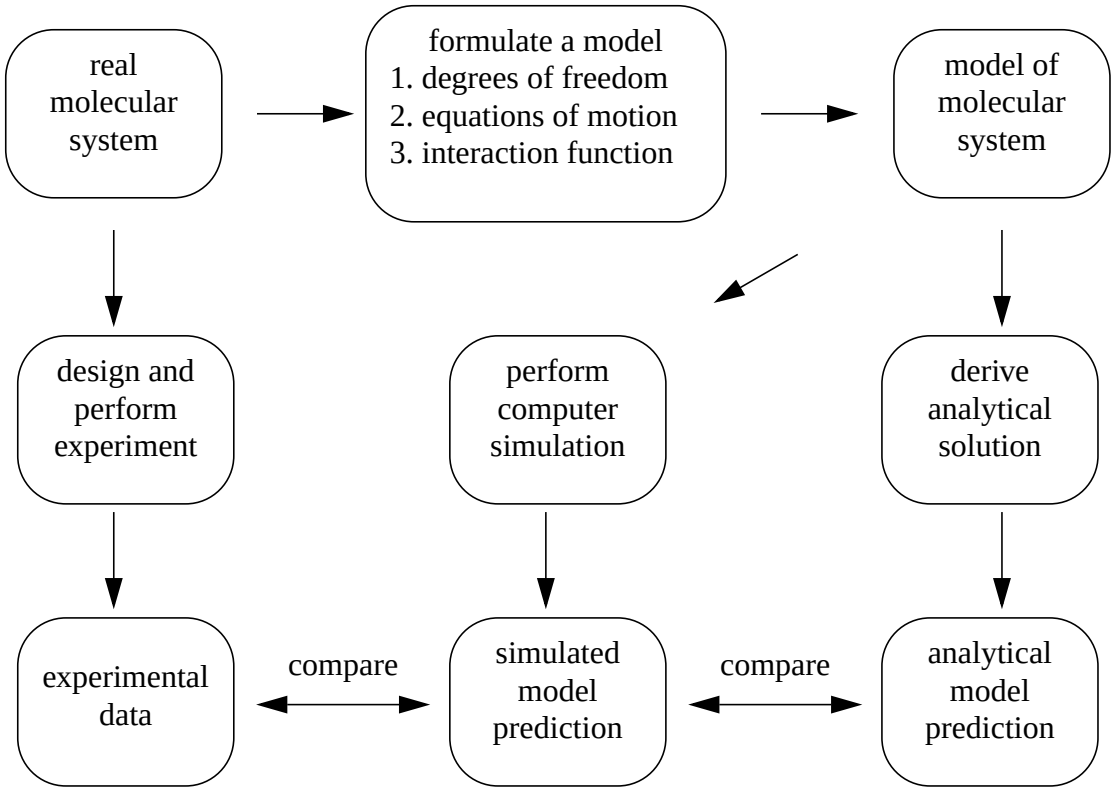
3.5. Conformational Analysis Including Energetic Contributions

The aim of a computer simulation is a good description of a real system by a model.



Since the simulation techniques utilize assumptions and reductions in the degrees of freedom, the result of the calculations has to be critically compared to the real data set.

Simulations are performed to gain insight into the

1. sampling of configuration space;
    - used for refinement of experimental structures from X-ray or NMR;

2. determination of equilibrium averages by weighting each point in configuration space with an appropriate Boltzmann factor;
    - structural and motional properties (e.g. atomic mean square fluctuation amplitudes);
    - thermodynamic properties;

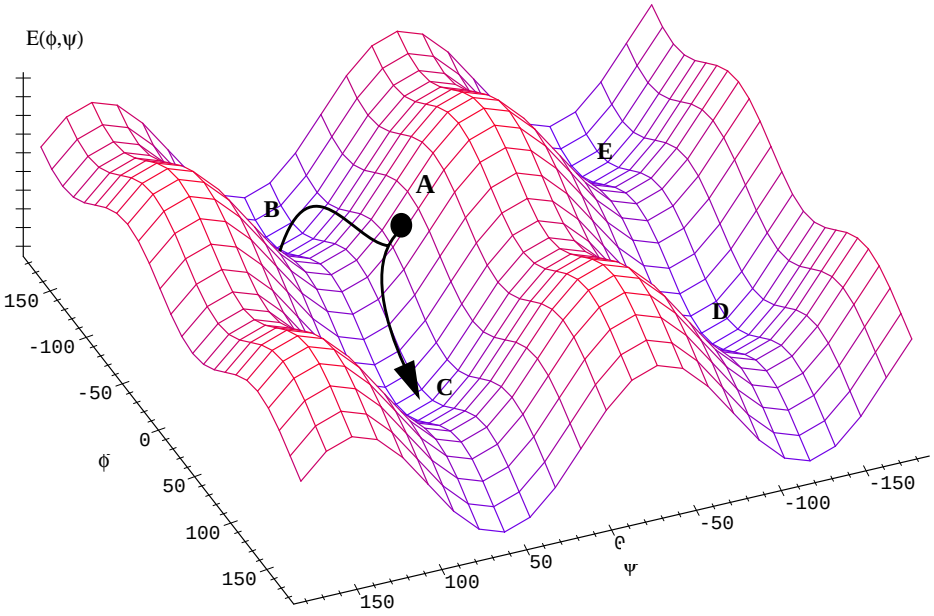
3. examination of dynamic trajectories;
    - time dependent development of system.

Several ***methods for the calculation of molecular properties*** exist, like

- ab initio molecular orbital methods;
- semi-empiricial molecular orbital methods;
- empirical force field methods.

For the size of biomacromolecules only force field methods are calcualable in practice.
Computer simulations of multiple atom systems, e.g. a biomacromolecule, rely on the possibility that the energy of the system can be evaluated as a function of its coordinates. By repeating the evaluation for various coordinate sets the multi-dimensional ***energy surfaces*** (hyperplanes) can be probed.

