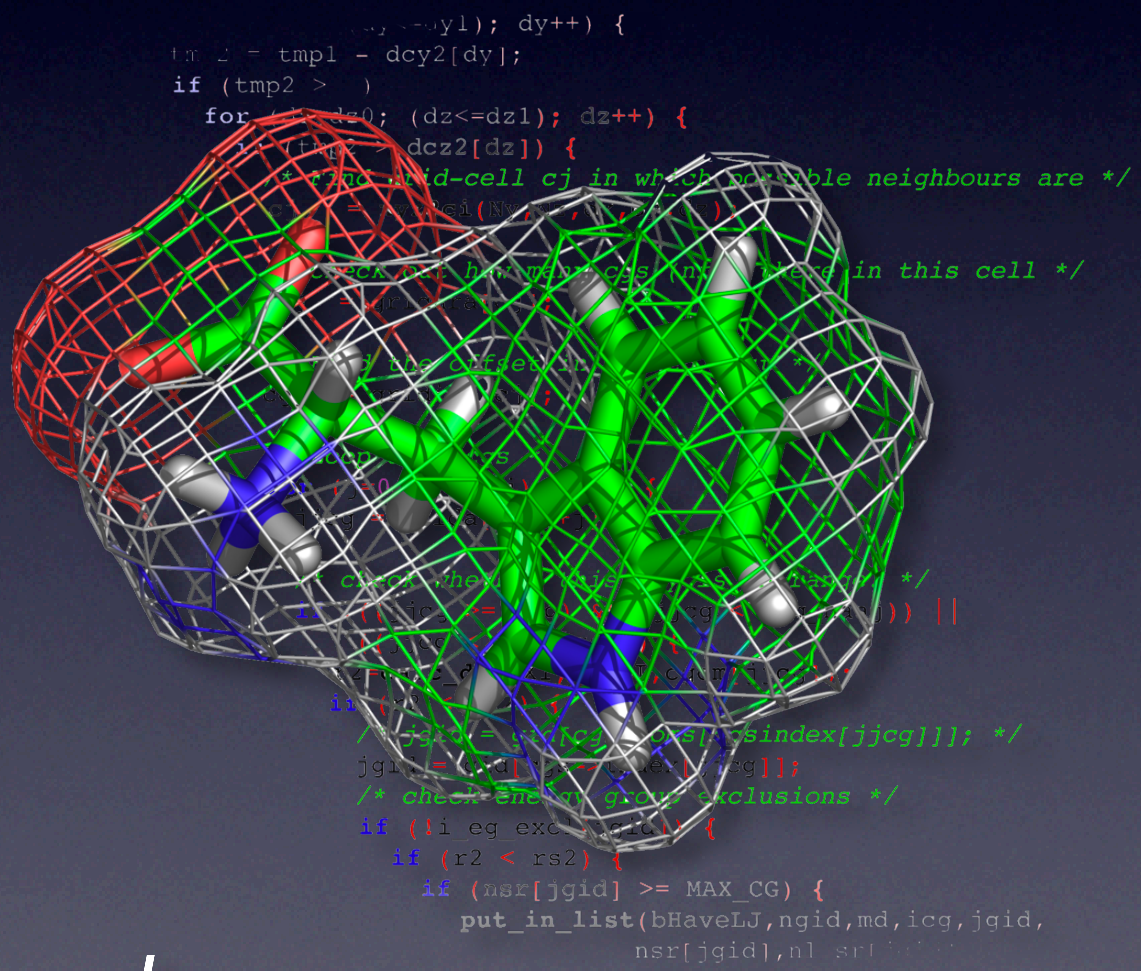
Gromacs Workshop Spring 2007 @ CSC

Erik Lindahl
Center for Biomembrane Research
Stockholm University, Sweden

David van der Spoel
Dept. Cell & Molecular Biology
Uppsala University, Sweden

Berk Hess
Max-Planck-Institut for Polymer Research
Mainz, Germany


dy++) {
tmp2 = tmp1 - dcy2[dy];
if (tmp2 >
for (dz<=dz1); dz++) {
dcz2[dz]) {
/* grid-cell cj in which possible neighbours are */
in this cell */
jgid = gid[icg + /s[sindex[jjcg]]]; */
/* check the group exclusions */
if (!i_eq_exc
if (r2 < rs2) {
if (nsr[jgid] >= MAX_CG) {
put_in_list(bHaveLJ,ngid,md,icg,jgid,
nsr[jgid],nl_sr

