dynamics 384
heat capacity and computer simulation 308–9, 348–9
Heitler–London model of hydrogen 124–5
helium 36–8, 39
  hydrogen molecular ion (HeH^+) 62–5
  Slater determinant for 41
helix 513–15, 583, 584
Hellmann–Feynman theorem 121
Helmholtz free energy 299–300, 411, 563–9
  computer simulation 307, 313–14
Hessian matrix 267–9, 274–5, 280, 282–5, 288
heterovalent substituent 623
heuristic searches and protein prediction 531–4
hexane 449, 462, 463, 672
hexapeptide 482
HF 123, 189, 196
hierarchical cluster analysis 494, 534
high throughput screening (HTS) 641
high-T_c superconductor $YBa_2Cu_3O_{7-x}$ 628–30
Hill potential 209
histidine 169–70, 510, 525, 556–7
HIV-1 protease 666, 667, 691–3
HMMs (Hidden Markov Models) 536–7, 548
Hodgkin–Richards index 679
Hohnberg–Kohn theorem 128
holonomic constraints 370
HOMO (highest occupied molecular orbit) 79, 112, 293–4
Hooke's law 172–3, 275, 486
HP model 518–19
HTS (high throughput screening) 641

Hückel theory 99–102
Human Genome Project 512, 548–9
Hunds rules 35
Hunter–Saunders approach 197–8
hybrid Monte Carlo/molecular dynamics methods 452–3
hydrodynamic vortex 377–8
hydrogen 70
  bonding 122–3, 291–2, 391, 578, 689
    bond order 83
    C–H bonds/interactions 98, 167, 180, 211, 233, 236, 362
    conformational analysis 490, 504–5
    force fields 196, 215–16, 221, 227–8
    new molecules 655, 658
    O–H 98, 620
  configuration interaction 112
  dissociation 109–10
  electron correlation 110–11
  energy minimisation methods 282–3, 291–2
  fluoride 81–2
  Heitler–London model of 124–5
  molecule 41–6
  -suppressed notation 644
hydrogen fluoride 620
hydrophobic effect 515–17, 518–19, 669
hysteresis 577

iceberg model 516
ID3 algorithm 705
ILP (inductive logic programming) 705
image charge computer simulation 340–1
immunoglobulin 544
immunosuppressant FK506 229–30
importance sampling 410
independent (random) samples 345
indicator variable 696
INDO (intermediate neglect of differential overlap) 86, 92–3, 94–6
inductive logic programming 705
initial configuration, prior to simulation 315–16
inorganic molecules, force fields for 234–6
Inorganic Structural Database 489
inside-out ligand design 687–8
integration
  algorithms for molecular dynamics simulation 359–60
  calculating properties by 412–14
  thermodynamic 568–9, 574, 577, 630–1
intermediate neglect of differential overlap 86, 92–3, 94–6
intermolecular processes and energy minimisation 278–9
internal coordinates 2–4, 257
Internet and World Wide Web 9–10, 548, 553
interstitials 622–3, 627
intrinsic reaction coordinate 288–9
inverse agonists 640
inverse of matrix 15–16
ionic solids, force fields for 238–40
ionisation potentials 74–5
IRC (intrinsic reaction coordinate) 288–9
iron, liquid 621–2
Isis system 645
island model 481
isodemic reactions 116
isoleucine 511, 525, 556–7
isomerism, subgraph 645
isothermal–isobaric ensemble, definition 307, 563, 569

Jahn–Teller effect 234
Jarvis–Patrick algorithm 496–7, 683
JBW (jumping between wells) 435
jellium 244
jump frequency 627–8
jumping between wells (JBW) 435
J-walking 533–5

kappa shape/kappa-alpha indices 672–4
keys, pharmacophore 674–6
Kohn–Sham scheme/orbitals 128–9, 131, 132, 134, 135–7, 156, 157, 616
Koopman's theorem 74–5
Kronecker delta 30, 396
kurtosis 680

lag, *ab initio* molecular dynamics 618
Lagrange multiplier 18, 52, 127, 191, 371–2
lags 453
Laguerre polynomials 31, 55
lambda dynamics 585–8
Langevin dipole method 601–3
Langevin equation 388–9, 391, 580, 601–3, 616
Langmuir–Blodgett films/layers 395, 400–2
large structures, reaction path for 289–92
large systems, deriving charge models for 191–2
latent variables 706
lattices
  models of polymers 424–8
  models for proteins 518–20
  solid state quantum mechanics 138–60 *passim*
  statics and dynamics in energy minimisation 295–300
LCAO (linear combination of atomic orbitals) 41–2, 56, 100, 241
LDA/LSDA (local (spin) density approximation) 130–1
leap-frog algorithm 356–7
least-squares approach 230–1
leave-one-out 701

Lee–Yang–Parr *see* BLYP
Legendre polynomials 32
length, units of 9
Lennard-Jones potential 253
  computer simulation 305, 319, 324, 327, 331–3, 341–2
  force fields 167, 207–10, 212, 214–16, 225–6, 237
  free energy calculations 579, 586, 613
  molecular dynamics simulation 361, 368, 402
  Monte Carlo simulation 428, 439, 441, 448, 450
LES (locally enhanced sampling) 575–6
leucine 487, 511, 525, 556–7
Levinthal paradox 550
libraries, combinatorial 711–19
LIE (linear interaction energy) 588–9
line search in one direction 262–3
linear combination of atomic orbitals 41–2, 56, 100, 241
linear congruential method 418–19
linear interaction energy 588–9
linear potential, piecewise 665
linear regression 666, 698–9, 702
linear response 588–9, 591, 631–2
linkage methods 493
lipids 338
  simulation of 397–400
liquid crystals 222–3
literature 9
lithium 111, 238, 323, 626
loadings 682
local density approximation 130–1
local spin DFT 129, 135
locally enhanced sampling 575–6
logP 668–70
London force 204–5
long-range correction 327
long-range forces and computer simulation 334–43
long time-tails, molecular dynamics 377
loop conformations 541–2
Lorentz–Berthelot mixing rules 210
low-mode search 478–9
Löwdin population analysis 80
lower-energy regions 564
lowest unoccupied molecular orbit 79, 112, 293–4
LR (linear response) 588–9, 591, 631–2
LSDFT (local spin density functional theory) 129, 135
LUDI program 689
LUMO (lowest unoccupied molecular orbit) 79, 112, 293–4
lysine 510, 525, 556–7

MACCS system 645
Maclaurin series 11
magnesium 238, 623, 626
many-body
  effects in empirical potentials 212–14
  perturbation theory 114–17
  potentials 241
mapping
  adiabatic 286
  distance 651
  pharmacophore 648
  Ramachandran 459–60, 514, 543, 547
marker atom 329–30
Markov chain 414–15
Markov models, hidden 536–7, 538
Marsaglia random number generator 420, 453–4
mass-weighted coordinates 274–5
mathematical concepts 10–24
  complex numbers 16–18
  multiple integrals 19–20
  series expansions 10–11
  statistics 20–1
  see also eigenvalues; Fourier; Lagrange; matrices; vectors
matrices 2–3, 9, 12–16, 415
  adjacency 647
  charge density 58–9
  distance 652
  elastic constant 296–7
  PAM 524–6, 531, 556–7
  positive definite 16, 258
  statistical weight 430
  stochastic 415
  see also Fock matrix; Hessian; Z-matrix
maxima 273
maximal segment pair 531–3
maximum dissimilarity algorithms 683–4
maximum likelihood method 657–8
MaxSum and MaxMin 683–4, 685
Maxwell–Boltzmann distribution 365, 367, 384
Mayer bond order 83
MC *see* Monte Carlo
MCSCF (multiconfiguration SCF) 113
MCSS (multiple-copy simultaneous search) 688
MDL (Molecular Design mol) format 643–4
mean field approach 307–9
mean squared displacement 322–3
mean square end-to-end distance, polymers 426
mechanics, molecular *see* force field
mesoscale modelling 402–4
messenger RNA (mRNA) 509
metals 147, 589, 607, 626, 649–50, 693
  force field potentials for 240–5
met-enkephalin 517
methane
  bond order 83
  force fields 189
  Monte Carlo simulation 441–2

methane (*cont.*)
octopole moment 76
population analysis 79
SMILES notation 644
methanol 573
methionine 511, 525, 556–7
methoxypromazine 678
methyl chloride 612–13, 614, 676–7
2-methyl propane 644
o-methylacetanilide 670
methylalanine 583–4
methylene group 160, 162, 330, 396, 448
energy minimisation methods 280, 291
force fields 181, 221
4-methyl-2-oxetanone 137
metric matrix 469
metrisation 472
Metropolis Monte Carlo simulation 306, 433, 436, 437, 447
conformational analysis 467, 505
implementation 417–20
new molecules 663, 685, 691
proteins 518
theoretical background 414–16
microcanonical ensemble, definition 307
MINDO/3 86, 94–6, 102–3
minima 272–3
minimal basis set 69–70
minimisation *see* energy minimisation
minimum image convention and computer simulation 324–34
mixing rules 210
MM2/MM3/MM4 programs 8, 615
force fields 169–71, 173, 176, 179, 187, 211, 233–4
MNDO (modified neglect of diatomic overlap) 86, 96–7, 98–9, 102–3, 192
MOD function 418–19
Modeller program 541, 549
modified INDO (MINDO/3) 86, 94–6, 102–3
modified neglect *see* MNDO
molar refractivity 671
molecular dynamics simulation **354–409**, 623
of chain amphiphiles 394–404
computer simulation 305–6, 307
conformational analysis 457, 475–6, 483–9
conformational changes from 392–3
constant pressure dynamics 385–7
constant temperature dynamics 382–5
constraint dynamics 368–74
continuous methods 355–64
energy conservation in 405–6
ensemble 653
free energy calculations 564, 572, 577, 579, 581, 588, 616–22, 628
Monte Carlo compared with 307, 387, 452–3
new molecules 664
proteins 552
setting up and running 364–8
simple models 353–4
solvent effects 387–90
time-dependent properties 374–82
see also computer simulation
molecular field analysis 708–11
molecular fitting 490–1
molecular fragments *see* fragments
molecular modelling *see* advanced *ab initio*; computer simulation, concepts, conformational analysis; energy minimisation, force field; free energy; molecular dynamics, Monte Carlo, new molecules; proteins; quantum mechanics
molecular orbital theories, semi-empirical 86, 89–96, 102–3
molecular surface *see* surface
Möller–Plesset *see* MP
moments theorem 241–2
monomers 289–90, 423, 550
new molecules 712–13, 717–18
Monte Carlo
configurational bias 443–50
force-bias 432
Grand canonical 440–2
smart 432
Monte Carlo simulation **410–56**
bias 432–3, 443–50
chemical potential, calculating 442–3
computer 306–7
conformational analysis 457, 475–6, 479, 483, 504–5
density functional theory 130
different ensembles, sampling from 438–42
force fields 189
free energy calculations 564, 577, 579, 588
chemical reactions 613, 616
PMF 581–2, 584
solid-state defects 623, 628
thermodynamic perturbation 572–3
Gibbs ensemble 450–1
integration, calculating properties by 412–14
molecular dynamics compared with 307, 387, 452–3
molecules 420–3
new 662–3, 685, 691
polymers 423–31
proteins 517–19, 551
quasi ergodicity 433–8
random number generators 418–20, 453–4
see also computer simulation, Metropolis
MOPAC program 8, 99
Morgan algorithm 644
Morokuma analysis 122–4

