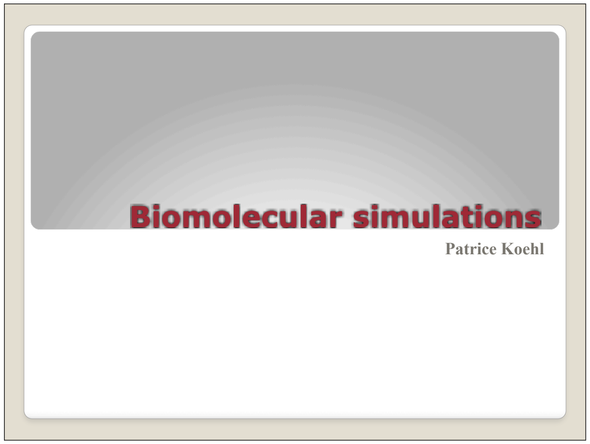
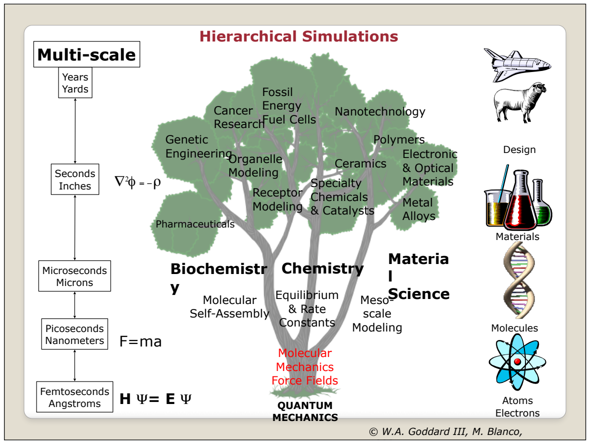


Biomolecular simulations

Patrice Koehl



Hierarchical Simulations

Multi-scale

Years
Yards

Seconds
Inches

Microseconds
Microns

Picoseconds
Nanometers

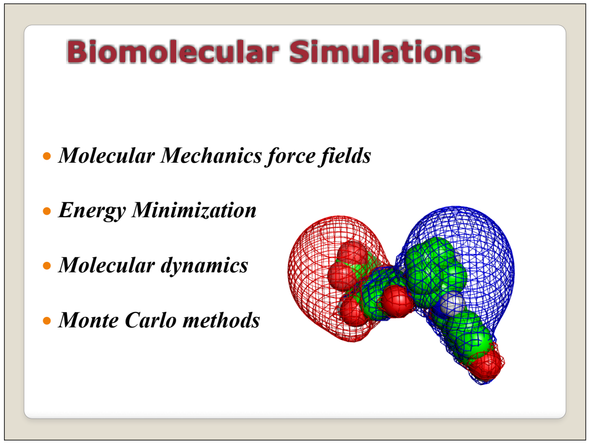
Femtoseconds
Angstroms

$\nabla^2\phi = -\rho$

F=ma

$H\ \Psi = E\ \Psi$

© W.A. Goddard III, M. Blanco,

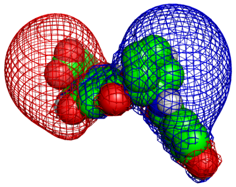


Biomolecular Simulations

- *Molecular Mechanics force fields*
- *Energy Minimization*
- *Molecular dynamics*
- *Monte Carlo methods*

Biomolecular Simulations

- ***Molecular Mechanics force fields***
- *Energy Minimization*
- *Molecular dynamics*
- *Monte Carlo methods*



The two major assumptions in molecular simulations

1. *Born-Oppenheimer approximation*

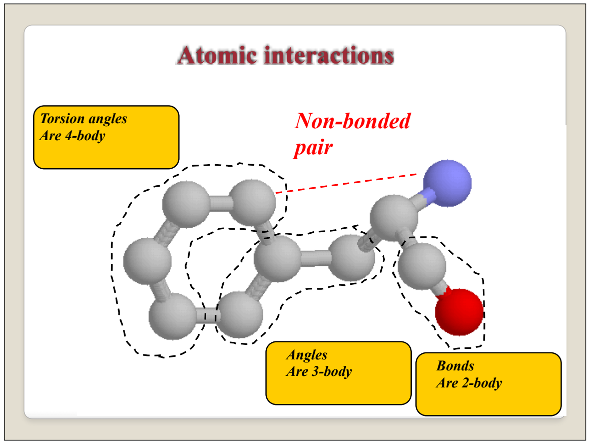
   “the dynamics of electrons is so fast that they can be considered to react instantaneously to the motion of their nuclei”

2. *Classical mechanics*

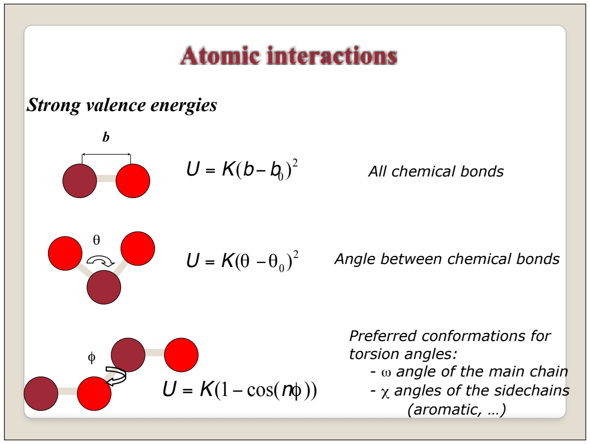
   “The nuclei are treated as point particles that follow the classical laws of mechanics.”