For CSC: Atte Sillanpää

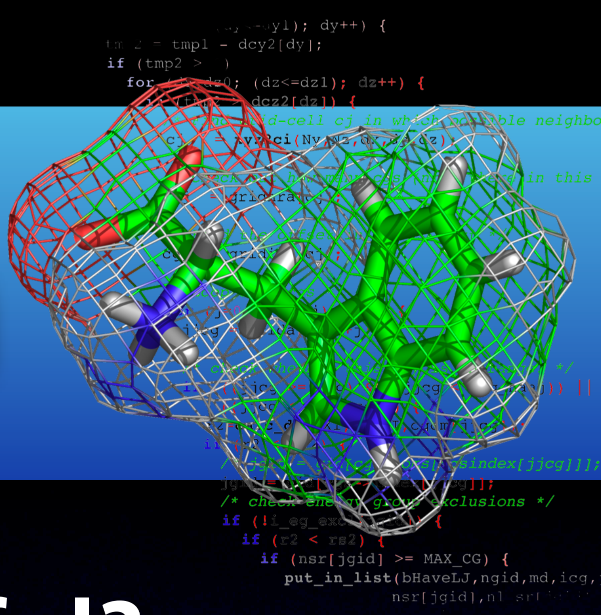
Introduction to Molecular Simulation and Gromacs

Erik Lindahl

lindahl@cbr.su.se
lindahl@gromacs.org

CBR

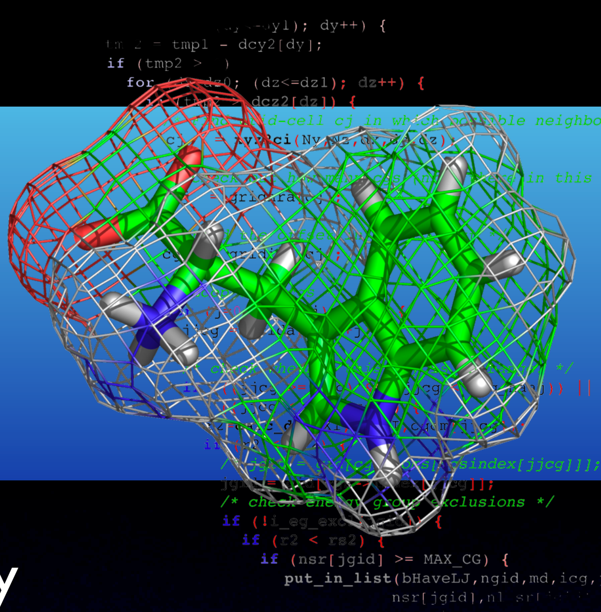
Outline: Introduction



```
            dy++) {
   tmp1 - dcy2[dy];
if (tmp2 >   )
  for (   dz0; (dz<=dz1); dz++) {
             dcz2[dz]) {
            /* check energy group exclusions */
            if (!i_eg_excl
              if (r2 < rs2) {
                if (nsr[jgid] >= MAX_CG) {
                  put_in_list(bHaveLJ,ngid,md,icg,
                                     nsr[jgid],nl
```