Morse potential/curve 170–2, 210
motifs 522
Mott–Littleton method 623–4, 625–7
MP (Möller–Plesset) perturbation theory 114, 115–16, 119
MR (molar refractivity) 671
MS (Murtaugh–Sargent) method 269–70
MSP (maximal segment pair) 531–3
Mulliken population analysis 79–80, 189
multicanonical Monte Carlo simulation 435–8
multiconfiguration SCF 113
multiple integrals 19–20
multiple linear regression 666, 699, 702
multiple sequence alignment 534–7
multiple-copy simultaneous search 688
multipole
  electric, calculation of 75–7
  fast 341–3, 364
  models 195–7, 219
multivariate problems 708
Murtaugh–Sargent method 269–70
mutation
  operator 480
  probability matrices for proteins 556–7

naphthalene 233
NCC (Nieser–Corongiu–Clementi) model 219–20
NDDO (neglect of diatomic differential overlap) 86, 93–4, 95, 96
nearly free-electron approximation 142, 147–53
Needleman–Wunsch algorithm 526–9, 534
neglect of differential overlap 86, 89–96
neighbour lists 325–7, 493–6
net dipole moment 378–9
net (partial) atomic charges 157–9, 181
netropsin 270–1
neural networks and QASR 703–5
new molecules **640–726**
  combinatorial libraries 711–19
  computer representations 642–7
  de novo structure based ligand design 687–94
  descriptors 668–79
  discovery of drugs 640–1
  diverse sets of compounds, selecting 680–7
  docking 661–8, 689
  partial least squares 702, 706–11
  similarity 668, 676–9
  3D 674–5, 687
    databases 659–61, 679
    pharmacophores 648–59, 674–5, 687
    searching 645, 647, 667–8
    similarity 678–9
  see also QSAR
Newton–Raphson energy minimisation 267–8, 270, 288, 625
Newton's laws 304, 309, 353, 366, 371
niching 481
nickel oxide 147
nicotine/nicotinic pharmacophore 653, 678
nitrogen 490
  amide 660
  basis sets 73
  bond order 83
  charge models 187–8
  distributed multipole model 196
  electrostatic potentials 188
  force fields 181
  substituents 693
NM23 547
NMR and X-ray crystallography 316
  conformational analysis 468, 474–5, 483–9, 490
  molecular dynamics simulation 379, 383, 395
  new molecules 647, 659, 661, 667, 689, 691, 693, 704, 713
  proteins 516, 522, 546–7, 552, 512.514
nodes
  on graphs 642–3
  on search trees 461
NOESY (nuclear Overhauser enhancement spectroscopy) 474–5, 486, 488
non-bonded cutoffs 324–34
non-bonded interactions 166, 181–212, 324
  cell multipole method for 341–3
  electrostatic 166, 181–204
  neighbour lists 325–7
  Van der Waals 166, 204–12
non-derivative energy minimisation 258–61
non-electrostatic contributions to solvation free energy calculations 608–9
non-holonomic constraints 370
non-periodic boundary methods 320–1
normal distribution *see* Gaussian functions
normal mode analysis and energy minimisation 273–8
normal vibrational modes 274
nuclear Overhauser *see* NOESY
nucleic acids 196–7

1-octanol and water, partition between 668–9
octopole 76, 181
one-electron
  atoms 30–4
  integrals 50–1
ONIOM approach 615
Onsager dipole model 593–5
open-shell systems 108–10
operators 28–9, 53, 57, 114, 480–1
  see also Hamiltonian
OPLS (optimised parameters for liquid simulations) 210, 228, 599

orbital
-based approach to band theory 142–6
calculations, molecular 26, 41–51
approximate theories 86
energy of closed-shell system 51
energy of general polyelectronic system 46–50
hydrogen 41–6
one- and two-electron integrals 50–1
semi-empirical 86, 89–96, 102–3
total electron density 77–9
see also STOs
electronegativity 192–3
linear combination of atomic 41–2, 56, 100, 241
virtual 61
see also Kohn–Sham
order
bond 81–3
order, of integration algorithm 358
parameters 321–2
orientational correlation 379–80
orthogonalisation, symmetric 60
orthonormal wavefunctions 30
oscillating charge 201–2
out-of-plane bending 176–8
outside-in ligand design 687–8
overlap
differential 86, 88–96
forces *see* repulsive forces
integral 52
oxides 147, 238
oxygen bonds/interactions 98, 237, 328, 620, 652

pairwise potential models 240–1
PAM matrices 524–6, 531, 556–7
parameters 567, 599
force field 221, 224–5, 228–32
substituent 695–7
see also Verlet
partial equalisation of orbital electronegativity 192–3
partial least squares (PLS) 702, 706–11
partial (net) atomic charges 157–9, 181
partition/partitioning 683–5
coefficients 572–3, 668–71
electron density 80–1
free energy 574–6
pattern recognition 491–7
Pauli principle 206
PCA (principal components analysis) 497–9, 681, 686
PCM (polarisable continuum method) 596–7, 598
PDB (Protein Databank) 489–90, 539
pdf (probability density function) 304, 541
Pearson correlation coefficient 681
penalty functions 483
pentane 253, 430, 462, 582
pepsin 545
peptides/polypeptides 277, 423, 509, 515, 517–18, 520, 691
conformational analysis 459, 482
dynamic programming 527–8
folding 552
force fields 196–7, 221, 231
free energy calculations 571, 583–4
loop conformations 541–2
peptoids 713
'threading' 546
see also amino acids; proteins
percentage sequence identity 524
pericyclic reactions transition structures 292–5
periodic boundary conditions 317–19
perturbation and free energy 566–8, 573–4, 582, 584, 595
thermodynamic 564–6, 569–73, 577, 592
perturbation theories 36, 114–17, 119
pharmacophore
mapping 648
keys 674–6
pharmacophores 647
see also new molecules
phase equilibria, simulation of 450–1
phase problem, in X-ray crystallography 484
phase space and computer simulation 312–15
phenylalanine 169, 286, 511, 525, 542, 546, 556–7
phonons and dispersion curve 298–9
π systems 197–9
benzene 126
delocalised 233–4
pivot algorithm 423
plane waves 155–6
PLS (partial least squares) 702, 706–11
PM3 98–9, 102
PMF (potentials of mean force) 387–90, 546, 580–5, 612–14
point defect 622
point-charge electrostatic models 187
Poisson equation 133
Poisson–Boltzmann equation 603–8
polarisation/polarisable basis functions 71
continuum method 596–7, 598
electrostatic non-bonded interactions 199–202, 203
energy component analysis 122
force field models for simulation of water 218–19
poling and conformational analysis 499–501
polyatomic systems 210–12
polyelectronic atoms and molecules 34–41

polymers
energy minimisation methods 289–90
free energy calculations 621, 622
molecular dynamics simulation 391, 404, 550, 551
Monte Carlo simulation of 423–31
see also amino acids; peptides; proteins
population analysis 79–80, 189
porphyrins 197–8
positive definite matrix 16, 268
potential 156–7, 275, 486, 546
computational quantum mechanics 74–5, 83–5
computer simulation 305, 319, 324–34, 338, 341–2
electrostatic 83–5
energy 4–5, 238, 253
force fields 167, 170–3, 188–92, 207–10, 212–17, 222–6, 237–8, 240–5
free energy calculations 549, 579, 580–5, 586, 612–14
ionisation 74–5
of mean force *see* PMF
models, pairwise 240–1
molecular dynamics simulation 387–90
Monte Carlo simulation 442–3
new molecules 665, 666
prediction of crystal structures 501–5
predictive residual sum of squares 701
predictor-corrector methods of molecular dynamics simulation 358–9
see also under proteins
preferential sampling 432
PRESS (predictive residual sum of squares) 701
pressure 309, 385–7
principal components analysis *see* PCA
principal components regression 706
probability density function 304, 541
probability matrices for proteins 556–7
product-based monomer selection 718
production phase in simulation 315
profile 535
proline 221, 511, 525, 556–7
PROMET 502–3
propane 167, 644
proteins 6, 423
computer simulation 329–30, 338–9
conformational analysis 475, 489–90
force fields 192, 221
free energy calculations 571
predicting structure of **509–62**
acronyms and abbreviations 553–4
basic principles 513–17
comparative model 539–45
comparison of methods 547–9
databases, list of 555
first principles methods 517–22
folding and unfolding 512, 516–17, 539, 545–7, 549–53
mutation probability matrices 556–7
sequence alignment 522–39
threading 545–7
Protein Databank 489–90, 539
see also amino acids; peptides
pseudo-acyclic molecules 463–4
pseudopotentials 156–7
pyrazine/pyridine 573
2-pyridone 597–8

Q^2 (cross-validated R^2) 701
QCISD (quadratic CISD) 113, 117, 119
QSAR (quantitative structure-activity relationships) 695–706, 710, 711
cross-validation 701
deriving equation 698–70
discriminant analysis 703–5
interpreting equation 702
neural networks 703–5
principal components regression 706
-property relationship 695, 702
selecting compounds for analysis 697–8
QSPR (quantitative structure-property relationship) 695, 702
quadratic region 283–4
quadrupole 76, 181, 183, 185–6, 196
quantitative structure-activity *see* QSAR
quantum mechanics
future role 160–1
and molecular mechanics combined in chemical reactions 614–16
solvation, free energy of 594–8
see also ab initio quantum mechanics; advanced *ab initio*, computational quantum mechanics
quasi ergodicity and Monte Carlo simulation 433–8
quasi-Newton energy minimisation 268–9
quaternions 422

R^2 699
R groups 716–17
radial distribution functions and computer simulation 310–12
Ramachandran map 459–60, 514, 543, 547
random number generators 418–20, 453–4
random sampling 345–7
random search 465–7, 476, 517–18
random tweak 542
range scaling 681
ranitidine 489, 644
RANTES 475
rapid free energy calculations, approximate 585–92

Rappé–Goddard method 193–4
RATTLE method 373–4
Rayleigh–Schrödinger perturbation theory 114
reaction
  field 339–40, 595–6, 597
  isodemic 116
  pathways 279–95
  transform 715–16
  zone 320–1
  see also chemical reactions
real gas contribution to virial 309, 349–50
reciprocal lattice 139–40
recombination operator 480–1
reduced units, in non-bonded interactions 212
re-entrant surface 7
refractivity, molar 671
regression 706
  equation 698–9
  linear 666, 698–9, 702
relative energies 226
relaxation time 376
reptation 427
repulsive forces 206
  see also Coulomb attraction/repulsion
residual sum of squares 699–700
RESP (restrained electrostatic potential fit) 191–2
response 697
restraints/restrained
  and constraints, difference between 369–70
  electrostatic potential fit 191–2
  molecular dynamics 483–4
  spatial, satisfaction of 540–1
reversible reference system 363–4
RHF (spin-restricted Hartree–Fock theory) 108–10
ribose phosphate 493–4
rigid molecules, simulation of 420–2
rigid-body method 540
ring critical point 81
RIS (rotational isomeric state) model 429–31
RMS (root-mean-square) 273, 359–60, 552, 667
RMSD (root-mean-square-distance) 491–3
RNA 509, 512
root nodes 461
root-mean-square 273, 359–60, 552, 667
Roothaan–Hall equations
  closed-shell systems 56–9, 86–8
  density functional theory 132
  illustrated 62–5
  solving 59–62
Rosenbluth weight 444–7
rotational isomeric state 429–31
rotational order 322
roulette wheel selection 480
r-RESPA (reversible reference system propagation algorithm) 363–4
RSS (residual sum of squares) 699–700
rule-based approaches to protein prediction 520–2

saddle points 253, 272–3, 280, 282–3, 291
  location 285–8, 478
  quadratic region 283–4
SAM1 (Semi-Ab-initio Model 1) 99, 102
sampling 345, 346–7, 410, 432, 438–42, 567–8, 575–6
SC24/halide system 571
scaling/scaled
  autoscaling 681
  coordinates 438–9
  mesoscale modelling 402–4
  particle theory 609
  range 681
scalar product and triple product 12, 14
SCF (self-consistent field) 117, 280
  complete active-space 113, 295
  computational quantum mechanics 54, 64, 73, 75, 87
  direct method 118–20
  energy component analysis 122
  free energy calculations 595–6, 597
  Hartree–Fock 75, 87, 119
  multiconfiguration 113
Schottky defect 622–3, 626
Schrödinger equations and solutions to
  computational quantum mechanics 27–8, 29–30, 32, 34–7 *passim*, 128
  computer simulation 347–8
  density functional theory 127, 128
  for Drude molecules 205, 246–7
  solid state quantum mechanics 147, 148
SCOP (Structural Classification of Proteins) 539
scoring functions for docking 664–7
SCRF (self-consistent reaction field) 595–6, 597
SCRs (structurally conserved regions) 539–40
SDEP measure 709
search
  depth-first 462, 663
  grid 459, 505
  heuristic 531–4
  line 262–3
  low-mode 478–9
  multiple-copy 688
  new molecules (3D) 645, 647, 667–8
  random 465–7, 476, 517–18
  systematic 458–64, 476, 505
  trees 461–5
  see also under conformational analysis
second-moment approximation 242
secondary structure of proteins 513
segment matching 540
self-consistent field *see* SCF; SCRF
self-penalty walk (SPW) 289–90, 584