3D-subplot (torsion dependency) of an energy hyperplane



E(A) > E(B)

E(A) > E(C)

E(C) < E(B)

The exploration of the energy hyperplane may use different approaches:

- Energy Minimization [EM];
- Monte Carlo methods [MC];
- Molecular Dynamics [MD];
- Stochastic Dynamics [SD];
- Normal Mode analysis [NM].

3.5.1. Energy minimization

Any real molecule (see A in the hyperplane) spontaneously tries to occupy a lower energetic state if the pathway for this change is open. At ***equilibrium*** (B or C) the system doesn't change and is in its ***minimum energy state***.
Given a structural model the problem is to find a coordinate set close to this conformation at which the energy surface has a minimum. This means, that a point on the high-dimensional conformational space is obtained where all forces on the atoms are balanced.
We are interested in the point of lowest energy of the hyperplane, also called ***global minimum*** (C). Several points usually exist where the atomic forces are almost zero and which are similar in their energetic state. Thus, on the hyperplane ***local minima*** (B, D, E) are distributed as well as maxima and one point among these minima represents the global minimum.
If the calculations correspond to reality (which, of course, depends on the force field assumptions) the ***minimum-energy conformations*** are expected to exist in nature.

The general approach is to

- select an equation describing the energy term of the system as function of the coordinates;
- select a appropriate starting conformation;
- calculate the total conformational energy;
- modify the independent variables;
- recalculate the total energy;
- detect the direction toward lower energy;
- adjust the independent variables for this direction (line search);
- iterate until a minimum-energy conformation is obtained.

The question of fast convergence is critical in minimization studies. Convergence can be defined by: all derivatives are zero and the second derivatives are positive.

Different algorithms for energy minimization exist to avoid excessive computation times:

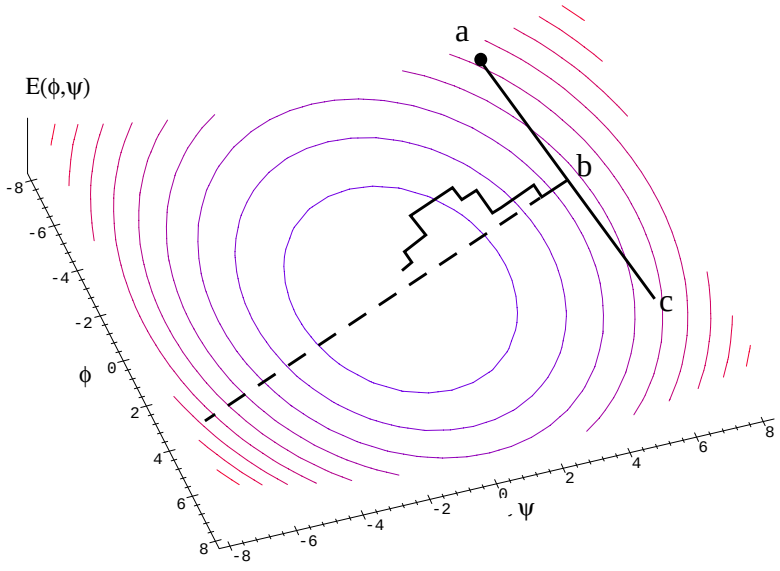
- steepest descent algorithm;
- conjugate gradient method;
- Newton-Raphson approach.

Let our energy function (***force field***) be

$$E(x, y) = x^2 + 2y^2$$

(The contour plot of E in the x,y-planes will then be used to show the different minimizer approaches.)

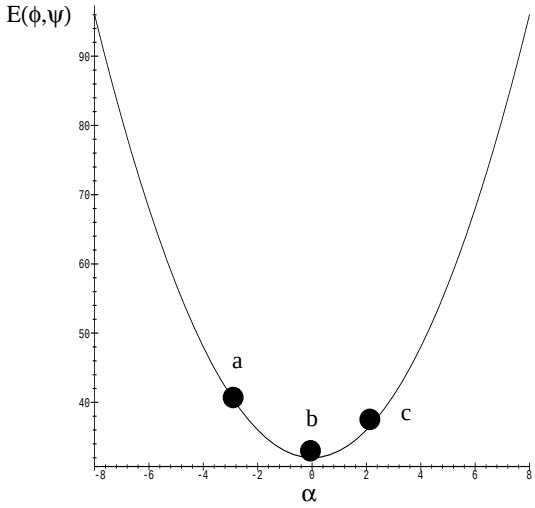
Given an arbitrary starting point in the x,y-plane (a) the minimizer has to determine direction and distance to the minimum, which is in our example at x=0, y=0.



A good initial direction turns out to be the derivative of the function at the current starting point, which in graphical representation give the slope at this point:

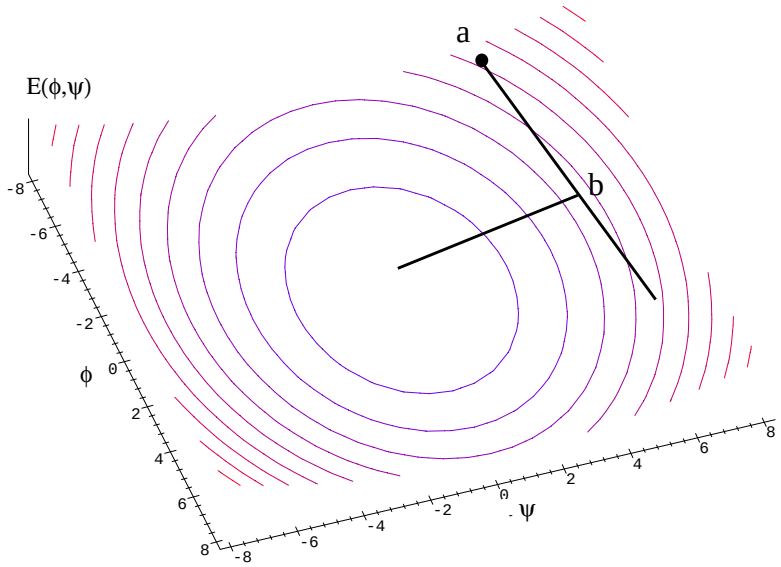
$$\left(\frac{\partial E}{\partial x}, \frac{\partial E}{\partial y}\right) = \nabla E = (2x, 4y)$$

The magnitude of the derivative is a rigorous sign to characterize the convergence of the minimization. Since it points downhill but not in all cases to the minimum, we have to readjust the direction after each step.
The line search is a one-dimensional minimization along a given direction resulting in a change of the coordinates to a new lower energy structure. The derivative at the minimum is perpendicular to the previous direction.