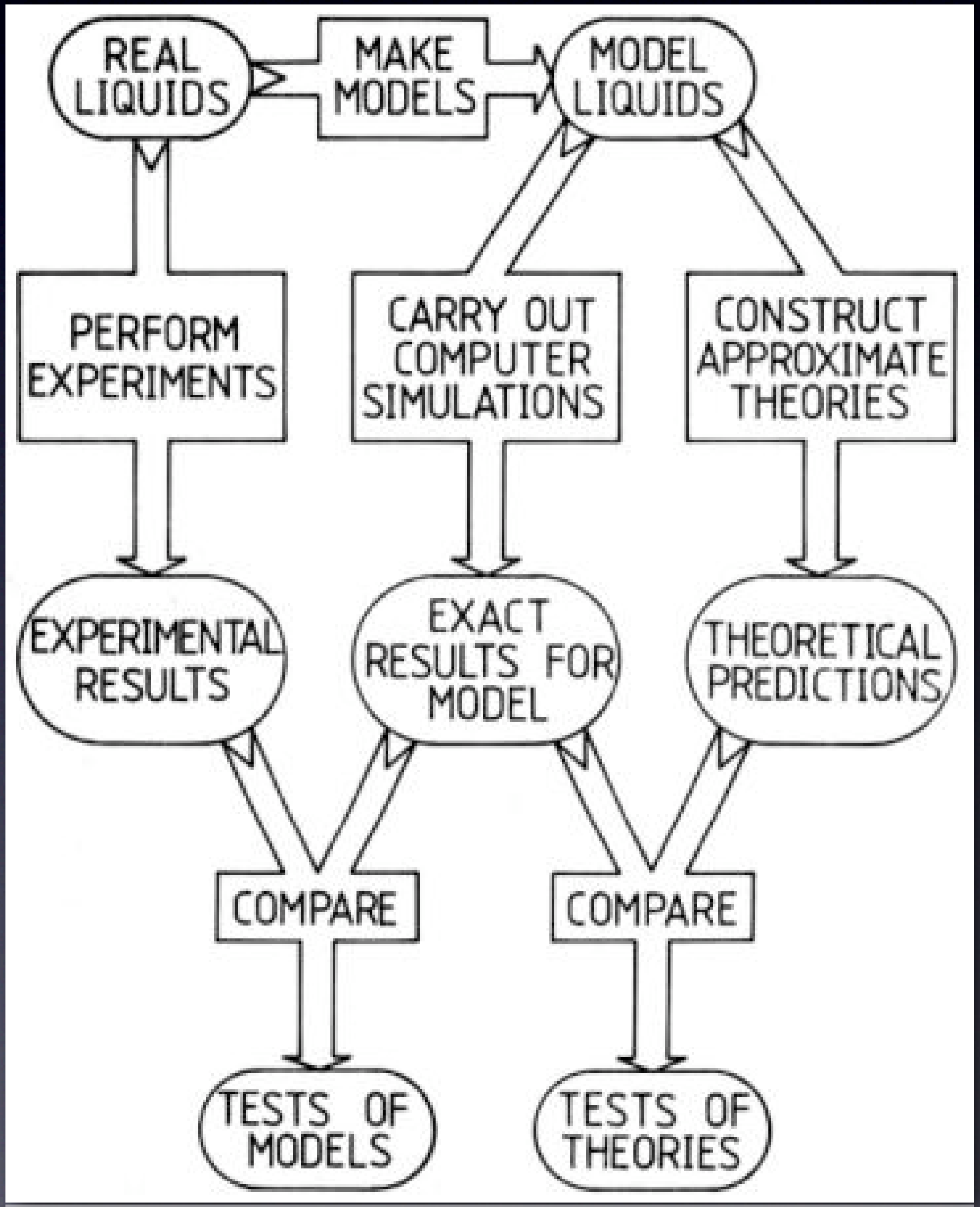
- Why and when are simulations useful?
- Various approaches, approximations
- Things we can (and cannot) model
- Time and length scales accessible
- Interactions, equations of motion
- Periodic boundary conditions & cells
- Ensembles: temperature, pressure, etc.
- When can you trust the results?

Why Simulations?



- Test models
- Test theories
- “atomic microscope”
- Way too many simulations are performed without any hypothesis / clear plan
- Think first, then simulate!