Semi-Ab-initio Model 1 (SAM1) 99, 102
semi-empirical methods of computational quantum mechanics 65, 86–99, 102–3
semi-empirical molecular orbital theories 86, 89–96, 102–3
semiconductors, force field potentials for 244–5
separation of variables 36–7
sequence alignment of proteins 522–39
sequence identity 546–7
sequential univariate minimisation 260–1
series expansions 10–11
serine 511, 525, 556–7
SHAKE procedure 369–74, 582, 618
shape anistropy parameter 224–5
SHAPES force field 235–7
shear viscosity 381
shielding constant 55–6
shifted potential 330–1
shorthand representation of electron integrals 50
sigmoidal dielectric model 202–4
silica 297–8
silicalite 449–50
silicon 483, 693
  -O bond 237
  and chlorine 620–1
  force fields 237, 245
  phases of 159–60
  shielding constant 55–6
  valence electron density 160
similarity
  calculating 676–8
  searching 668
  and 3D properties 678–9
simple force field models for simulation of water 216–18
simplex method of non-derivative energy minimisation 258–60
Simpson's rule 412–13
simulated annealing
  in conformational analysis 483
  in *ab initio* molecular dynamics 616–18
  in X-ray refinement 484–6
simulations *see* computer simulation; conformational analysis; molecular dynamics; Monte Carlo
SINDO1 program 99
single-linkage cluster algorithm 493–6
site points 689
skewness 680
Slater determinants
  density functional theory 135, 136
  general polyelectronic systems 38–41
  many-body perturbation 115
  orbitals *see* STOs
  Slater functions and basis sets 67–9
  Slater's Rules and Hartree–Fock equations 54–6
slow growth free energy calculations 568–9, 577, 631
smart Monte Carlo method 432–3
Smart Region Definition 710–11
SMILES notation 643–5, 715
Smith–Waterman algorithm 529–30
S_N2 reaction
  potential of mean force 612–14
  transition state of 280–2
sodium 181, 589, 626
sodium chloride 238
Soergel distance 676–8
solid-state
  defects and free energy calculations 622–30
  energy minimisation methods 295–300
  force fields for 236–40
  quantum mechanical methods for studying 138–60
solvation/solvents 320
  dielectric models of electrostatic non-bonded interactions 202–4
  free energy of 576, 592–610
    continuum models 592–3, 598–601
    electrostatic contributions 593–608
    non-electrostatic contributions 608–9
    simple models 609–10
  molecular dynamics simulation 387–90
  Monte Carlo simulation 432, 452
space group 138
spatial restraints, satisfaction of 540–1
SPC model 216–18
sphere-exclusion algorithm 684
spherical cut-off 324
spherical harmonic 30–1
spin
  -coupled valence bond theory 125–6
  density 109
  local 129, 135
  orbitals 35
  -polarised density functional theory 129
  -restricted Hartree-Fock 108–10
  -unrestricted Hartree-Fock 108–10
split valence double zeta basis sets 70
SPV (self-penalty walk) 289–90, 584
squares
  least-squares approach 230–1
  partial least 702, 706–11
  root-mean 273, 359–60, 552, 667
  square-well potentials 354
  sum of 699–701
SRD (Smart Region Definition) 710–11
ST2 potential 217
standard deviation 20
statistical inefficiency 346

statistics/statistical 20–1
  mechanics and computer simulation 347–8
  weight matrix 430
steady state genetic algorithm 481
stearic acid 394, 400–1
steepest descent energy minimisation 262
step size 264
steric energy 226
Stillinger–Weber model 241, 244–5
stochastic boundary conditions 320–1
stochastic collisions method 384
stochastic dynamics simulations 390–2
stochastic matrix 415
Stokes law 388
STOs (Slater-type orbitals) 46–8, 55, 72
  density functional theory 131–2
  force fields 194
  STO-*n*G basis sets 62, 69, 85, 123
strain 296
  energy 226–7, 627
stratified sampling 346
streptavidin 641
stress 296
Structural Classification of Proteins 539
structural databases
  conformational analysis 482, 489–90, 493–4, 499
  proteins 537–9, 555
structural genomics 512
structural key 645
structural properties, calculating 85–6
structurally conserved regions 539–40
structurally variable regions 539–40
structure factor 484
subgraph 642–3, 645
subset selection 717–18
substitutionals 623
substructure search 465, 642–7
sulphur dioxide 117
sum of squares 699–701
SUMM (systematic unbounded multiple minimum) 477–8
superfamily (proteins) 539
superoxide dismutase 607
surface 6–8
  area model 609
  energy 4–5, 253, 475
  Fermi 153–5
  van der Waals 7, 600
Sutton–Chen potential 241, 243
SVRs (structurally variable regions) 539–40
SWISSPROT database 537
switching function 331–4
symmetric matrix 13
symmetric orthogonalisation 60
systematic sampling 346–7
systematic search 649–51
  conformational 458–64, 476, 505
systematic unbounded multiple minimum 477–8

Tanimoto coefficient 676–8, 685
Taylor series 10–11, 230, 267, 342, 355, 358, 439, 592
temperature 240
  computer simulation 309–10
  molecular dynamics simulation 368, 382–5
template forcing 491
templating effect 694
tensor properties 183
terminal nodes 461
Tersoff model 241, 244–5
thermodynamic(s)
  computer simulation 307–12
  cycles 569–70
  force fields 226–8
  integration 568–9, 574, 577, 630–1
  perturbation 564–6, 569–73, 577, 592
  properties 85–6, 307–12
thermolysin 571–2
thiazole 490
Thomas–Fermi model 127
threading, predicting proteins by 545–7
three-body problem/effects 34, 212–14, 244
3D
  profiles 543–4, 547
  see also under new molecules
  3_{10} 513, 583–5
threonine 511, 525, 546, 556–7
thrombin 522–3
thymidylate synthase 667
thymine 227
'TIM barrel' 522–3
time
  -averaged NMR 487–9
  averages 303–5
  correlation coefficients 374
  -dependent properties and molecular dynamics simulation 374–82
  step and molecular dynamics simulation 360–4
TiN 155
TIP3P/TIP4P models 216–17, 219, 327–8
topological indices 671–4
torsion/torsional 166
  angle/bend 3–4, 179–80, 254, 515
  driving 286–8
  improper 176–8
  parameters 229
  terms 173–6
total electron density and molecular orbitals 77–9
total sum of squares 699–700
Toxvaerd anisotropic model 221–2
transfer RNA (tRNA) 509

transferability and force fields 168
transition structures 255, 279–95
transport and molecular dynamics 380–2
transpose of matrix 15
trapezium rule 412–13
tree representation 461–2
trial and error and parametrisation 228–9
triangle smoothing 468–9, 660–1
sym-triazine 198–9
1,3,5-trifluorobenzene 198–9
trimethoprim 278–9
truncating potential 324–34
trypsin 522–3, 587, 607
tryptophan 169, 511, 525, 556–7
TSS (total sum of squares) 699–700
Tversky similarity 677–8
twin-range method 327
2D substructure searching 642–7
two-electron integrals 50–1
tyrosine 286, 510, 525, 542, 556–7

UFF (Universal Force Field) 193–4, 232, 235, 237
UHF (spin-unrestricted Hartree–Fock theory) 108–10
Ullmann algorithm 646–7
umbrella sampling 581–2, 584
underlying matrix 415
unit cell 138
united atom force fields and reduced representations 221–5
Universal Force Field *see* UFF
uracil-2,6-diaminopyridine (DAP) 227–8
Urey–Bradley force field 179, 235

vacancy 622, 627
  formation energy 240
valence
  band 142
  bond theories 124–6
  electron density 160
  split 70
valine 511, 525, 546, 556–7
van der Waals
  energy 253
  interactions 320, 486
    force fields 166–7, 204–12, 231, 237
    free energy calculations 566–7, 576, 586, 588–9
  parameters 229
  potentials 666
  radii 84, 470, 592, 596, 660
  surface 7, 600
variable metric *see* quasi-Newton
variables (factors) 697
variance 20
variance–covariance matrix 498
variation
  coefficient of 680–1
  theorem 51–2
vectors 11–12
  path, gradient 80–1
  product 12, 14
Veillard–Baron order parameter 322
velocity autocorrelation 376–8
velocity Verlet algorithm 357
Verdier Stockmayer algorithm 427
Verlet algorithm/parameters 321–2, 325–6, 355–9, 403–4
  see also SHAKE
vibrational modes 274
virial 309, 349–50
virial theorem of Clausius 309
virtual molecules 642
virtual orbitals 61
virtual screening 715
VWN (Vosko–Wilk–Nusair) standard local correlation 136

water
  bond order 83
  and carbon dioxide 616–17
  computer simulation 317, 327–8
  dimer analysis 123–4, 327–8
  force field models 201, 216–20
  free energy calculations 573
    differences 569–70
    see also solvation
  infrared spectra 379
  normal modes of 274
  and 1-octanol, partition between 668–9
wavefunction 41–6, 161
Wigner–Seitz cells 140, 350
Wiswesser line notation 643–4
Woodward–Hoffmann rules 102, 292–3, 295
World Drug Index 685
world wide web (WWW) 9–10

X-ray crystallography *see* NMR

YETI force field 215–16

ZDO (zero-differential overlap) 88–9, 91
ZEBEDDE (ZEolites By Evolutionary De novo DEsign) 694
zeolites 298, 449–50
  force fields for 236–7
  synthesis 693–4
zero-differential overlap 88–9, 91
zero-point energy 274
zinc 607, 649–50
ZINDO program 99
Z-matrix 2–3, 9, 74, 255, 271–2
Zwanzig expression 588, 631–2

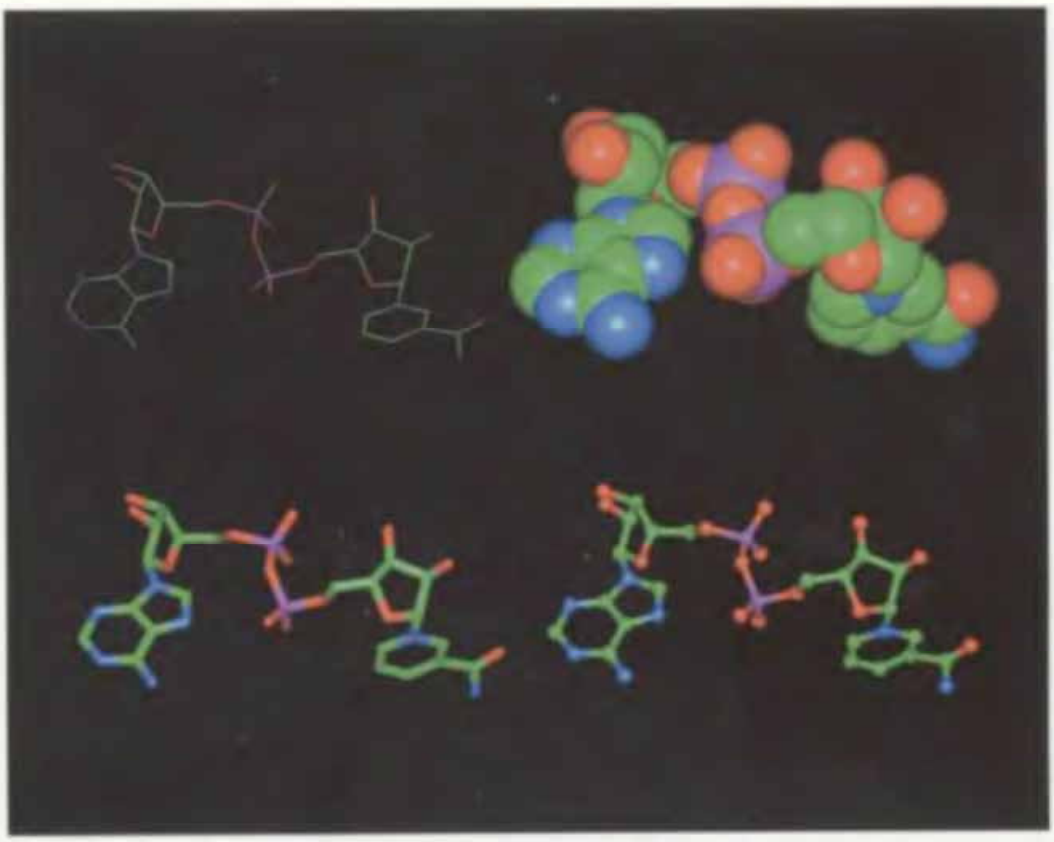


Fig. 1.4: Some of the common molecular graphics representations of molecules, illustrated using the crystal structure of nicotinamide adenine dinucleotide phosphate (NADPH) [Reddy et al. *1981]. Clockwise, from top left: stick, CPK/space filling, 'balls and stick' and 'tube'.*