What is an atom?

- Classical mechanics: a point particle
- Defined by its position (x,y,z) and its mass
- May carry an electric charge (positive or negative), usually partial (less than an electron)



Atomic interactions



Atomic interactions

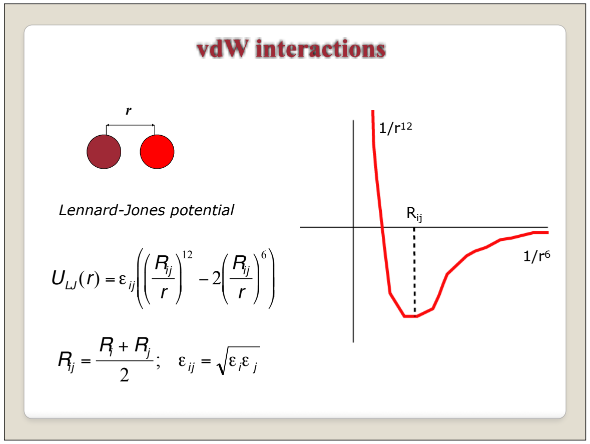
Strong valence energies

$U = K(b - b_0)^2$ *All chemical bonds*

$U = K(\theta - \theta_0)^2$ *Angle between chemical bonds*

$U = K(1 - \cos(n\phi))$

Preferred conformations for torsion angles:
- *ω angle of the main chain*
- *χ angles of the sidechains (aromatic, …)*

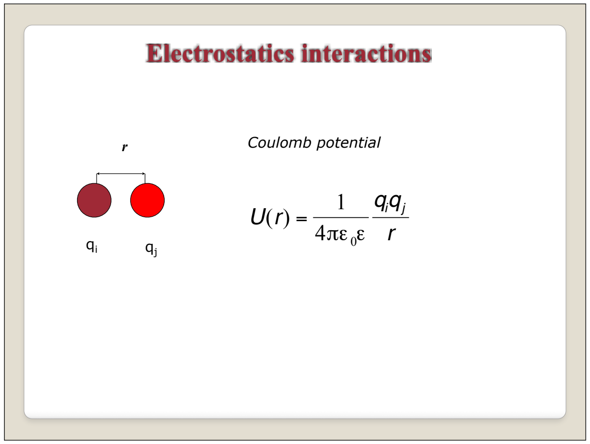


vdW interactions

Lennard-Jones potential

$$U_{LJ}(r) = \varepsilon_{ij}\left(\left(\frac{R_{ij}}{r}\right)^{12} - 2\left(\frac{R_{ij}}{r}\right)^{6}\right)$$

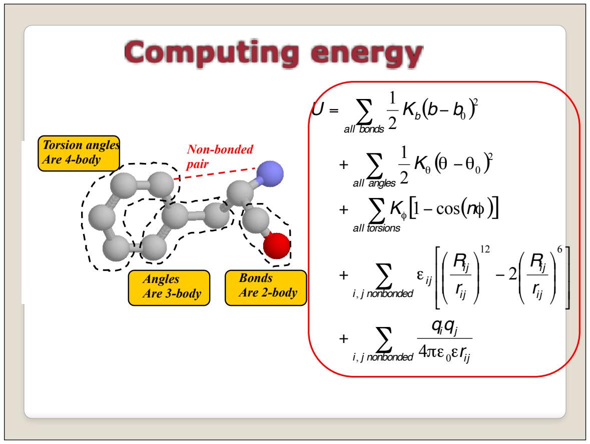
$$R_{ij} = \frac{R_i + R_j}{2}; \quad \varepsilon_{ij} = \sqrt{\varepsilon_i \varepsilon_j}$$