Allen & Tildesley



REAL LIQUIDS

MAKE MODELS

MODEL LIQUIDS

PERFORM EXPERIMENTS

CARRY OUT COMPUTER SIMULATIONS

CONSTRUCT APPROXIMATE THEORIES

EXPERIMENTAL RESULTS

EXACT RESULTS FOR MODEL

THEORETICAL PREDICTIONS

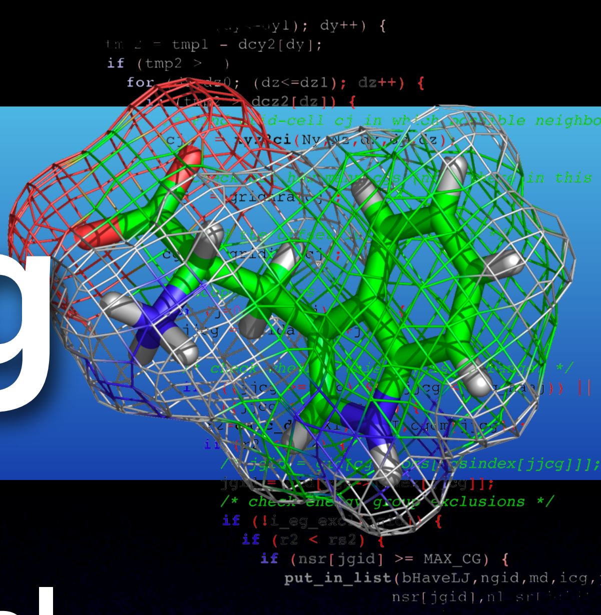
COMPARE

COMPARE

TESTS OF MODELS

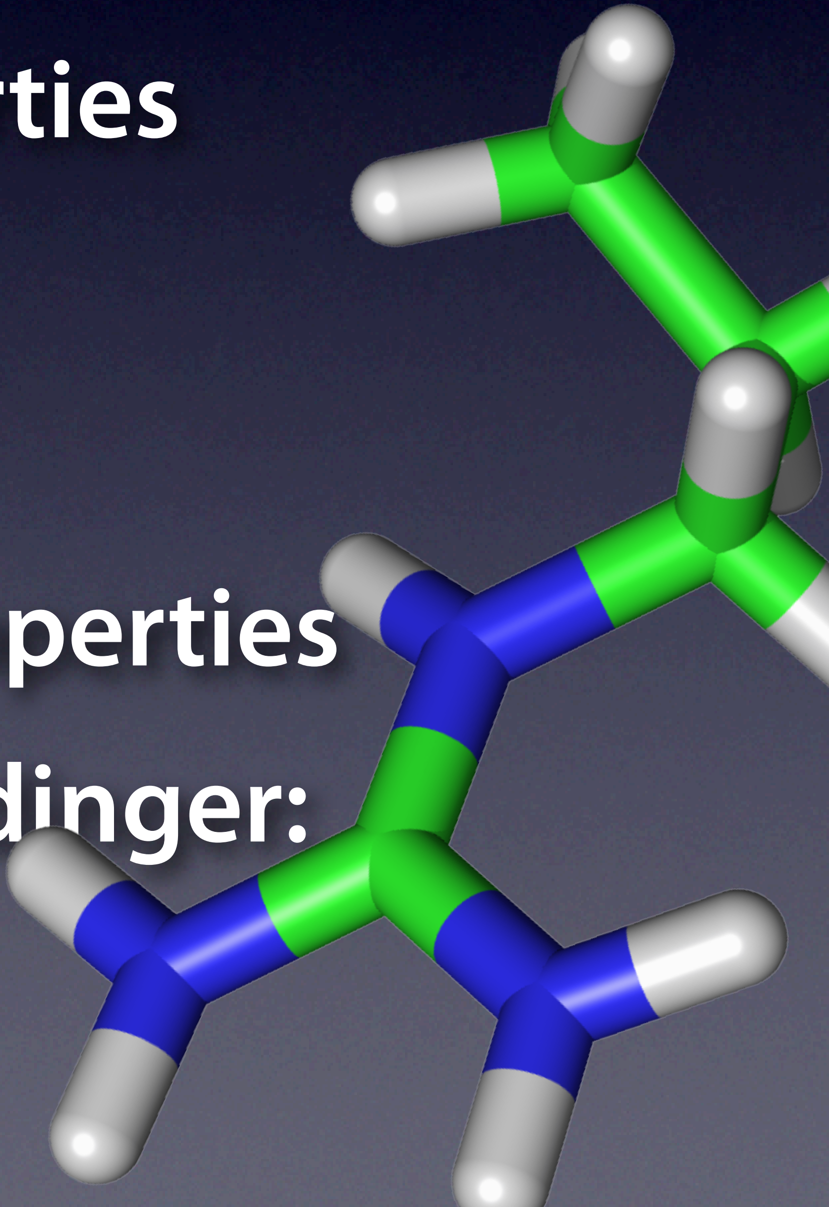
TESTS OF THEORIES

Molecular Modeling

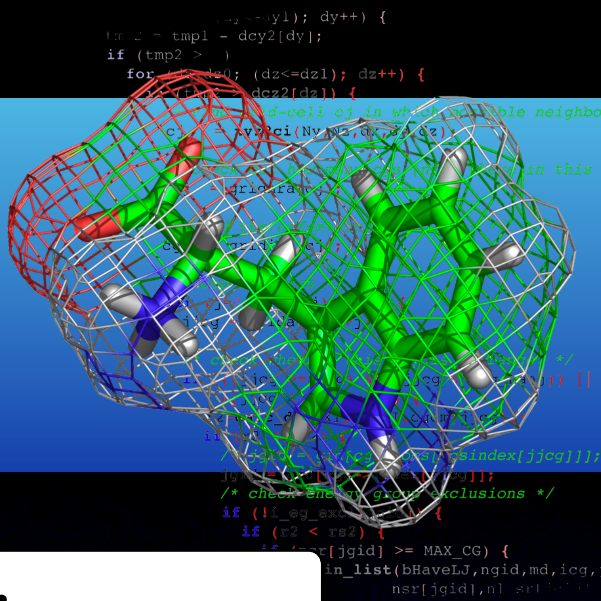

yl); dy++) {
tmp1 - dcy2[dy];
if (tmp2 >
for (dz=dz0; (dz<=dz1); dz++) {
dcz2[dz]) {
ble neighbo
in this
sindex[jjcg]]];
/* check the group exclusions */
if (nsr[jgid] >= MAX_CG) {
put_in_list(bHaveLJ,ngid,md,icg,
nsr[jgid],nl

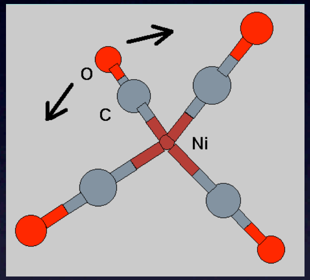