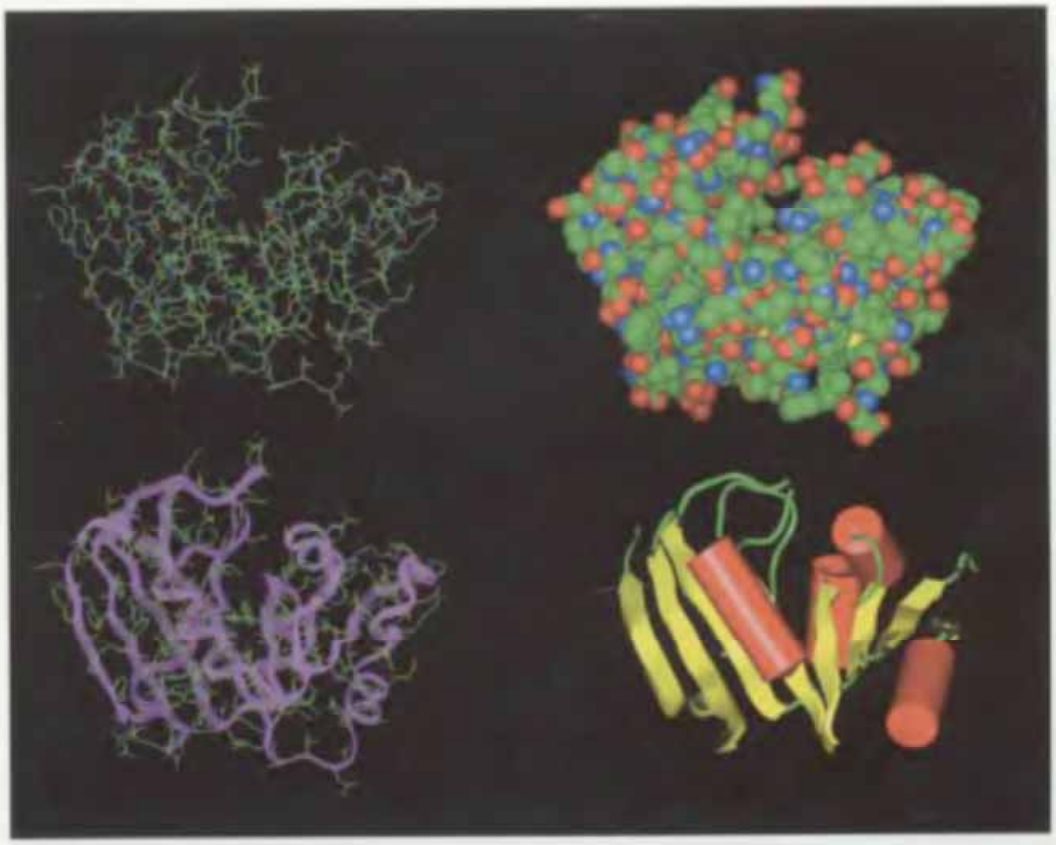


Fig. 1.5: Graphical representations of proteins illustrated using the enzyme dihydrofolate reductase [Bolin et al. *1982]. Clockwise from top left: stick, CPK, 'cartoon' and 'ribbon'.*

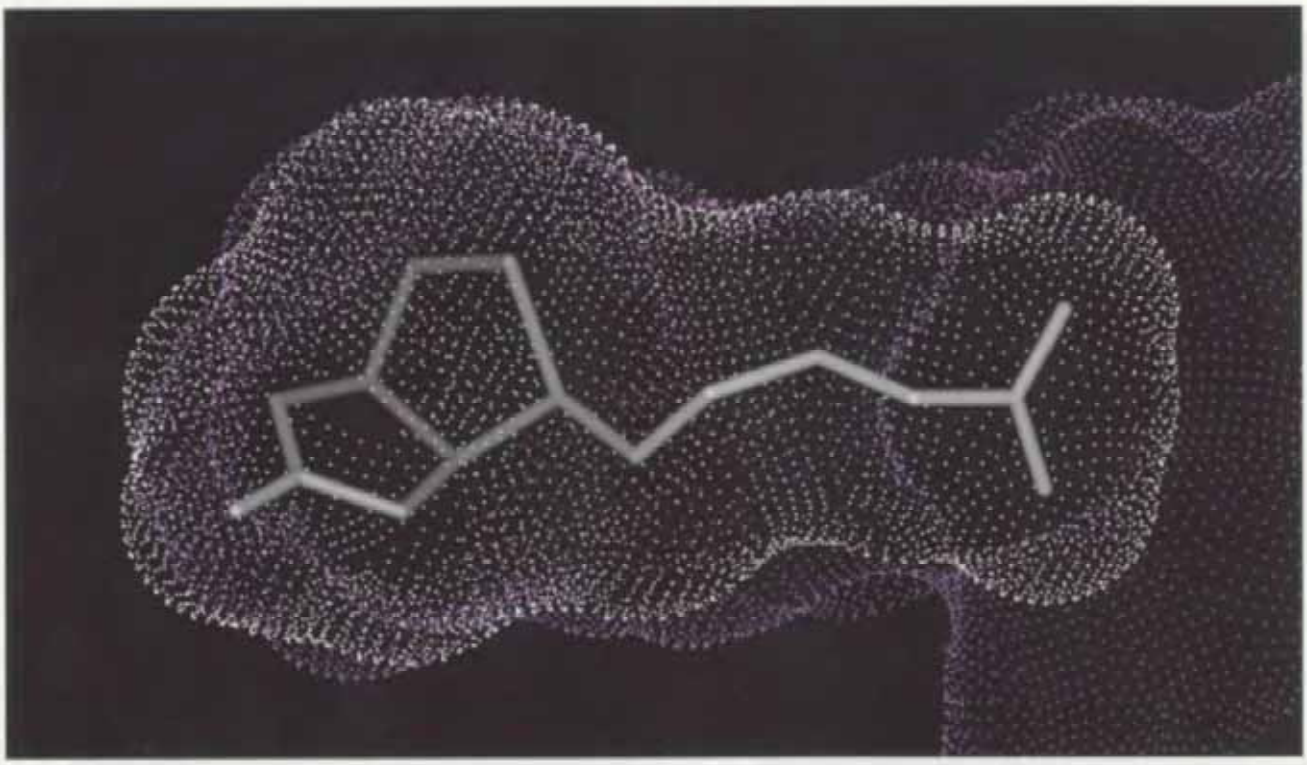


Fig. 11.12: Surface complementarity of the protein streptavidin (purple) and the ligand biotin (white) [Freitag et al. 1997].

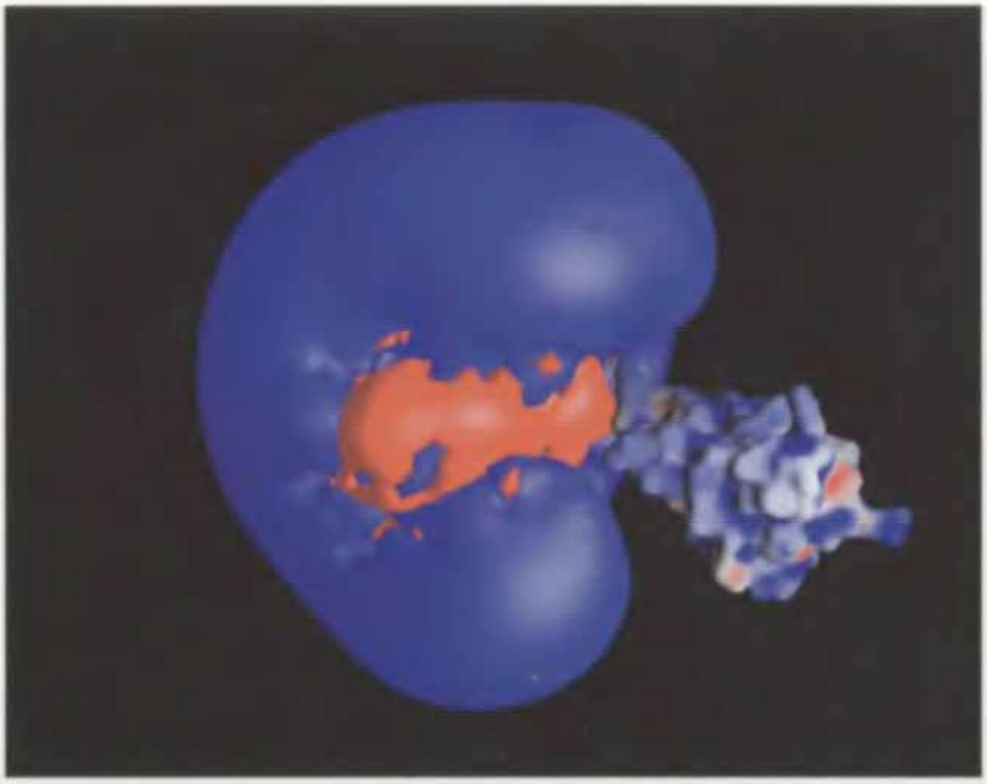
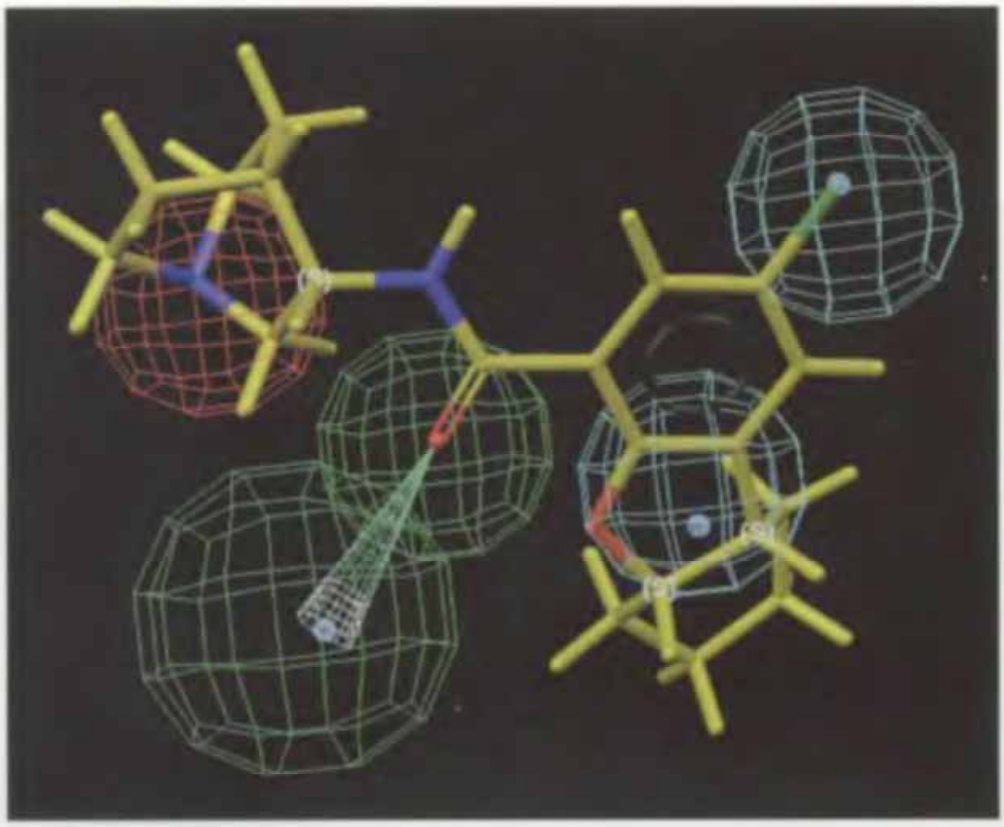