The ***steepest descents*** minimization may use the line search direction as the current gradient. The old direction is replaced by the gradient at the minimum of the first search and the line search is repeated. Since gradients are used to determine the direction, this leads to oscillations on the way downhill and the progress of a previous step is retraced in a later one often by an overcorrection. The gradient is approaching 0 in the surrounding of the minimum, thereby slowing down the convergence. Thus, it shows merits for systems far away from the minimum.

The ***conjugate gradients*** is faster in convergence since it doesn't cancel earlier progress. The minimizer doesn't proceed down the new gradient, but rather in a direction that is constructed to be conjugate to the old gradient and all previous direction vectors. It is computed by adding the gradient at the starting point of the vector to the previous direction scaled with a constant. The minimization proceeds by producing a complete basis set of mutually conjugate directions directly pointing toward the minimum.



This process requires convergence in the line search before a new direction is chosen. Conjugate gradients minimization is used for large systems and in harmonic systems since the Newton-Raphson algorithm requires storage of an additional matrix.

In the ***Newton-Raphson*** algorithm the second derivative matrix is evaluated. By this the curvature of the function is scanned and the point along the gradient is predicted when the function will change the direction (i.e. pass through a minimum). Since the complete second derivative matrix defines the curvature in each gradient direction, multiplication of this matrix by the gradient results in a vector which translates the system directly to the minimum.

$$r_{min} = r_0 - E''^{-1}_0 \cdot \nabla V(r_0)$$

(r_{min} = minimum, r_0 = arbitrary starting point, E''_0 = 2nd derivative of energy with respect to coordinates, ∇V = gradient of the potential energy at r_0)

The algorithm may compute a large step when the forces are large and the curvature is small (e.g. steep repulsive wall of van-der-Waals potential) and thereby overshoot the minimum. This can lead to points further away from the minimum than the starting point resulting in a total divergence. While it is very slow in multi-dimensional space (anharmonic energy surface) but converges rapidly in the harmonic area near the minimum it is often employed when conformations are very close to the minimum. Also the dimensions of the Hessian matrix for a 10.000 atom system (= $3N^2$ or 300.000.000 words) require an extended computer memory (1.2 GB at single precision).

Thus, for a biomacromolecule an acceptable minimization approach consists of a number (~100) of steepest descent steps followed by conjugate gradient minimization (100 - 500).

Molecular dynamics may also provide a way to minimize a system. The integration algorithm simulates a temperature bath and adjusts the average kinetic energy of the atoms to maintain a given temperature. By setting this temperature to a low value (1 K) or reducing it gradually (***dynamic quenching***) the kinetic and potential energy of the molecules is reduced. The procedure, in contrast to the related steepest descents, may overcome small energy barriers during the relaxation phase and reach lower energy minima.

3.5.2. Monte Carlo method

None of the common EM algorithm is able to pass from one local minimum to another one over an intervening barrier in order to fold the molecule to the global minimum.
The Metropolis Monte Carlo method, mainly applicable to small molecules, samples the conformational space using a Boltzmann factor as weighting function.

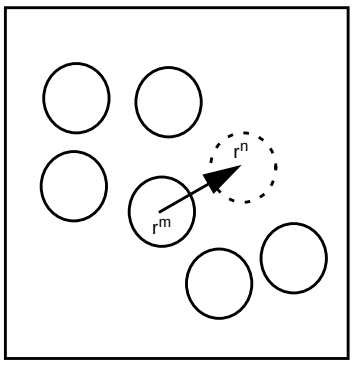
$$W(E) \sim \exp\left(\frac{-E}{k_B T}\right)$$

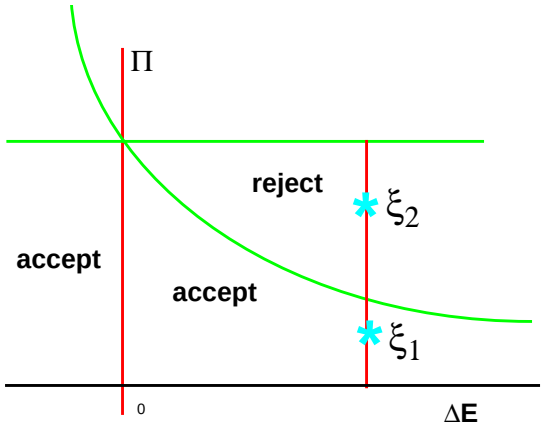
It assumes that an equilibrated system at temperature T has its energy distributed among all different energy states E. Thus, even at very low temperatures a small probability W exists to find a system in a high energy state. The approach includes the thermodynamic assumption that any local minimum is accessible from any other local minimum by a finite number of random sampling steps.

State n is produced from state m by displacement of atom i. The change in the potential energy is calculated by computing the energy of atom i with all the other atoms before and after displacement.The Metropolis test is defined by

$$\frac{p_n}{p_m} = \exp\left(\frac{-E_n + E_m}{k_B T}\right) = \Pi$$

If $E_n < E_m$ it follows $\Pi > 1$ specifying, that the system makes a downhill move and the new configuration is accepted.
If $E_n > E_m$ it follows $\Pi < 1$, then the detected local minimum is examined by the ***Metropolis criterion***, i.e. a random number ξ between 0 and 1 is generated and compared. Only those structures where ξ is less then Π are accepted.