- The art of describing complex chemical systems from atomic/quantum models
- Prediction of macroscopic properties
  - Ensemble averages
  - Static / Equilibrium properties
  - Dynamic / Non-equilibrium properties
  - Everything described by Schrödinger:

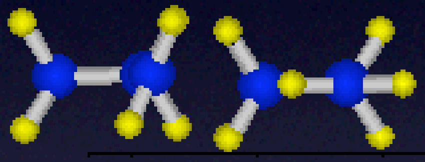
$$H\Psi(x,t) = E\Psi(x,t)$$

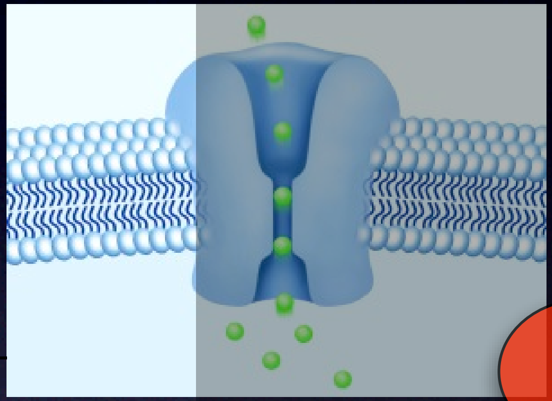


Timescales









Experiments

Efficient averaging

Less detail

Where we need to be

10^{-15}s 10^{-12}s 10^{-9}s 10^{-6}s 10^{-3}s 10^{0}s 10^{3}s

Simulations

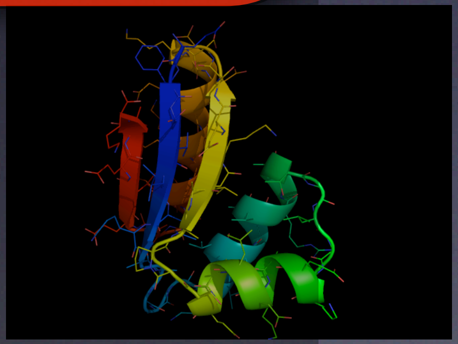
Where we are

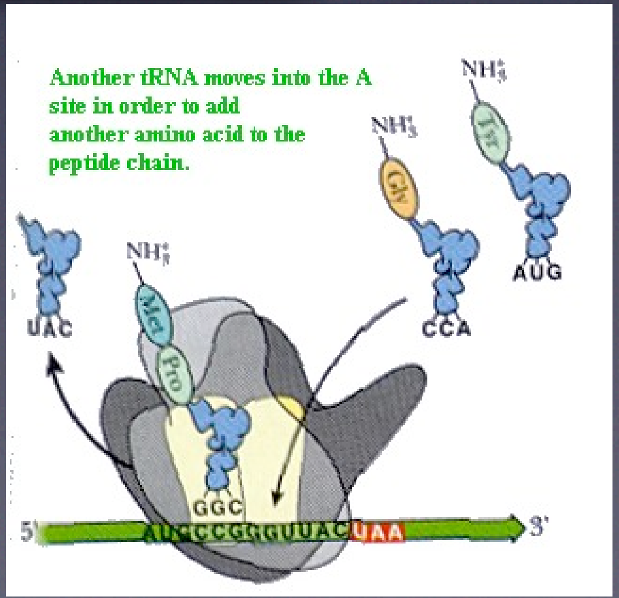
Where we want to be

Extreme detail

Sampling issues?