Fig. 11.29: 3D Electrostatic isopotential contours around trypsin [Marquart et al. *1983]. Contours are drawn at $-1k_BT$ (red) and $+1k_BT$ (blue). The trypsin inhibitor is also represented with its electrostatic potential mapped onto its molecular surface.*

Fig. 11.30: Electrostatic potential around Cu–Zn superoxide dismutase [McRee et al. *1990]. Red contours indicate negative electrostatic potential and blue contours indicate positive electrostatic potential. Two active sites are present in each dimer, at the top left and bottom right of the figure where there is a significant concentration of positive electrostatic potential.*



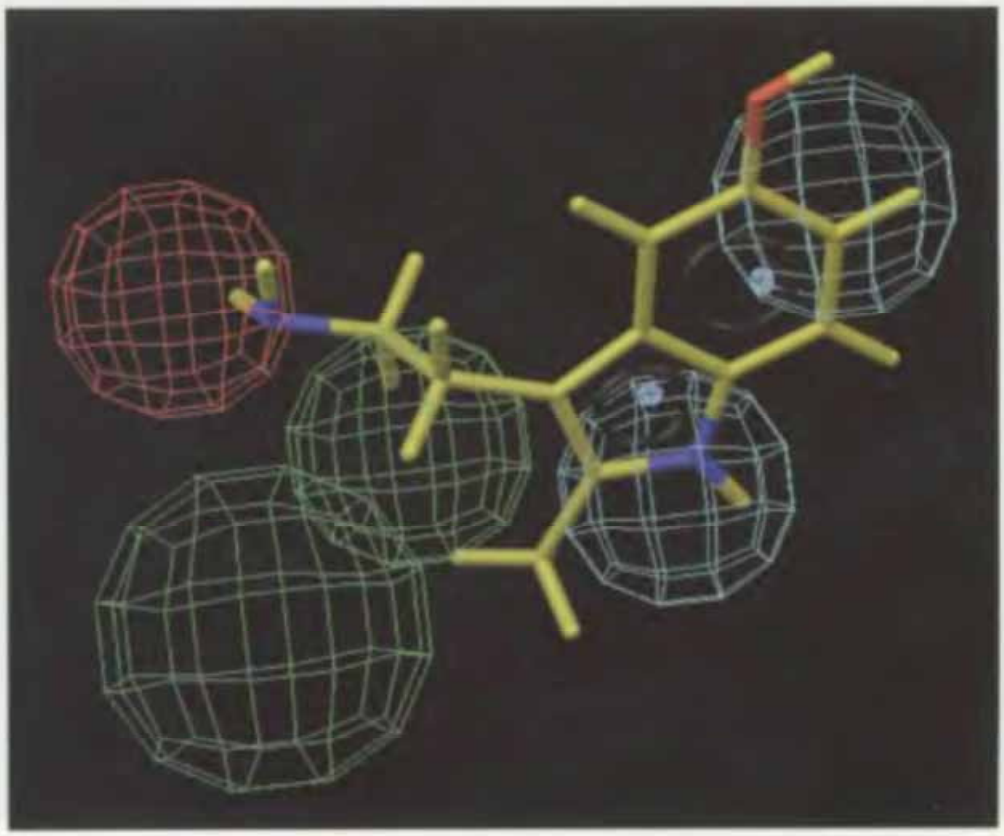


Fig. 12.16: 3D pharmacophore derived for a series of molecules with activity at the $5HT_3$ receptor. The spheres indicate location constraints where an appropriate pharmacophore group should be located (red: positively ionisable, green: hydrogen-bond acceptor, blue: hydrophobic region). The figure shows a very active molecule, JMC-35-903-10 superimposed on the pharmacophore (top) and a much less potent molecule, 2-Me-5HT (bottom). The inactive molecule is not able to match all of the points in the pharmacophore in a low-energy conformation.

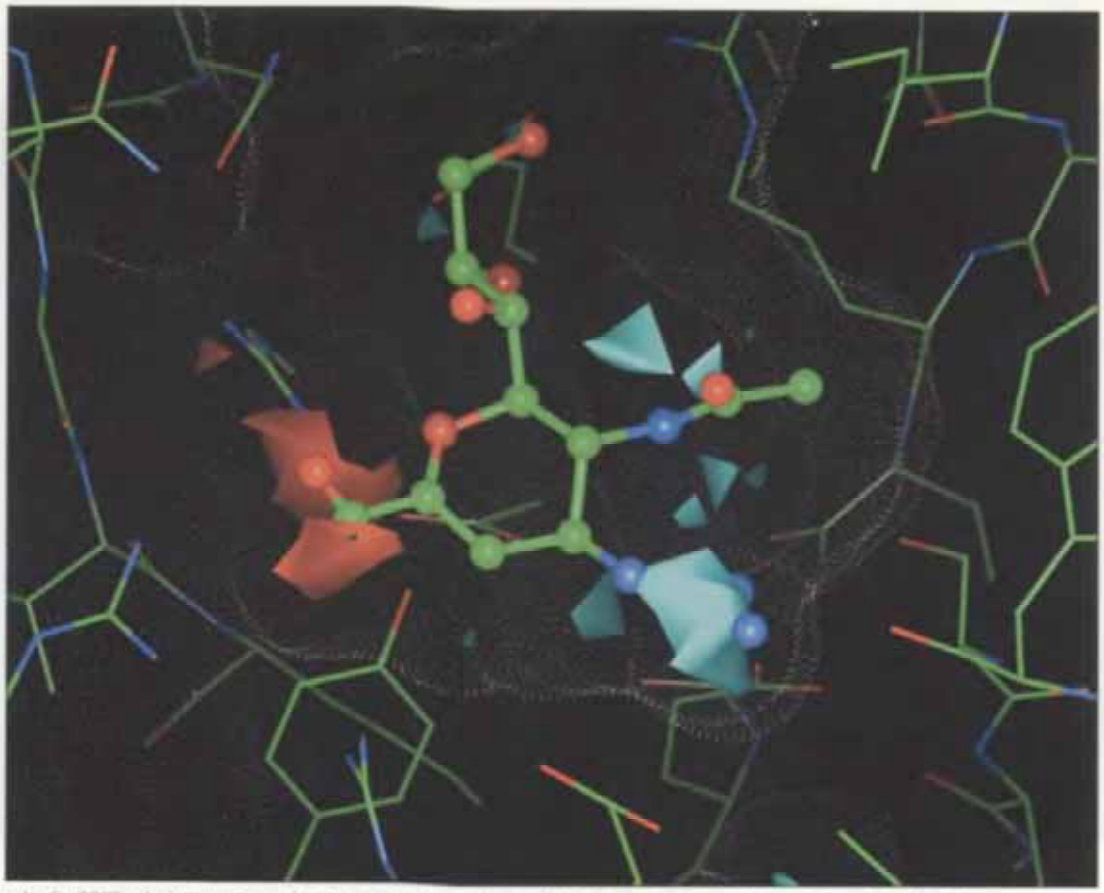


Fig. 12.32. The result of a GRID calculation using carboxylate and amidine probes in the binding site of neuraminidase. The regions of minimum energy are contoured (carboxylate red; amidine blue). Also shown is the inhibitor 4-guanidino-Neu5Ac2en which contains two such functional groups [von Itzstein et al. *1993].*

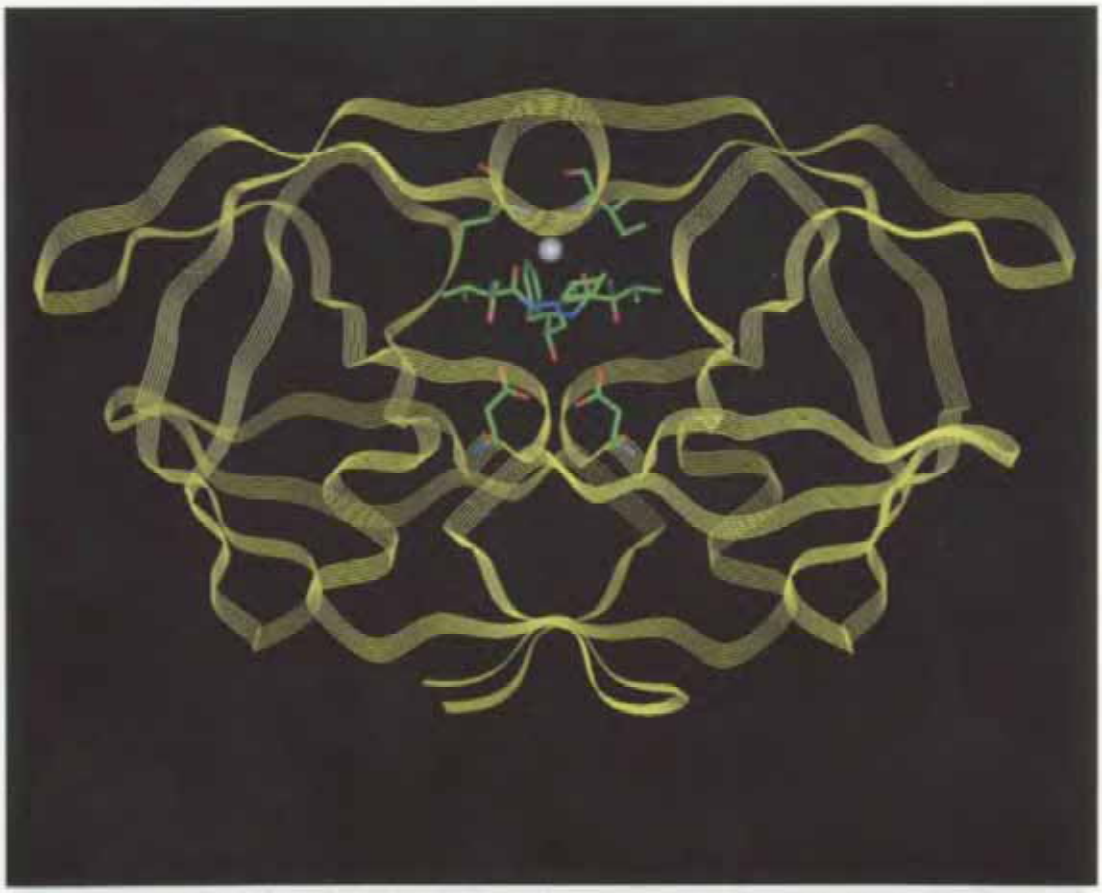


Fig. 12.34. The HIV-1 protease with the inhibitor CGP53820 bound [Priestle et al. 1995]. The water molecule that forms hydrogen bonds both to the inhibitor and to the 'flaps' of the protein is drawn as a white sphere and the catalytic aspartate groups are also represented.

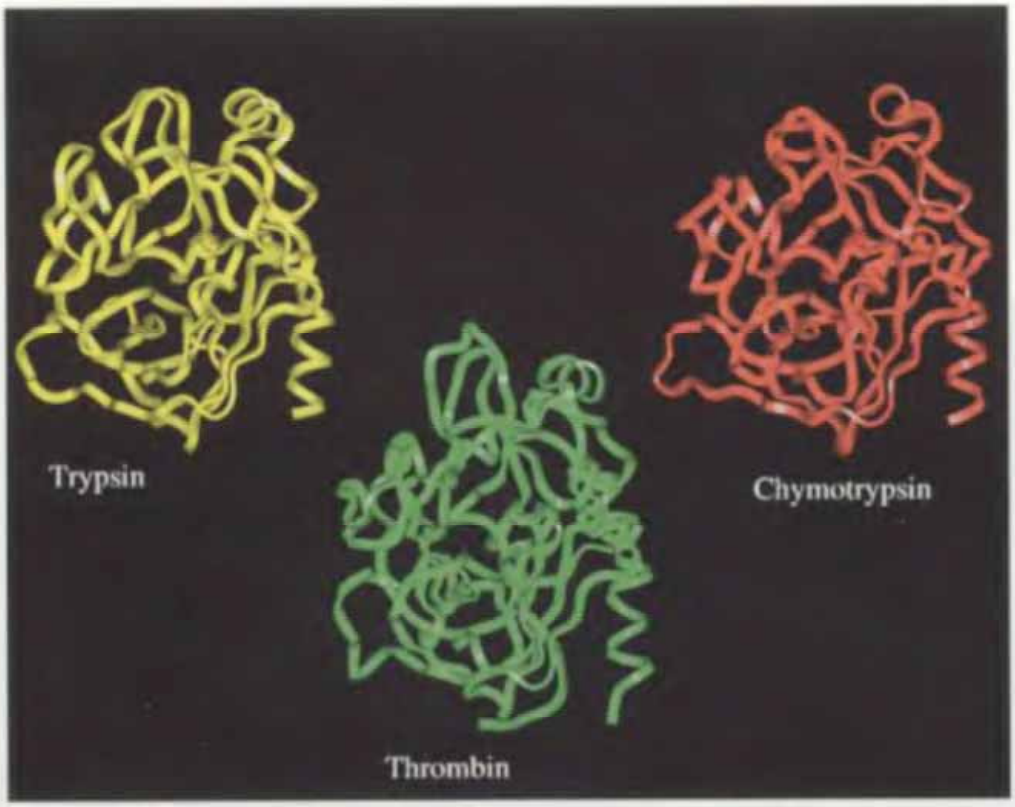


Fig. 10.9: Trypsin (top left) [Turk et al. 1991], chymotrypsin (top right) [Birktoft and Blow 1972] and thrombin (bottom) [Turk et al. 1992] have similar three-dimensional structures.