Electrostatics interactions

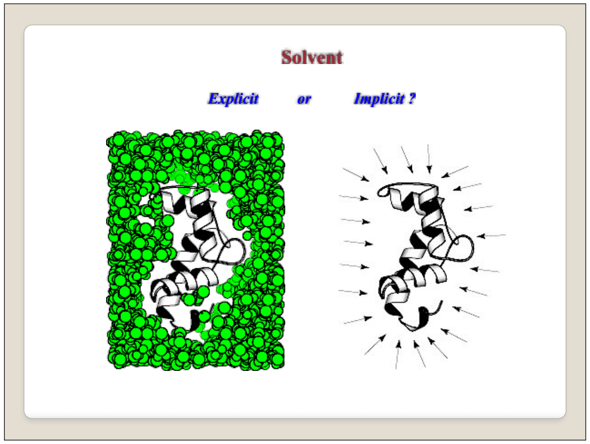
Coulomb potential

$$U(r) = \frac{1}{4\pi\varepsilon_0\varepsilon}\frac{q_iq_j}{r}$$



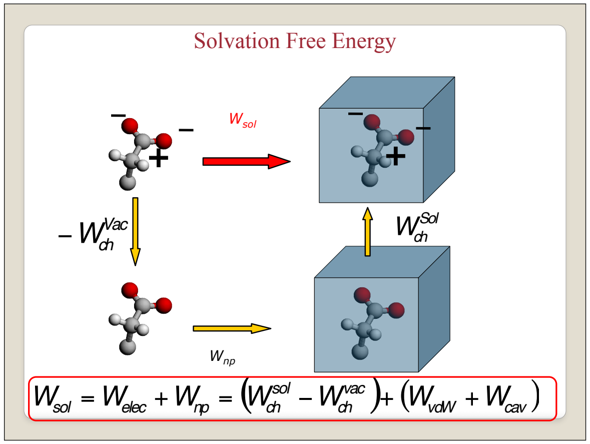
Computing energy

$$U = \sum_{all\ bonds}\frac{1}{2}K_b(b-b_0)^2 + \sum_{all\ angles}\frac{1}{2}K_\theta(\theta-\theta_0)^2 + \sum_{all\ torsions}K_\phi[1-\cos(n\phi)] + \sum_{i,j\ nonbonded}\varepsilon_{ij}\left[\left(\frac{R_{ij}}{r_{ij}}\right)^{12} - 2\left(\frac{R_{ij}}{r_{ij}}\right)^{6}\right] + \sum_{i,j\ nonbonded}\frac{q_iq_j}{4\pi\varepsilon_0\varepsilon r_{ij}}$$



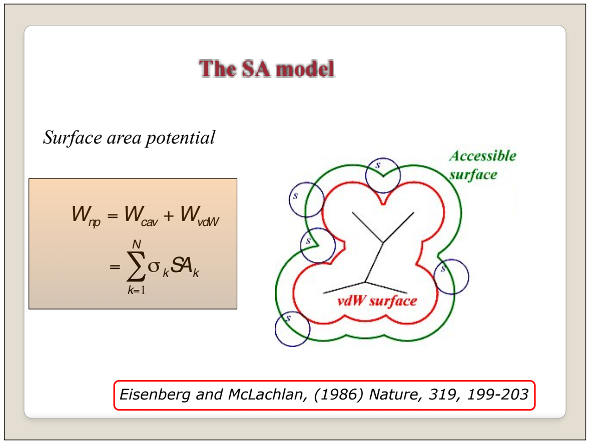
Solvent

Explicit *or* *Implicit ?*



Solvation Free Energy

$$W_{sol} = W_{elec} + W_{np} = \left(W_{ch}^{sol} - W_{ch}^{vac}\right) + \left(W_{vdW} + W_{cav}\right)$$

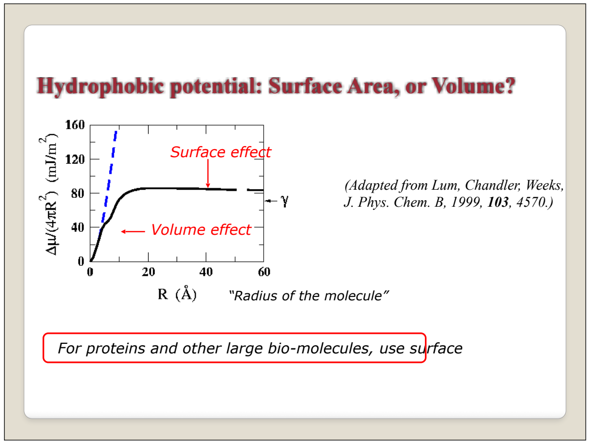


The SA model

Surface area potential

$$W_{np} = W_{cav} + W_{vdW} = \sum_{k=1}^{N} \sigma_k SA_k$$

Eisenberg and McLachlan, (1986) Nature, 319, 199-203

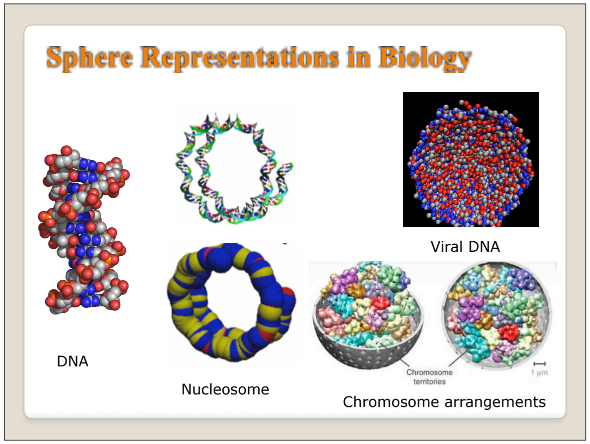


Hydrophobic potential: Surface Area, or Volume?