Acceptance of up-hill moves in MC

For values of Π lower 1 the uphill direction sometimes occurs. The random sampling assures that during the course of the run the net result is that energy changes of ΔE are accepted with a probability of Π.

3.5.3.1. Molecular dynamics simulations (MD)

A full quantum mechanical calculation of molecular structure and dynamics on the basis of the Schrödinger equation is not feasible for large biomacromolecules. An working assumption is the Born-Oppenheimer separation of electrons and nuclei: the electrons provide an average potential field for motions of the nuclei and the electrons adjust instantaneously to nuclear motions. Therefore, the molecule is treated classically as particles (atoms) moving in a potential field. The equations of motion of classical mechanics are generally in Newton´s formulation in cartesian coordinates.

By integration of ***Newton's equations of motion***s for the solute and solvent a number of configurations with time dependence (***trajectory***) are generated.

$$m_i \cdot \frac{d^2 \boldsymbol{r}_i}{dt^2} = m_i \cdot \frac{d\boldsymbol{v}_i}{dt} = -\boldsymbol{F}_i(r_1, r_2, \ldots, r_N, t)$$

(Here, m_i is the mass and r_i the position of an atom i with its coordinates x_i, y_i, z_i in a molecule with N atoms. F is the potential energy defined from the energy surface at the current atom positions.)

The first derivative of a position with respect to time gives the velocity of the atom, therefore, starting a biopolymer MD simulation needs

- an initial structural model, obtained by NMR/DG calculations, X-ray crystallography or pure modelling;
- a solvent box, if a simulation in solution is desired;
- the assignment of velocities for each atom.

Velocities are generated from a Maxwellian distribution at a temperature below the desired virtual temperature of the simulation. The temperature T(t) is defined in terms of a mean kinetic energy contribution

$$T(t) = \frac{1}{(3N-n)k_B} \sum_{i=1}^{N} m_i \cdot \left|\boldsymbol{v}_i\right|^2$$

depending on the total number of unrestrained degrees of freedom in the system (3N-n), the velocity of the atoms v_i.

While volume and total energy of the system are kept constant T and the pressure p may vary in small ranges. In order to compensate an increase in T or p produced by small distortions of the molecule geometry, these two variables are coupled to a bath. The equations are modified for relaxation of 1. order (relaxation constant τ about 10 steps)

$$\frac{dT}{dt_{bath}} = \frac{(T_0 - T)}{\tau_T}$$

This allows to minimize local perturbations but conserves global effects since the velocity is multiplied in each step with a factor λ and the coordinates by an appropriate term representing the

pressure.

$$v_i \leftarrow \lambda \cdot v_i$$

$$= \sqrt{1 + \left(\frac{T_0}{T - 1} \cdot \frac{\Delta T}{\tau_T} \right)} \cdot v_i$$

$$T = \frac{\sum_{i=1}^{N} \frac{1}{2} \cdot m_i \boldsymbol{v}_i^2}{\left(\frac{3Nk}{2} \right)} \qquad <= \ E_{kin}$$

For a simulation in solution, the atom positions of the system are generated by fitting the solute into a preequilibrated box of solvent molecules. Of course, this step is omitted for ***in vacuo*** simulations.

Various forms of potentials depending on application may be defined:
- fluids: pairwise interactions depending on interparticle distances;
- crystals: displacements from average lattice positions;
- macromolecules: forces related to covalent structure.

A ***force field*** is a set of equations and parameters used by the program which together with a given set of coordinates yields the energy of the system.

A standard protein force field has e.g. the following definition

$F(r_1, r_2, \ldots, r_N) = \sum \frac{1}{2} K_b (b - b_0)^2$ covalent bond interactions

$+ \sum \frac{1}{2} K_\theta (\theta - \theta_0)^2$ bond angles

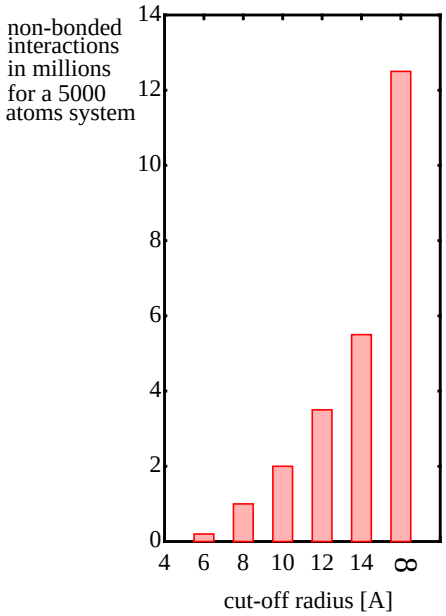
$+ \sum \frac{1}{2} K_\xi (\xi - \xi_0)^2$ (harmonic term for torsions without transitions)

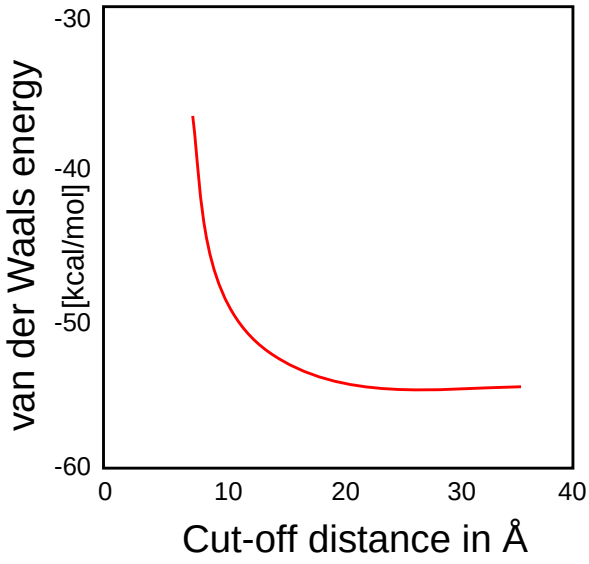
$+ \sum K_\Phi (1 + \cos(n\Phi - \delta))$ sinoidal term for torsions with transitions

$+ \sum_{i,j} \left(\frac{C_{12}}{d_{ij}^{12}} - \frac{C_6}{d_{ij}^6} \right)$ non-bonded interactions (van der Waals contr.)