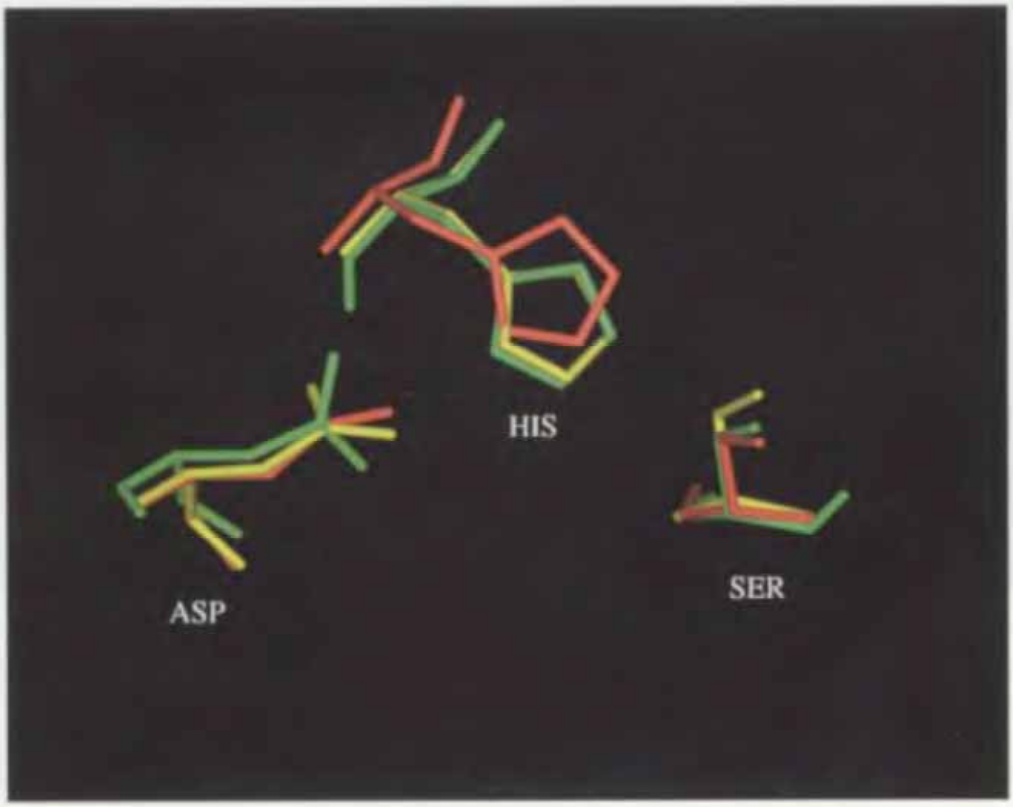


Fig. 10.11: A superposition of the aspartic acid, histidine and serine amino acids in the active sites of trypsin (yellow), chymotrypsin (red) and thrombin (green).

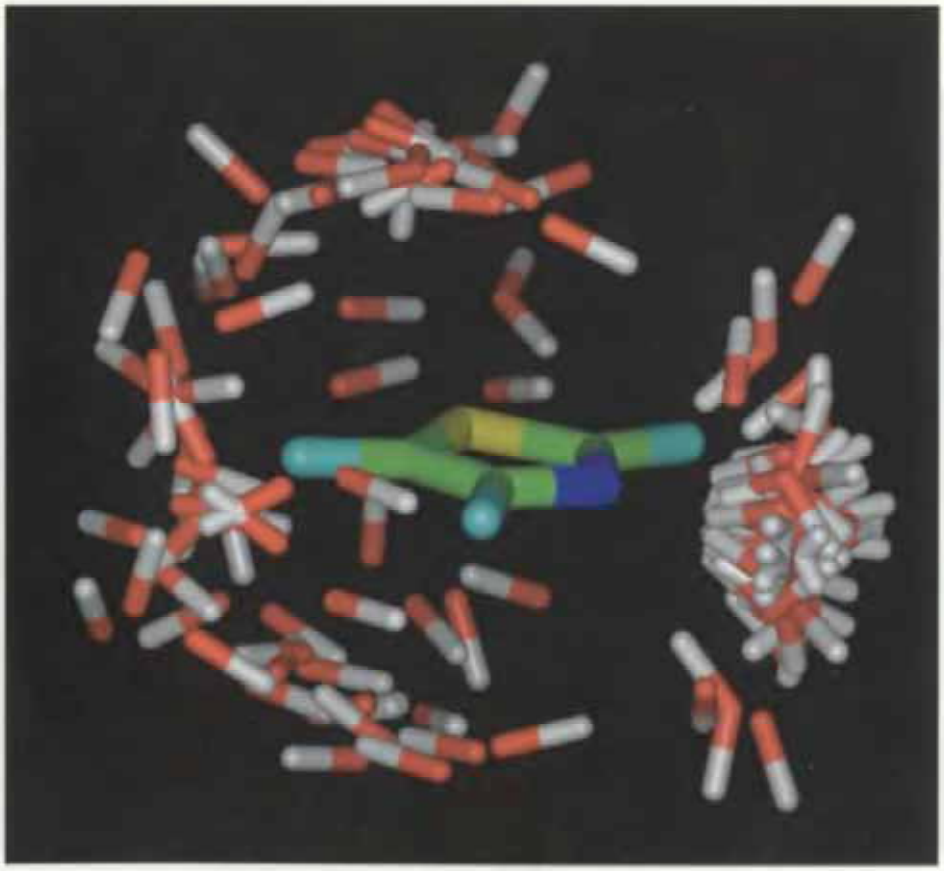


Fig. 9.27: Distribution of hydroxyl groups around thiazole ring systems as extracted from the Cambridge Structural Database [Bruno et al. 1997], illustrating the greater propensity of the nitrogen atom to act as a hydrogen-bond acceptor.

Fig. 2.12: HOMO of formamide. The red contour indicates the negative part of the wavefunction and blue the positive part of the wavefunction. The formamide molecule is oriented with the oxygen atom on the left pointing towards the viewer, as in Fig. 2.11.

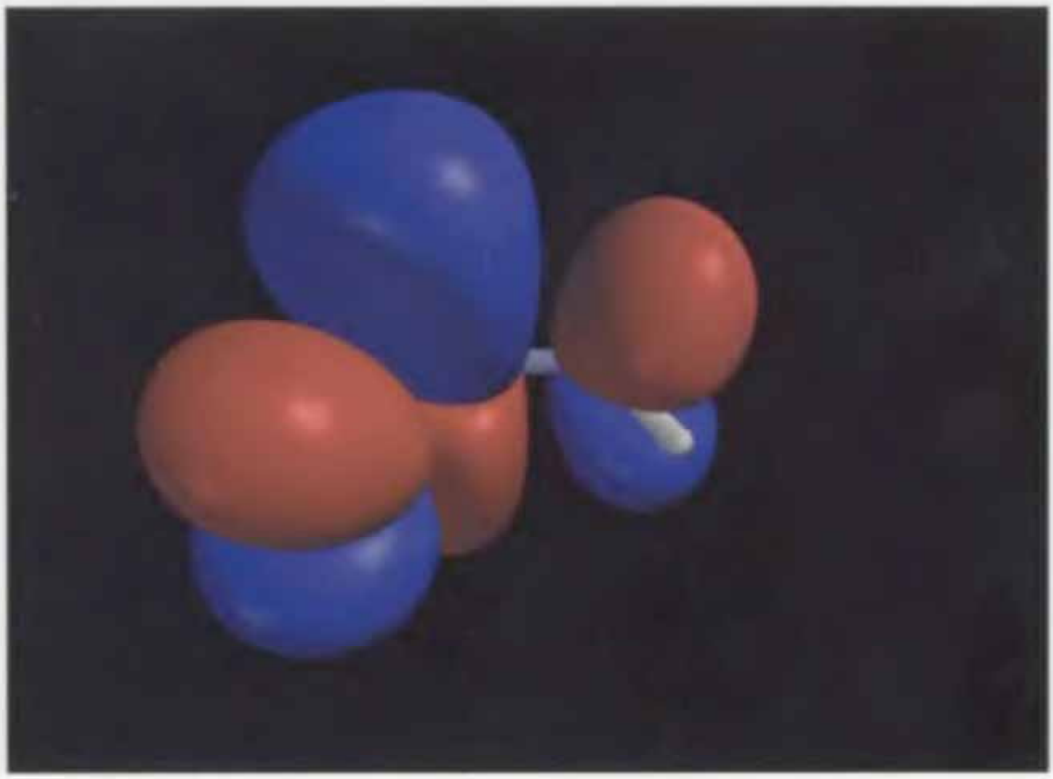


Fig. 2.13: LUMO of formamide.

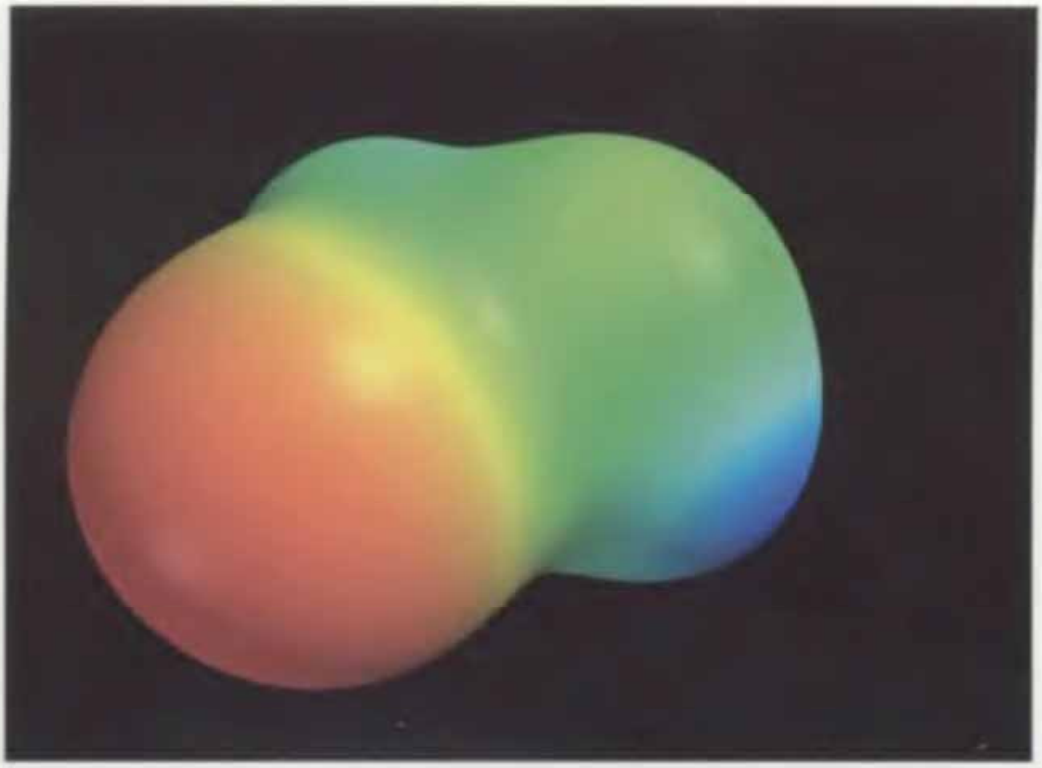


Fig. 2.18: Electrostatic potential mapped onto the electron density surface for formamide. The orientation of the molecule is as in Fig. 2.11. Red indicates negative electrostatic potential and blue is positive potential.