"Radius of the molecule"

(Adapted from Lum, Chandler, Weeks, J. Phys. Chem. B, 1999, ***103****, 4570.)*

For proteins and other large bio-molecules, use surface



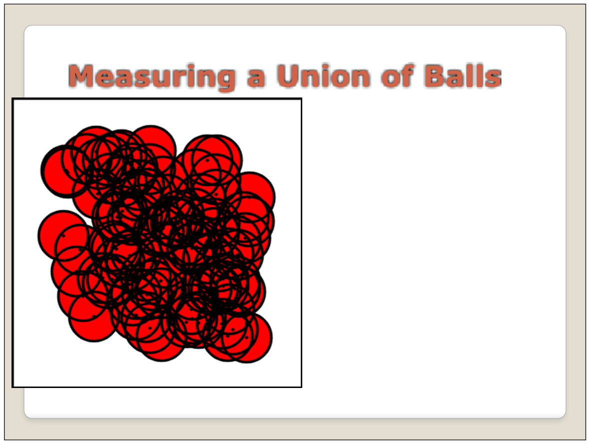
Sphere Representations in Biology

Viral DNA

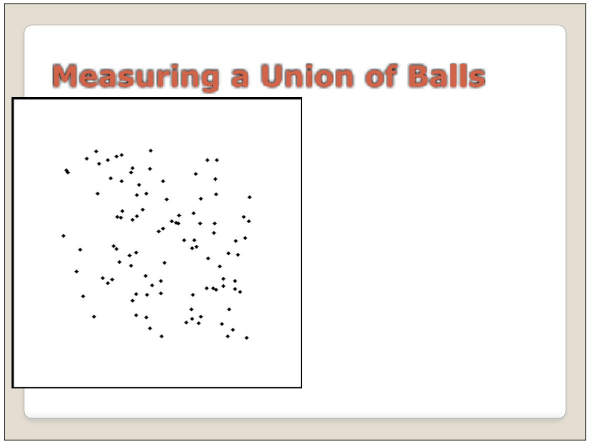
DNA

Nucleosome

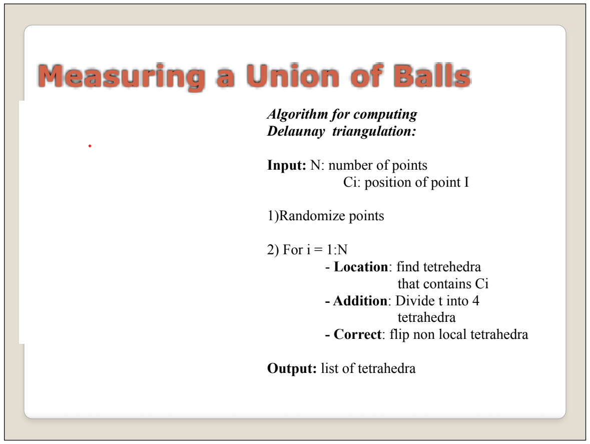
Chromosome arrangements



Measuring a Union of Balls



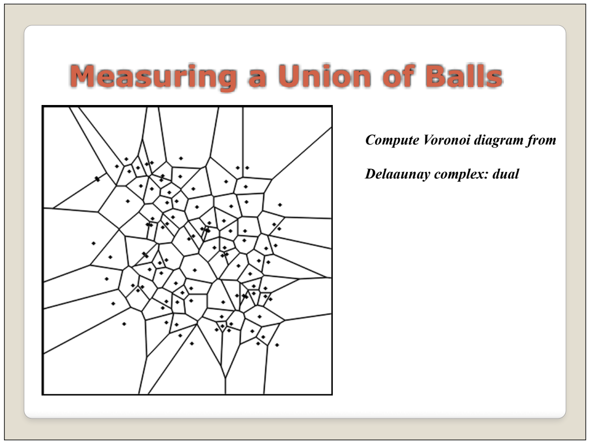
Measuring a Union of Balls



Measuring a Union of Balls

Algorithm for computing Delaunay triangulation:

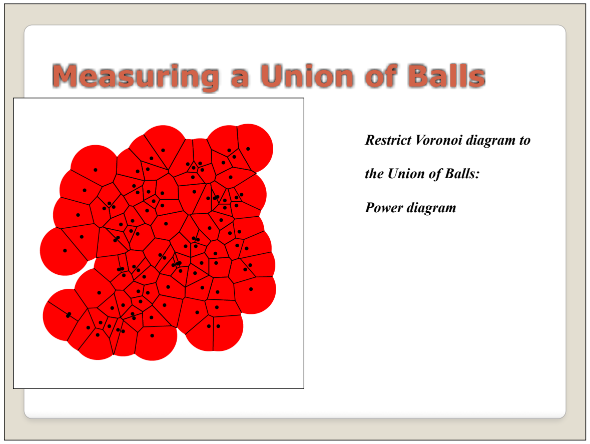
```
Input: N: number of points
           Ci: position of point I

1)Randomize points

2) For i = 1:N
        - Location: find tetrehedra
                    that contains Ci
        - Addition: Divide t into 4
                    tetrahedra
        - Correct: flip non local tetrahedra

Output: list of tetrahedra
```



Measuring a Union of Balls

Compute Voronoi diagram from

Delaaunay complex: dual

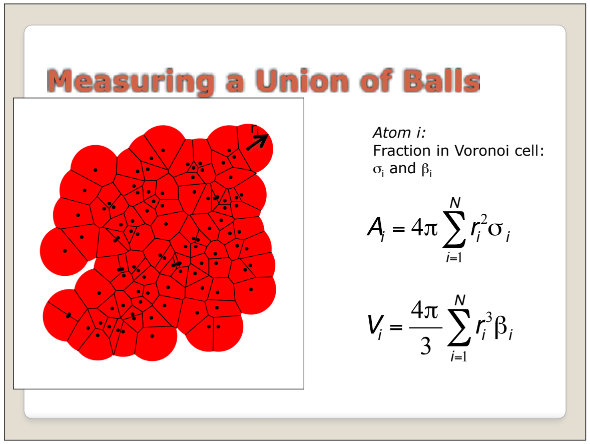


Measuring a Union of Balls

Restrict Voronoi diagram to

the Union of Balls:

Power diagram

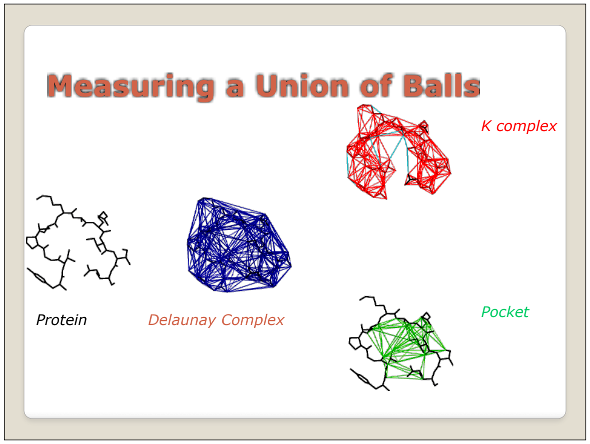


Measuring a Union of Balls

Atom i:
Fraction in Voronoi cell:
σ_i and β_i

$$A_i = 4\pi \sum_{i=1}^{N} r_i^2 \sigma_i$$

$$V_i = \frac{4\pi}{3} \sum_{i=1}^{N} r_i^3 \beta_i$$



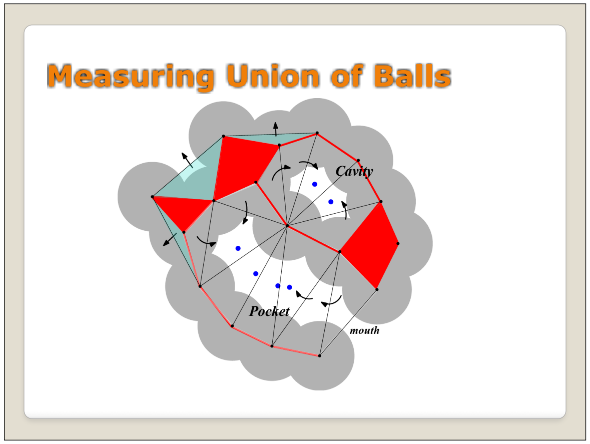
Measuring a Union of Balls

K complex

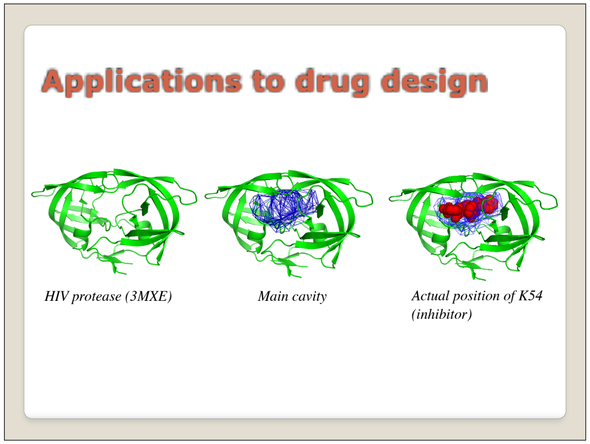
Protein

Delaunay Complex

Pocket



Measuring Union of Balls

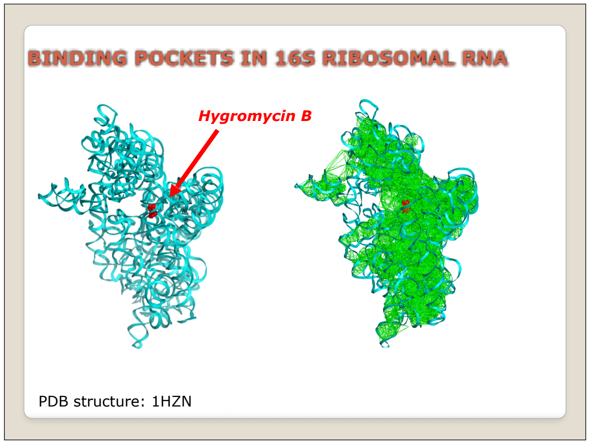


Applications to drug design

HIV protease (3MXE)

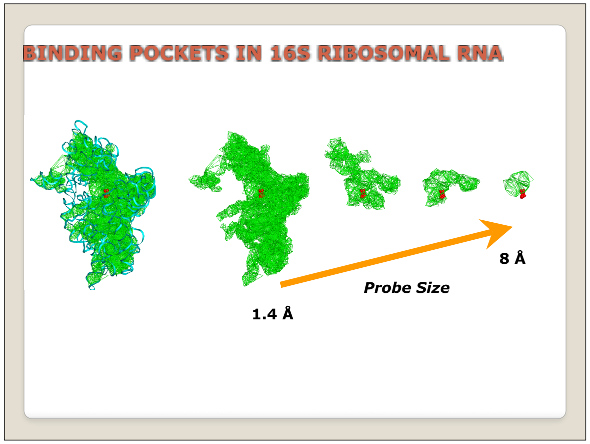
Main cavity

Actual position of K54 (inhibitor)



BINDING POCKETS IN 16S RIBOSOMAL RNA

PDB structure: 1HZN



BINDING POCKETS IN 16S RIBOSOMAL RNA

Computing energy

Bonded interactions are local, and therefore their computation has a linear computational complexity (O(N), where N is the number of atoms in the molecule considered.