$+ \sum_{i,j} \frac{q_i q_j}{4\pi\varepsilon_0 \varepsilon_d d_{ij}}$ non-bonded interaction (Coulomb contr.)

The common form for bond stretching and valence angle bending are quadratic terms derived by neglecting higher order terms in an expansion of the energy function. Sometimes a Morse potential is applied for the bond strain evaluation.
A typical form for non-bonded van der Waals interactions is the Lennard-Jones 6-12 potential whereas electrostatic contributions are often taken as a Coulombic interaction between point charges. Since the Coulombic interaction describes the ***long-range forces*** (which is an extensive computational problem) the evaluation of this term is often restricted by introduction of a ***cut-off radius***. Within this cut-off (e.g. 0.6 - 0.9 nm) all non-bonded interactions are treated and stored in a neighbour list. This list is updated every 10 - 20 calculation steps.





Note, that the time to compute the energy of the system is approximately proportional to the number of non-bonded interactions.

Further terms may be introduced to mimick a

- ***hydrogen bonding potential***,
- to include off-diagonal interactions of internal coordinates, (cross-terms, like $\Sigma\Sigma\ K_{bb'}[b-b_0][b'-b'_0]$ or $\Sigma\Sigma\ K_{b\theta}[b-b_0][\theta-\theta_0]$)
- to restrain the out-of-plane motions of aromatics,
- to consider experimental data (NOE, J) as ***restraining potentials***

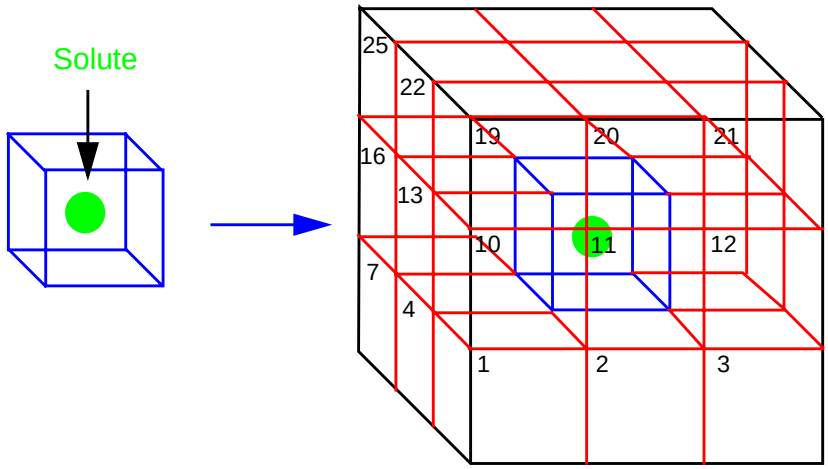
$$\begin{aligned} \ldots + K_{dc}(d_{ij} - u_{ij})^2 \quad & \text{if } d_{ij} > u_{ij} \\ + K_{dc}(l_{ij} - d_{ij})^2 \quad & \text{if } d_{ij} < l_{ij} \\ + 0 \quad & \text{otherwise} \end{aligned}$$

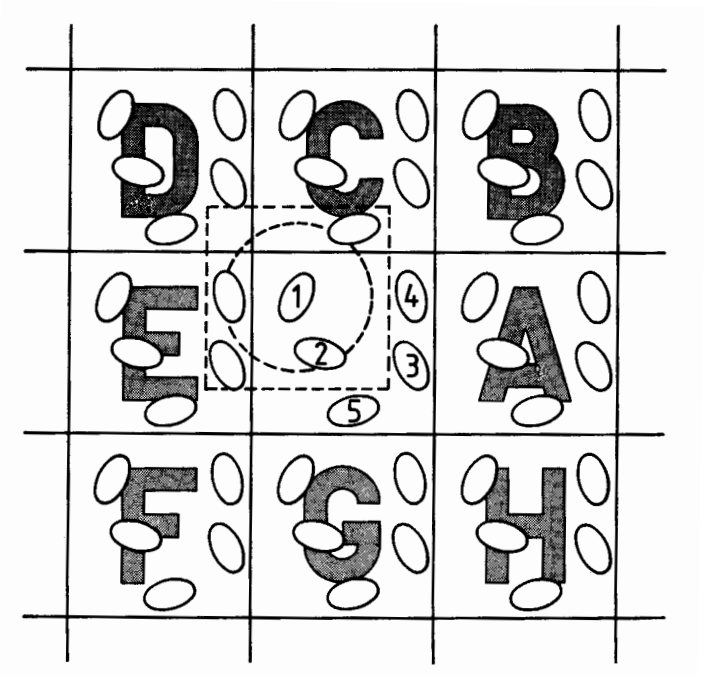
Some MD programs provide beside the ***all-atom force field*** the possibility to reduce the number of atoms to be considered in the integration

- by incorporation of the non-polar hydrogens into the common heavy atom (CH, CH_2 and CH_3). Mass and radii of the ***united atom*** are increased according to the number of incorporated hydrogens. Exchangeable/polar hydrogens are treated explicitly.
- by representing a whole biopolymer residue in terms of a ***united residue*** through a single

pseudo-atom.