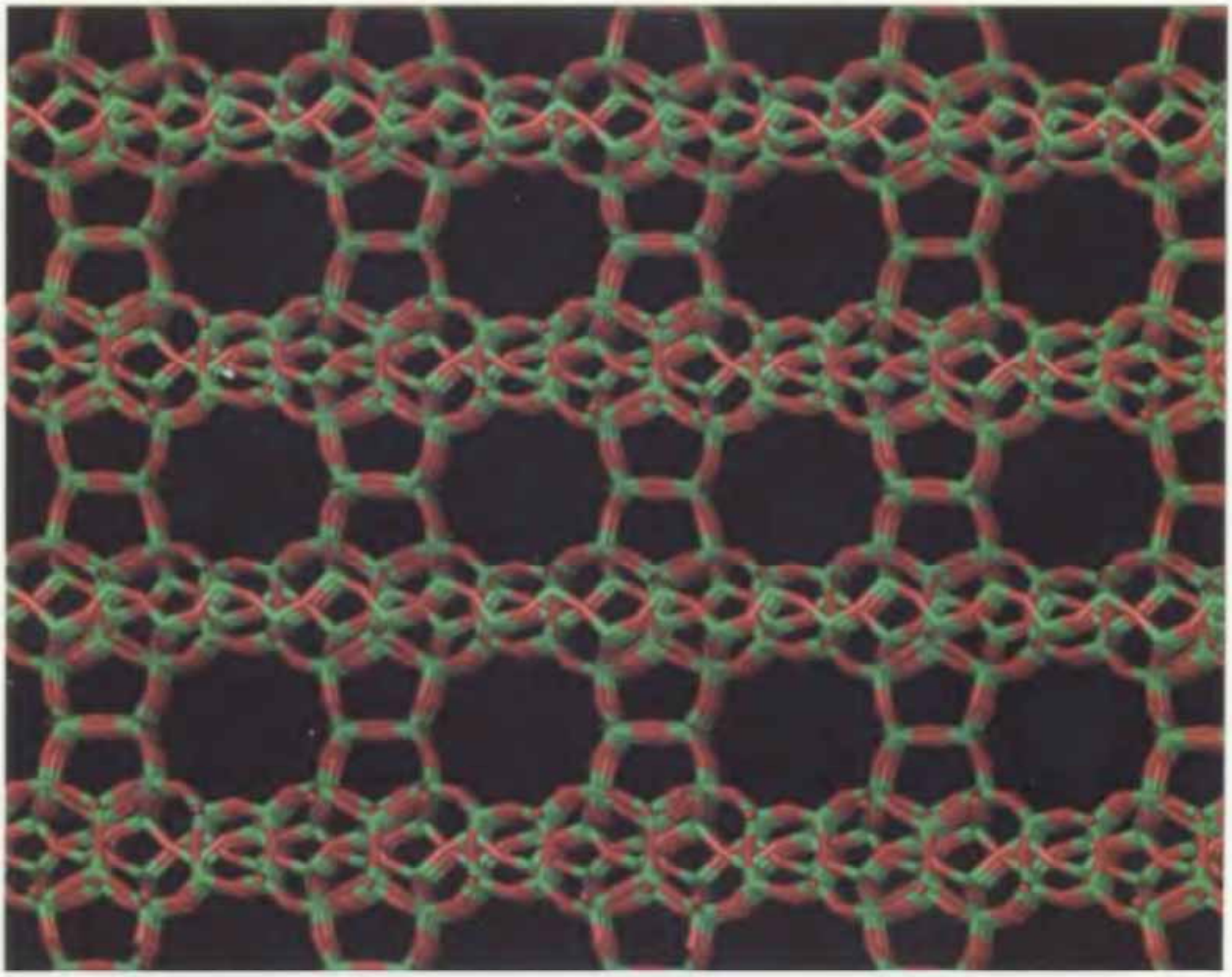


Fig. 5.36: The zeolite NU-87.

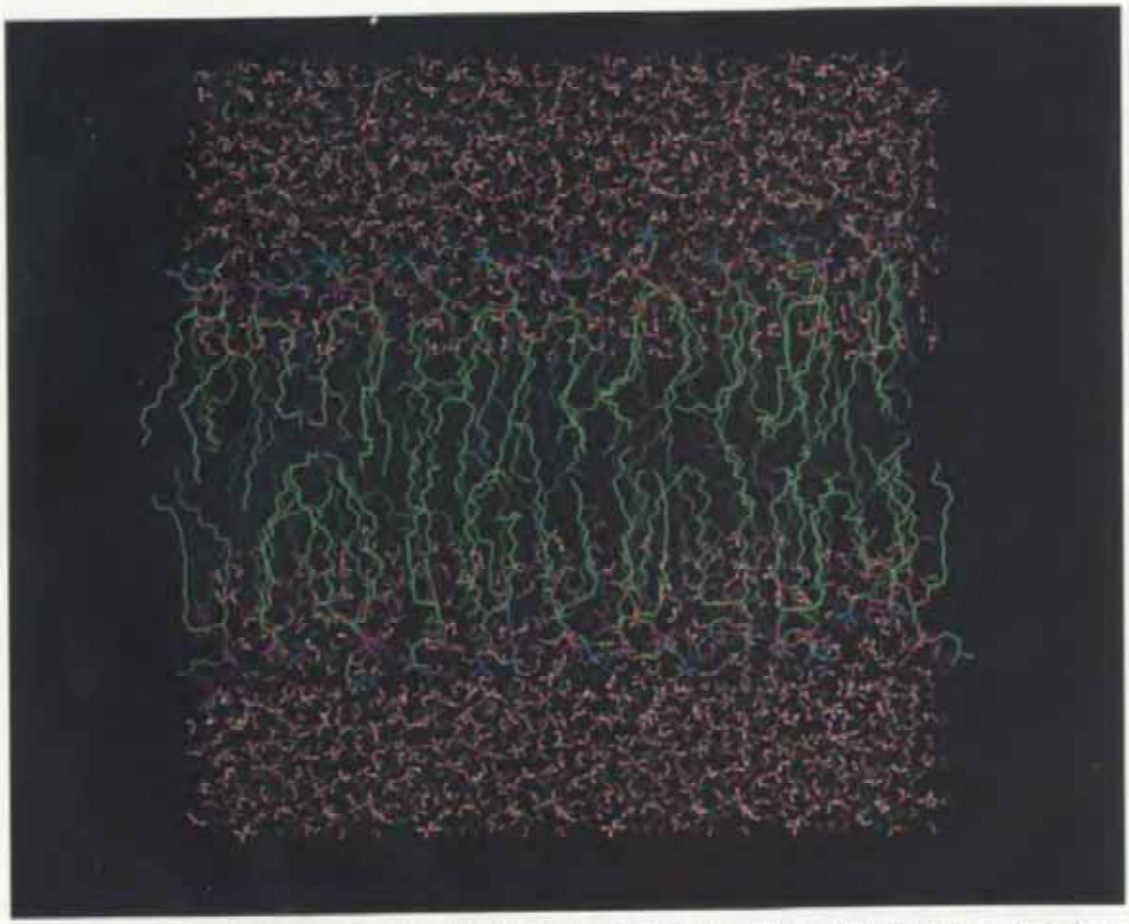


Fig. 7.21: Snapshot from a molecular dynamics simulation of a solvated lipid bilayer [Robinson et al. 1994]. The disorder of the alkyl chains can be clearly seen.

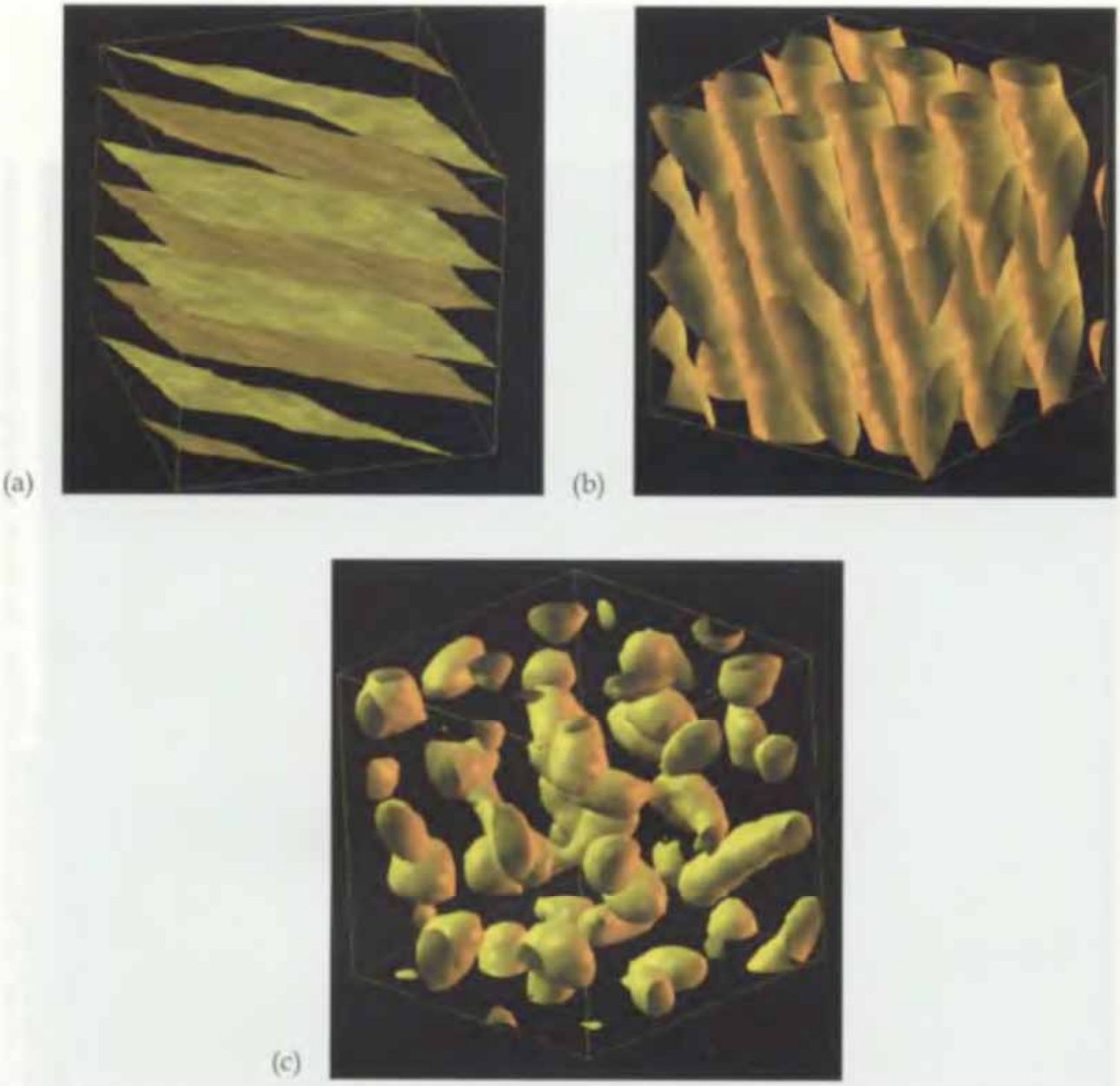


Fig. 7.24: Graphical representation of final configurations obtained from dissipative particle dynamics simulations on block copolymers. (a) shows the lamellar phase obtained for the A_5B_5 system, (b) the hexagonal phase from A_3B_7 and (c) the body-centred-cubic phase obtained for A_2B_8. Figure redrawn from Groot, R. D. and Madden,T. J. 1998. Dynamic simulation of diblock copolymer microphase separation. The Journal of Chemical Physics, 108: 8713–8724.