The direct computation of the non bonded interactions involve all pairs of atoms and has a quadratic complexity (O(N2)).
This can be prohibitive for large molecules.

$$U_{NB} = \sum_{i,j\ nonbonded} \varepsilon_{ij}\left[\left(\frac{R_{ij}}{r_{ij}}\right)^{12} - 2\left(\frac{R_{ij}}{r_{ij}}\right)^{6}\right] + \sum_{i,j\ nonbonded} \frac{q_i q_j}{4\pi\varepsilon_0 \varepsilon r_{ij}}$$

Cutoff schemes for faster energy computation

$$U_{NB} = \sum_{i,j} \omega_{ij} S(r_{ij}) \varepsilon_{ij}\left[\left(\frac{R_{ij}}{r_{ij}}\right)^{12} - 2\left(\frac{R_{ij}}{r_{ij}}\right)^{6}\right] + \sum_{i,j} \omega_{ij} S(r_{ij}) \frac{q_i q_j}{4\pi\varepsilon_0 \varepsilon r_{ij}}$$

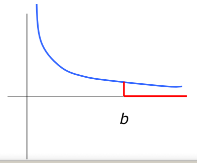
- ω_{ij} : weights *(0< ω_{ij} <1). Can be used to exclude bonded terms, or to scale some interactions (usually 1-4)*

- *S(r) : cutoff function*.

Three types:

1) Truncation:

$$S(r) = \begin{cases} 1 & r < b \\ 0 & r \geq b \end{cases}$$

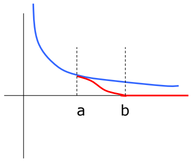


Cutoff schemes for faster energy computation

2. Switching

$$S(r) = \begin{cases} 1 & r < a \\ 1 + y(r)^2\left[2y(r) - 3\right] & a \leq r \leq b \\ 0 & r > b \end{cases}$$

with $y(r) = \dfrac{r^2 - a^2}{b^2 - a^2}$

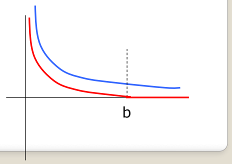


3. Shifting

$$S_1(r) = \left[1 - \left(\frac{r}{b}\right)^2\right]^2 \quad r \leq b$$

or

$$S_2(r) = \left[1 - \frac{r}{b}\right]^2 \quad r \leq b$$



Units in Molecular Simulations

Most force fields use the AKMA (Angstrom – Kcal – Mol – Atomic Mass Unit) unit system:

Quantity	AKMA unit	Equivalent SI
Energy	1 Kcal/Mol	4184 Joules
Length	1 Angstrom	10^{-10} meter
Mass	1 amu (H=1amu)	1.6605655 10^{-27} Kg
Charge	1 e	1.6021892 10^{-19} C
Time	1 unit	4.88882 10^{-14} second
Frequency	1 cm-1	18.836 10^{10} rd/s

Some Common force fields in Computational Biology

ENCAD (Michael Levitt, Stanford)

AMBER (Peter Kollman, UCSF; David Case, Scripps)

CHARMM (Martin Karplus, Harvard)

OPLS (Bill Jorgensen, Yale)

MM2/MM3/MM4 (Norman Allinger, U. Georgia)

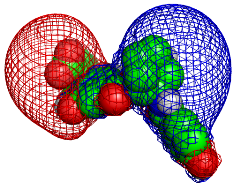
ECEPP (Harold Scheraga, Cornell)

GROMOS (Van Gunsteren, ETH, Zurich)

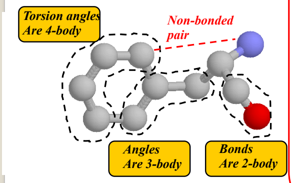
Michael Levitt. The birth of computational structural biology. Nature Structural Biology, 8, 392- (2001)

Biomolecular Simulations

- *Molecular Mechanics force fields*
- ***Energy Minimization***
- *Molecular dynamics*
- *Monte Carlo methods*



Computing energy



$$\begin{aligned} U = & \sum_{\text{all bonds}} \frac{1}{2} K_b (b - b_0)^2 \\ & + \sum_{\text{all angles}} \frac{1}{2} K_\theta (\theta - \theta_0)^2 \\ & + \sum_{\text{all torsions}} K_\phi [1 - \cos(n\phi)] \\ & + \sum_{i,j \text{ nonbonded}} \varepsilon_{ij} \left[\left(\frac{R_{ij}}{r_{ij}} \right)^{12} - 2 \left(\frac{R_{ij}}{r_{ij}} \right)^{6} \right] \\ & + \sum_{i,j \text{ nonbonded}} \frac{q_i q_j}{4\pi\varepsilon_0 \varepsilon r_{ij}} \end{aligned}$$

U is a function of the conformation C of the protein.
The problem of "minimizing U" can be stated as finding C such that U(C) is minimum.

The minimizers

Minimization of a multi-variable function is usually an iterative process, in which updates of the state variable x are computed using the gradient, and in some (favorable) cases the Hessian.

Iterations are stopped either when the maximum number of steps (user's input) is reached, or when the gradient norm is below a given threshold.