Especially for *in vacuo* simulations the behaviour of atoms located near the protein surface is found to be distorted. This is an artifact arising from the fact, that the vacuum around the molecule doesn't compare to a real system. In order to treat all atoms within the simulated system equally ***periodic boundary conditions*** are introduced. All atoms of the system are considered to be in a cube, truncated octahedron or any other periodically spacefilling shaped box. This geometric element is copied in every dimension so that the initial cube is surrounded in every dimension by 2 other cubes leading to 26 + 1 cells, every an exact reproduction. Since a calculation of forces for any atom will consider all atoms within the cut-off radius, this assures (with $R_{box} > 2\ R_{cutoff}$) that every atom will experience the full contribution of its nearest neighbour set. In principle, an *in vacuo* simulation corresponds then to a simulation of a crystal. For a protein in solution no disturbing effects are introduced by periodic conditions if the box size is big enough. The periodic boundary conditions thereby allow atoms near the surface to interact with solvent molecules in another cell.





from:
M.P. Allen & D.J. Tildesley "Computer simulation of liquids", Oxford University Press, 1987

The minimum image model assures that only interactions to the closest molecular images are

evaluated (dashed box = X). Thus, the closest image of 4 to X1 is not X4 but E4 which is -hardly- within the cut-off radius (dashed circle).
From the different methods to integrate the differential equations two should be mentioned:

- The ***predictor-corrector algorithm*** (Gear) has the following scheme which is repeated every MD step
  a. predict the positions, velocities and accelerations (second derivatives) at time t+δt using the current values;
  b. evaluate the forces and accelerations from the predicted positions;
  c. correct the predicted positions, velocities and accelerations using the new acceleration;
  d. calculate the energy or other parameters and return to a.

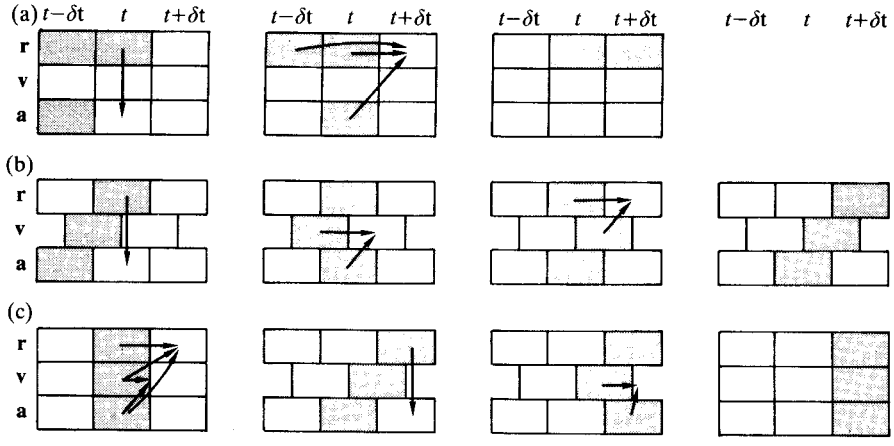
- In the leap-frog algorithm (Verlet) the velocities are calculated with a time-shift of δt/2 against the positions and accelerations:

$$r(t+\delta t) = r(t) + \delta t \cdot v\left(t + \frac{1}{2}\delta t\right)$$

$$v\left(t + \frac{1}{2}\delta t\right) = v\left(t - \frac{1}{2}\delta t\right) + \delta t \cdot a(t)$$

  a. From the current positions the accelerations are calculated,
  b. current acceleration and previous velocity are combined to new velocities,
  c. positions and new velocities give the new coordinates,
  d. calculate the energy or other parameters and return to a.

Illustration of the leap-frog scheme (b) which got its name from the fact that the velocities leap over the coordinates to give the next mid-step. In this figure also other algorithms are shown to solve the equations of motion.



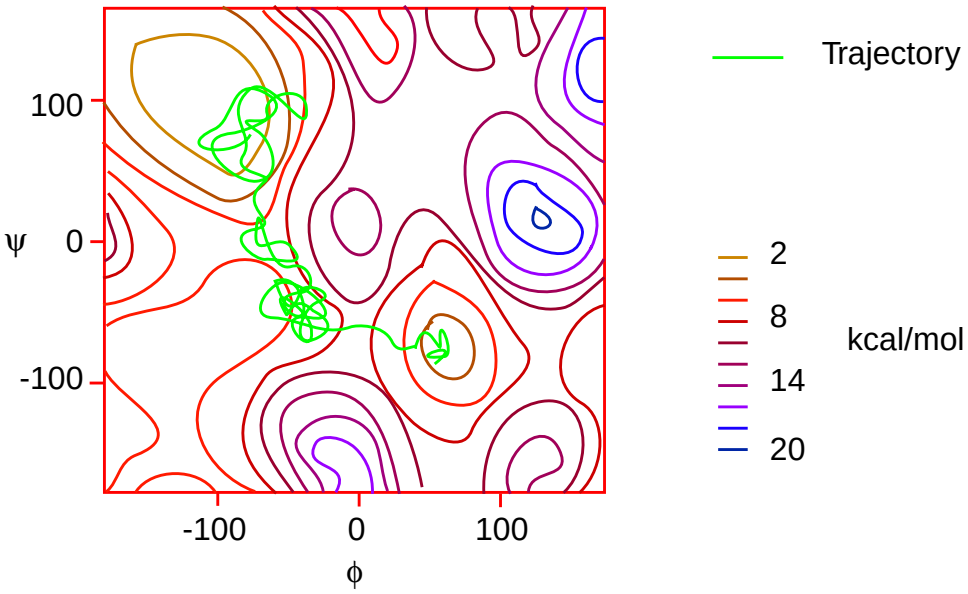
Various forms of the Verlet algorithm. (a) Verlet's original method. (b) The leap-frog form. (c) The velocity form. We show successive steps in the implementation of each algorithm. In each case, the stored variables are in grey boxes.

from: M.P. Allen & D.J. Tildesley
"Computer simulation of liquids", Oxford University Press, 1987

Recipe for a MD simulation:

1. Generate a reasonable starting structure.
2. Perform 100 steps steepest descents energy minimization.
3. Perform a conjugate gradient energy minimization until the derivatives are less 0.001 kcal mole^{-1}.
4. Adjust the periodic box and -for a simulation in solution- add solvent molecules.
5. Initialize the dynamics run by random velocities assigned consistent to a Maxwell-Boltzmann distribution for the target temperature.
6. Calculate with a time step of 0.1 fs or less to prevent that the forces on atoms do not change within the used time step.
7. Analyze the trajectory.