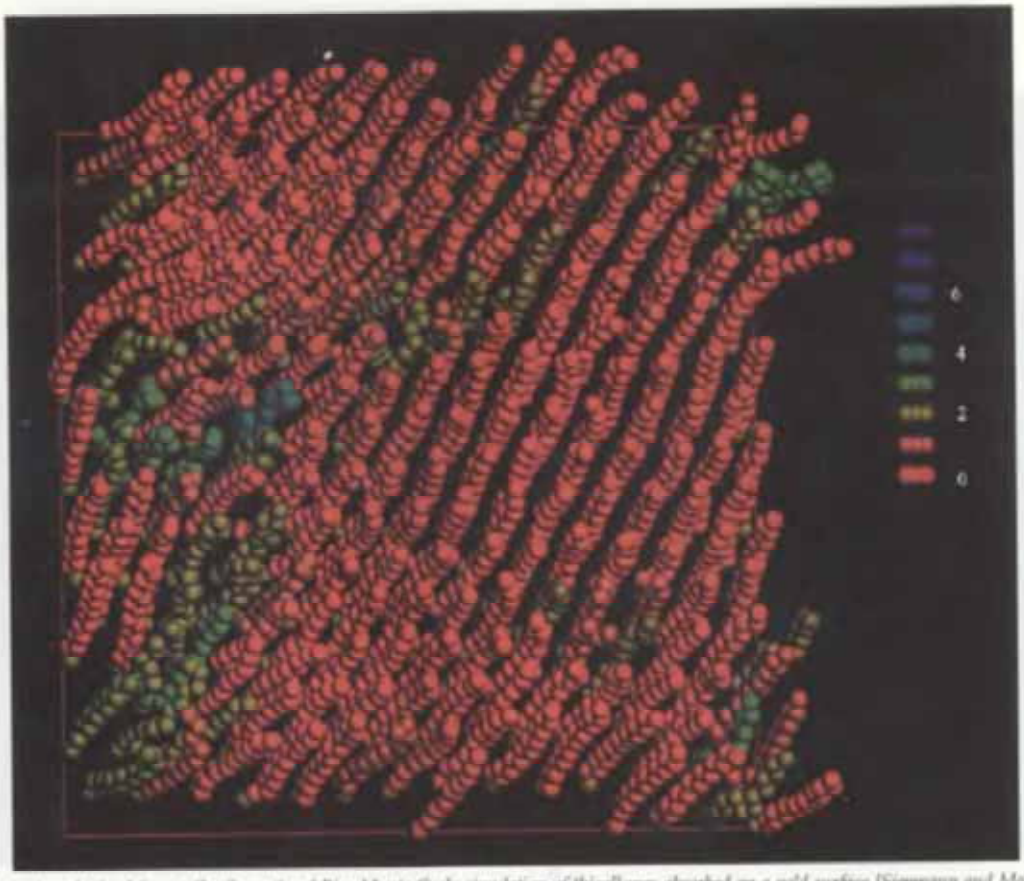


Fig. 8.21: Final configuration obtained from a Configurational Bias Monte Carlo simulation of thioalkanes absorbed on a gold surface [Siepmann and MacDonald 1993a]. The system contains 224 molecules which are colour coded according to the number of gauche defects, with red chains being all trans, yellow chain containing three gauche bonds and green chains containing five gauche bonds.

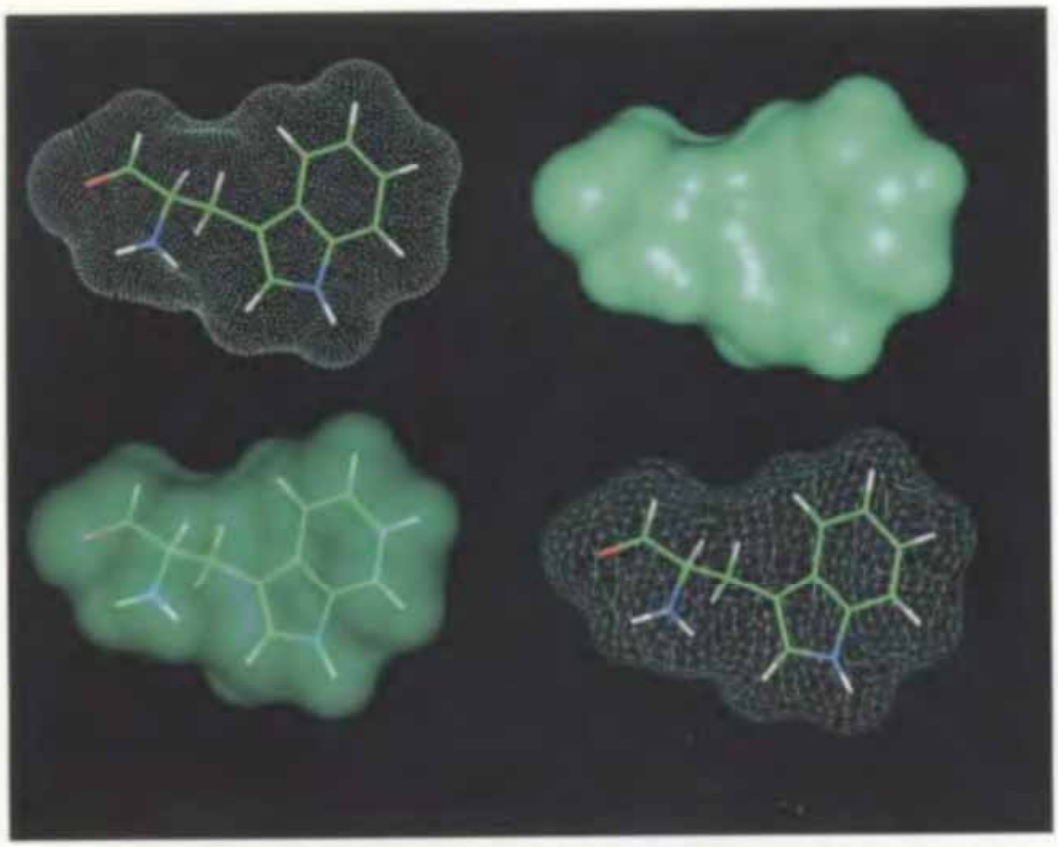


Fig. 1.7: Graphical representations of the molecular surface of tryptophan. Clockwise from top left: dots, opaque solid, mesh, transluscent solid.

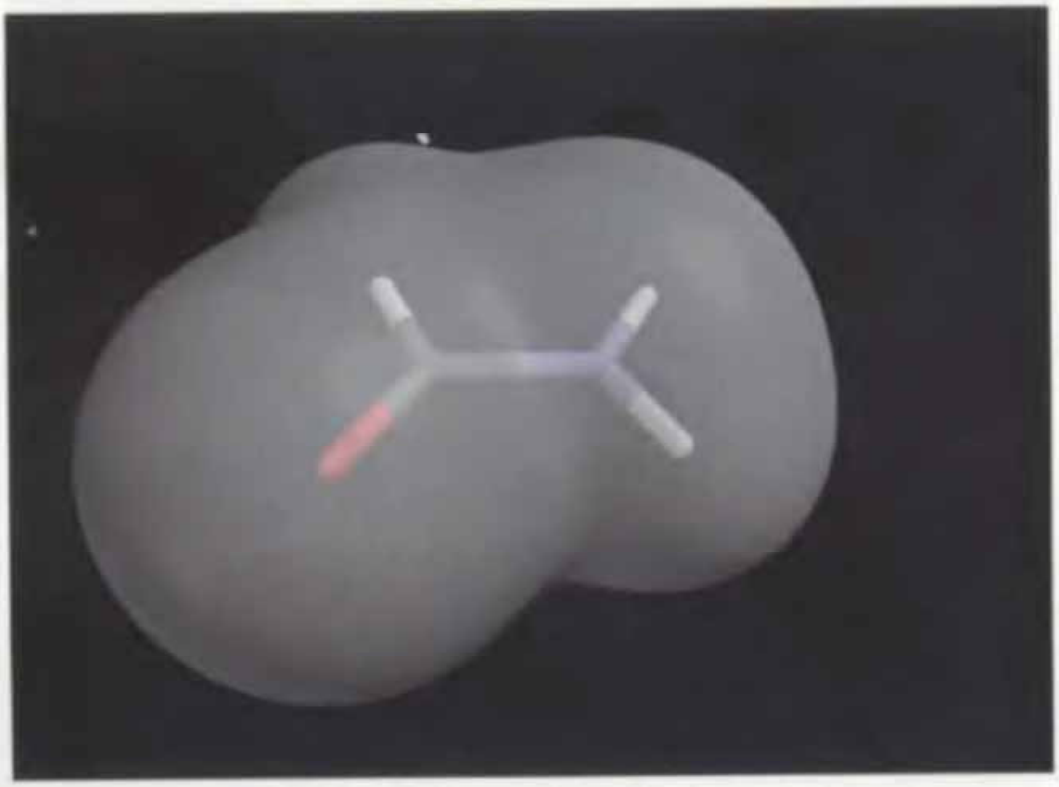


Fig. 2.11: Surface representation of electron density around formamide at a contour of 0.0001 au (electrons/bohr^3).

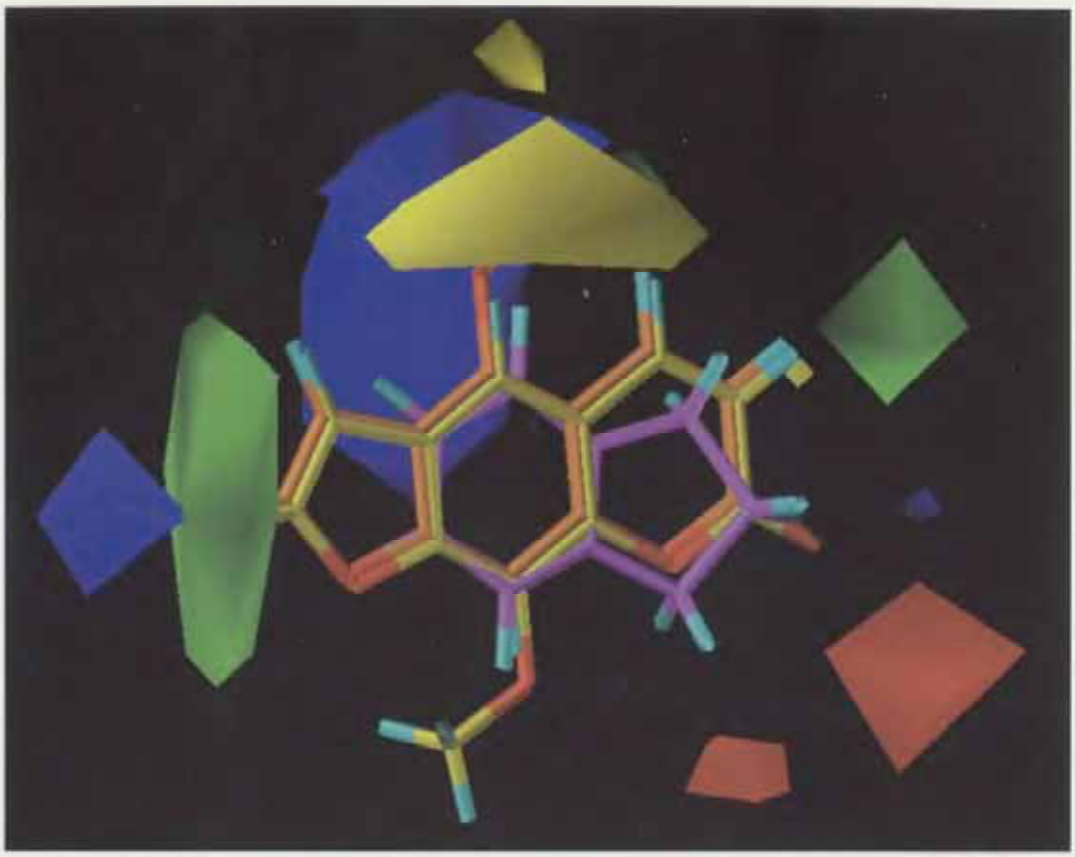


Fig. 12.41: Contour representation of key features from a CoMFA analysis of a series of coumarin substrates and inhibitors of cytochrome P_{450}2A5 [Poso et al. 1995]. The red and blue regions indicate positions where it would be favourable and unfavourable respectively to place a negative charge and the green/yellow regions where it would be favourable/unfavourable to locate steric bulk.