Steepest descent (SD):

The simplest iteration scheme consists of following the "steepest descent" direction:

$$\boxed{x_{k+1} = x_k - \alpha \nabla f(x_k)}$$

(α sets the minimum along the line defined by the gradient)

Usually, SD methods leads to improvement quickly, but then exhibit slow progress toward a solution.

They are commonly recommended for initial minimization iterations, when the starting function and gradient-norm values are very large.

The minimizers

Conjugate gradients (CG):

In each step of conjugate gradient methods, a search vector p_k is defined by a recursive formula:

$$\boxed{p_{k+1} = -\nabla f(x_k) + \beta_{k+1} p_k}$$

The corresponding new position is found by line minimization along p_k:

$$\boxed{x_{k+1} = x_k + \lambda_k p_k}$$

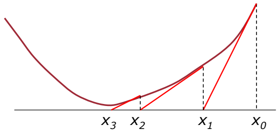
the CG methods differ in their definition of β.

The minimizers

Newton's methods:

Newton's method is a popular iterative method for finding the 0 of a one-dimensional function:

$$\boxed{x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}}$$



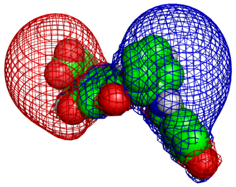
It can be adapted to the minimization of a one –dimensional function, in which case the iteration formula is:

$$\boxed{x_{k+1} = x_k - \frac{g'(x_k)}{g''(x_k)}}$$

Several implementations of Newton's method exist: quasi-Newton, truncated Newton, "adopted-basis Newton-Raphson" (ABNR),…

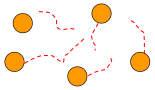
Biomolecular Simulations

- *Molecular Mechanics force fields*
- *Energy Minimization*
- ***Molecular dynamics***
- *Monte Carlo methods*



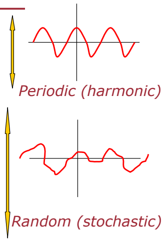
What is a molecular dynamics simulation?

- Simulation that shows how the atoms in the system move with time
- Typically on the nanosecond timescale
- Atoms are treated like hard balls, and their motions are described by Newton's laws.



Characteristic protein motions

Type of motion	Timescale	Amplitude
Local:		
bond stretching	0.01 ps	
angle bending	0.1 ps	< 1 Å
methyl rotation	1 ps	
Medium scale		
loop motions		
SSE formation	ns – μs	1-5 Å
Global		
protein tumbling	20 ns	
(water tumbling)	(20 ps)	> 5 Å
protein folding	ms – hrs	



Why MD simulations?

- Link physics, chemistry and biology
- Model phenomena that cannot be observed experimentally
- Understand protein folding…
- Access to thermodynamics quantities (free energies, binding energies,…)

How do we run a MD simulation?

- **Get the initial configuration**

From x-ray crystallography or NMR spectroscopy (PDB)

- **Assign initial velocities**

At thermal equilibrium, the expected value of the kinetic energy of the system at temperature T is:

$$\langle E_{kin} \rangle = \frac{1}{2}\sum_{i=1}^{3N} m_i v_i^2 = \frac{1}{2}(3N)k_B T$$

This can be obtained by assigning the velocity components vi from a random Gaussian distribution with mean 0 and standard deviation ($k_B T/m_i$):

$$\langle v_i^2 \rangle = \frac{k_B T}{m_i}$$

How do we run a MD simulation?

- **For each time step:**
  - Compute the force on each atom:

$$F(X) = -\nabla E(X) = -\frac{\partial E}{\partial X}$$

X: cartesian vector of the system

  - Solve Newton's 2nd law of motion for each atom, to get new coordinates and velocities

$$M\ddot{X} = F(X)$$

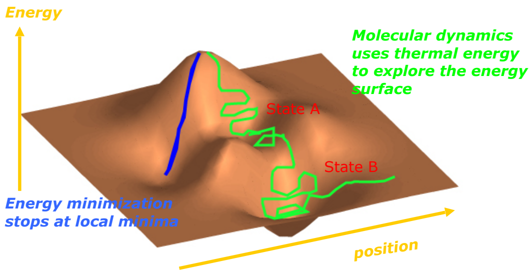
M diagonal mass matrix
$\ddot{}$ means second order differentiation with respect to time

  - Store coordinates

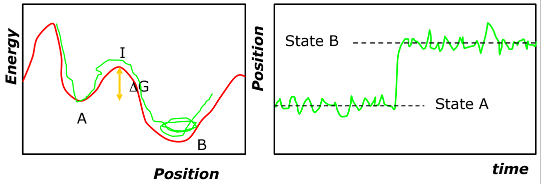
- Stop

Newton's equation cannot be solved analytically:
→ Use stepwise numerical integration

MD as a tool for minimization


Energy
Molecular dynamics uses thermal energy to explore the energy surface
State A
State B
Energy minimization stops at local minima
position

Crossing energy barriers



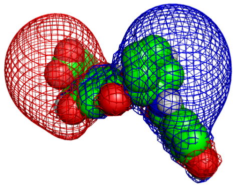
The actual transition time from A to B is very quick (a few pico seconds).