Every MD simulation starts with an ***equilibration period*** (of 10 to 50 ps) in which the temperature and pressure of the system is relaxed. This process decreases the probability that localized fluctuations in the energy will persist throughout the whole simualtion.
Once the properties of the system are stable which can be estimated e.g. by a constant average kinetic energy, the trajectory is calculated for an extended period adequate for the goals of analysis (1 ns). The figure shows a trajectory of one residue in a biopolymer superimposed to an energy contour map.



The calculated energy of a fully minimized system gives the enthalpy at absolute zero. The entropy is neglected in the MD simulations.
For all forcefields the zero-energy is an arbitrary point. Thus, comparison of potential energies between different systems within the same force field or the same system in different force fields is not possible. The validation depends critically on the data used for parametrization of the force field.

A comparison of the CHARMM, AMBER and ECEPP force fields (from: J.Biomol Struct.&Dyn. (1989) 7, 421) for N-acetyl alanine N´-methyl amide is shown in the figures below.

CHARMM

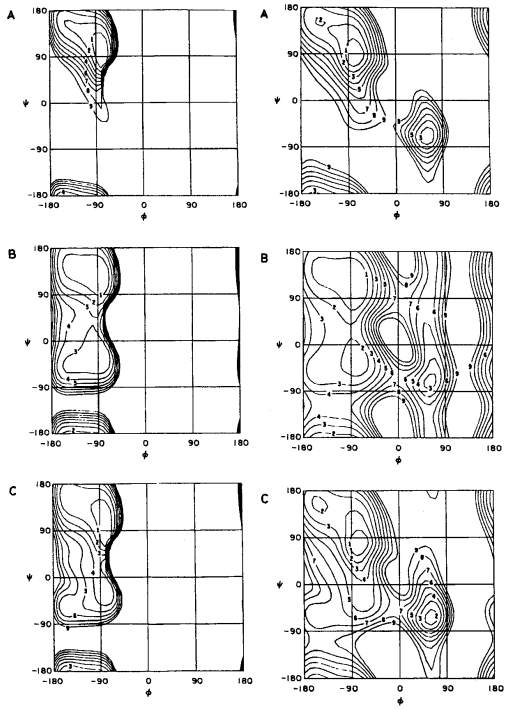


Figure 1: ϕ-ψ map of N-acetyl alanine N′-methyl amide using CHARMM potential, without adiabatic relaxation. A, ε = 1.0; B, ε = 4.0; C, distance-dependent dielectric constant. Contours are drawn at intervals of 1.0 kcal/mol.

Figure 2: ϕ-ψ map of N-acetyl alanine N′-methyl amide using CHARMM potential, after adiabatic relaxation. A, ε = 1.0; B, ε = 4.0; C, distance-dependent dielectric constant. Contours are drawn at intervals of 1.0 kcal/mol.

AMBER

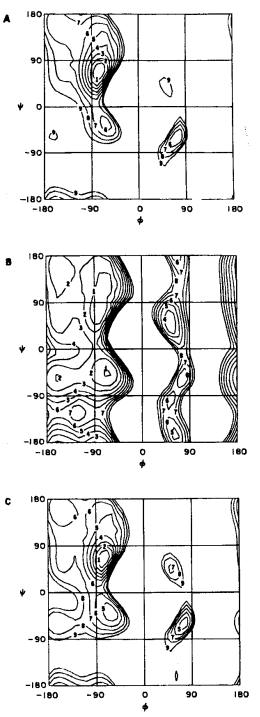


Figure 5: ϕ-ψ map of N-acetyl alanine N′-methyl amide using the AMBER potential, without adiabatic relaxation. A, ε = 1.0; B, ε = 4.0; C, distance dependent dielectric constant. Contours are drawn at intervals of 1.0 kcal/mol.

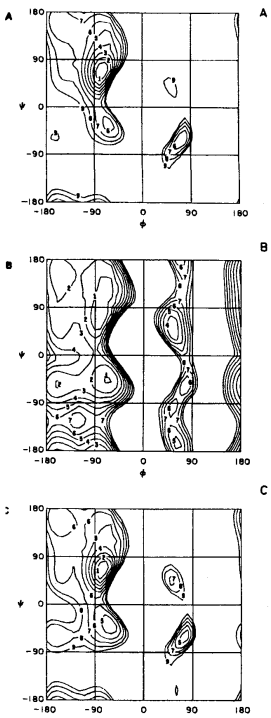


Figure 6: ϕ-ψ map of N-acetyl alanine N′-methyl amide using the AMBER potential, after adiabatic relaxation with pseudorigid geometry. A, ε = 1.0; B, ε = 4.0; C, distance-dependent dielectric constant. Contours are drawn at intervals of 1.0 kcal/mol.

ECEPP

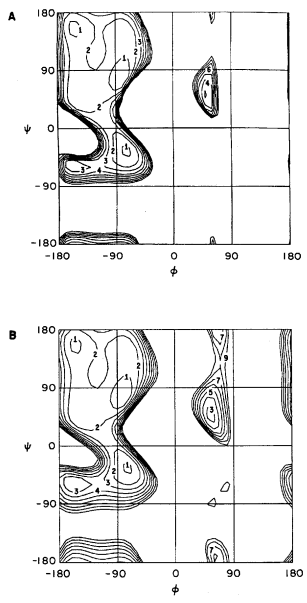


Figure 10: ϕ-ψmap of N-acetyl alanine N′-methyl amide using the ECEPP/2 potential. A, without adiabatic relaxation; B, after adiabatic relaxation. Contours are drawn at intervals of 1.0 kcal/mol.

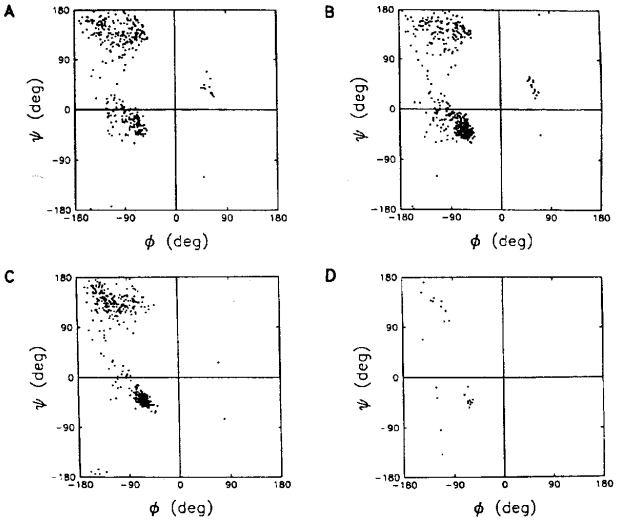
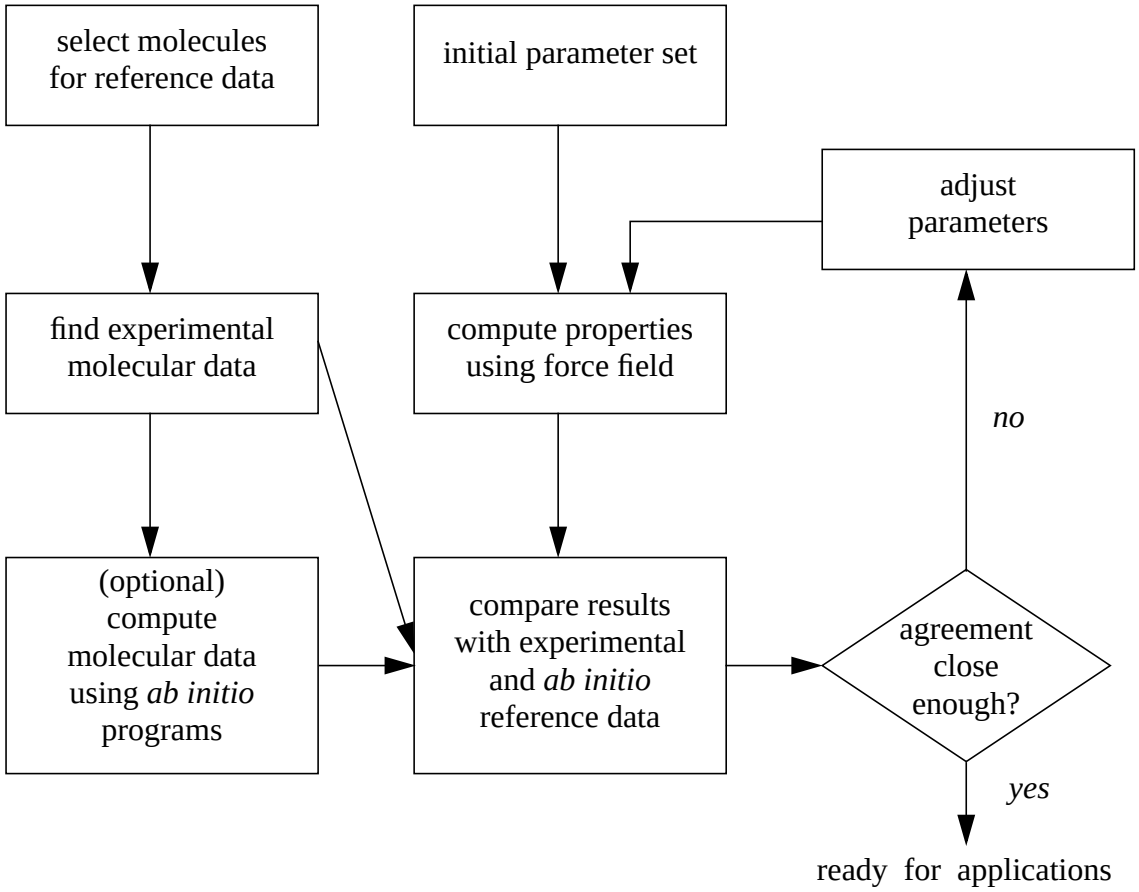


Figure 12: Distribution of ϕ-ψ values for residues from 16 high-resolution protein crystal structures (see Methods). Glycine and proline residues, and the two N-terminal and C-terminal residues in any chain, were excluded from the data set. A, residues whose backbone atoms are not involved in any hydrogen bond; B, residues whose backbone atoms are involved in one hydrogen bond; C, residues whose backbone atoms are involved in two hydrogen bonds; D, residues whose backbone atoms are involved in three hydrogen bonds.

Basic principle for a parameter development is an iterative adjustment



In principle one can envisage automated exploration of appropriate parameters, functional forms, cross terms, atom types etc.
In practice force field continues to rely heavily on experience, judgement and intuition.

3.5.3.2. Simulated annealing (SA)

High temperatures greatly increase the efficiency of producing conformational transitions according to the Arrhenius equation

$$k = \exp\left(\frac{-\Delta E}{RT}\right) \cdot A$$

I.e. for an energy barrier of 5 kcal mole^{-1} this leads to

1 transition per 100 ps	at 300 K and
250 transitions	at 900 K.

Another effect is that high temperature dynamics leads to higher energy minima since by starting from a higher point on the surface one falls into a higher local minimum.

The simulated annealing procedure may be regarded as a hybrid between a MC simulation technique and MD with an artificial temperature coupling and was introduced from crystallographic refinement (1987).

- The temperature of the system is raised (e.g. to 4.000 K) in order to make more conformations

accessible.
- An equilibration period (of 4 ps) at high temperature follows.
- By slow cooling (down to 300 K over 12 ps) local minima are overcome and the system arrives at a minimum of configurations which were accessible at high temperature.
- Repeating of the procedure for a second time.

The SA method has merits for constrained problems and performs an optimization using the kinetic energy to explore the large conformational space for a global minimum. By steady and slow lowering of the temperature strains can be removed and new minima in the surrounding are detected.
It suffer from the small time steps required for a simulation at this high temperatures and the long/ slow cooling period which lead to an inefficient long procedure.