What takes time is waiting. The average waiting time for going from A to B can be expressed as:

$$\tau_{A\to B} = Ce^{\frac{\Delta G}{kT}}$$

Biomolecular Simulations

- *Molecular Mechanics force fields*
- *Energy Minimization*
- *Molecular dynamics*
- ***Monte Carlo methods***



Monte Carlo: random sampling

A simple example:

Evaluate numerically the one-dimensional integral:

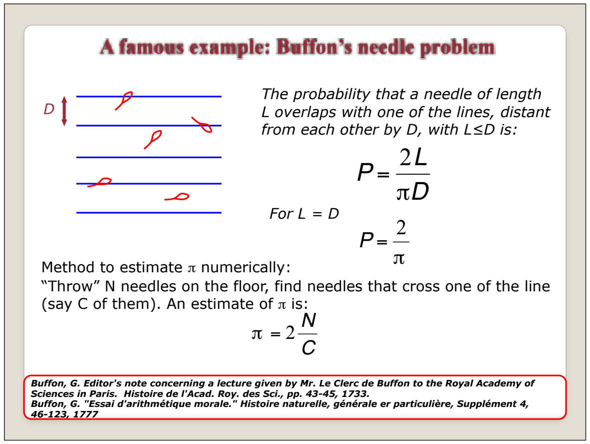
$$I = \int_a^b f(x)dx$$

Instead of using classical quadrature, the integral can be rewritten as

$$I = (b-a)\langle f(x)\rangle$$

<f(x)> denotes the unweighted average of f(x) over [a,b], and can be determined by evaluating f(x) at a large number of x values randomly distributed over [a,b]

Monte Carlo method!



A famous example: Buffon's needle problem

The probability that a needle of length L overlaps with one of the lines, distant from each other by D, with L≤D is:

$$P = \frac{2L}{\pi D}$$

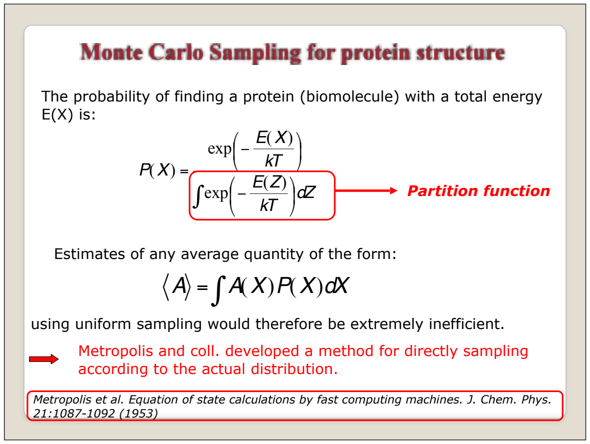
For L = D

$$P = \frac{2}{\pi}$$

Method to estimate π numerically:
"Throw" N needles on the floor, find needles that cross one of the line (say C of them). An estimate of π is:

$$\pi = 2\frac{N}{C}$$

Buffon, G. Editor's note concerning a lecture given by Mr. Le Clerc de Buffon to the Royal Academy of Sciences in Paris. Histoire de l'Acad. Roy. des Sci., pp. 43-45, 1733.
Buffon, G. "Essai d'arithmétique morale." Histoire naturelle, générale er particulière, Supplément 4, 46-123, 1777



Monte Carlo Sampling for protein structure

The probability of finding a protein (biomolecule) with a total energy E(X) is:

$$P(X) = \frac{\exp\left(-\frac{E(X)}{kT}\right)}{\boxed{\int \exp\left(-\frac{E(Z)}{kT}\right) dZ}}$$

Partition function

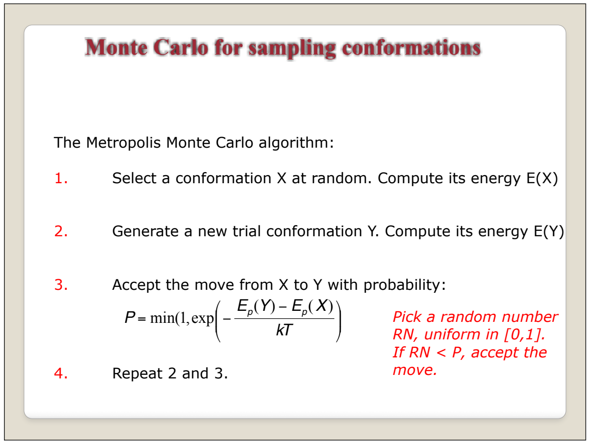
Estimates of any average quantity of the form:

$$\langle A \rangle = \int A(X)P(X)dX$$

using uniform sampling would therefore be extremely inefficient.

➡ Metropolis and coll. developed a method for directly sampling according to the actual distribution.

Metropolis et al. Equation of state calculations by fast computing machines. J. Chem. Phys. 21:1087-1092 (1953)



Monte Carlo for sampling conformations

The Metropolis Monte Carlo algorithm:

1. Select a conformation X at random. Compute its energy E(X)
2. Generate a new trial conformation Y. Compute its energy E(Y)
3. Accept the move from X to Y with probability:

$$P = \min\left(1, \exp\left(-\frac{E_p(Y) - E_p(X)}{kT}\right)\right)$$

Pick a random number RN, uniform in [0,1]. If RN < P, accept the move.

4. Repeat 2 and 3.