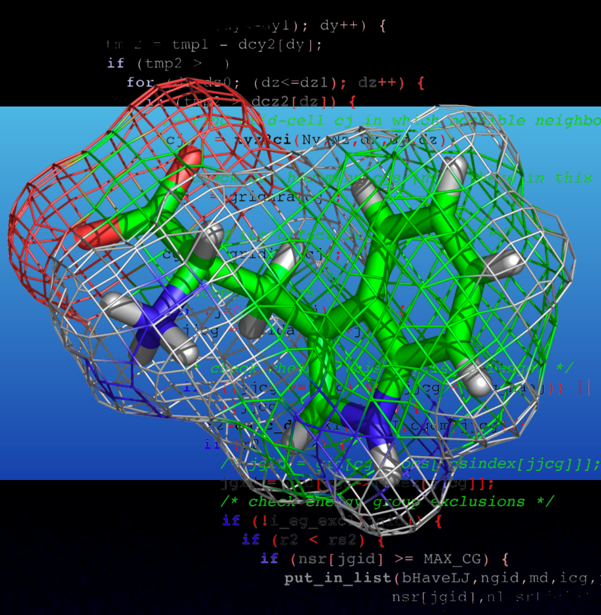
Parameter quality?





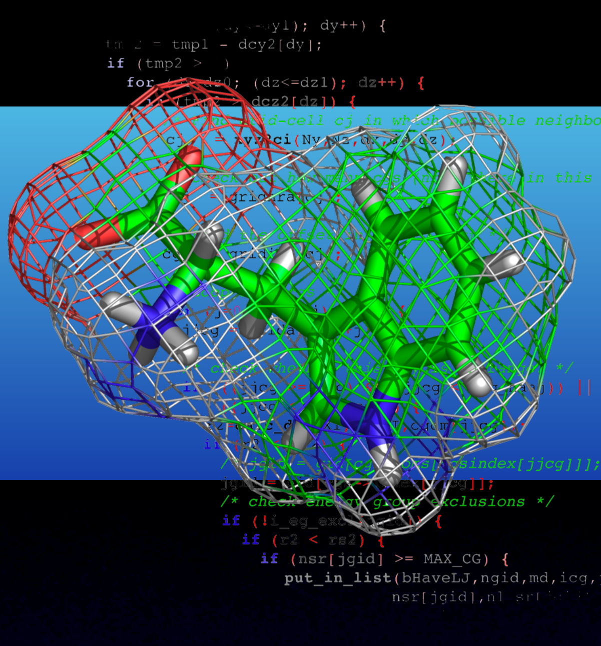
Another tRNA moves into the A site in order to add another amino acid to the peptide chain.

Limitations of QM



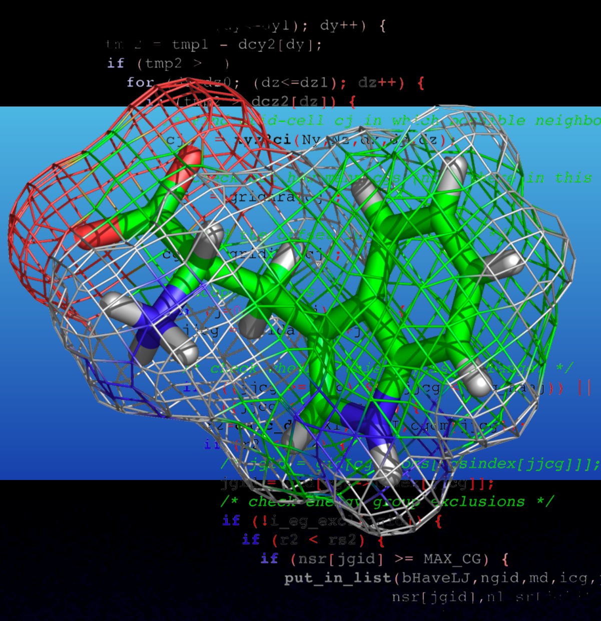
- QM is necessary to describe electronic, and sometimes hydrogen, motion
- Heavy atoms can be treated classically
- Can only solve up to ~100 atoms accurately
- No time dependence
- Several approximations exist, but they are only accurate on QM scale - you cannot extrapolate 10 orders of magnitude!

Empirical models



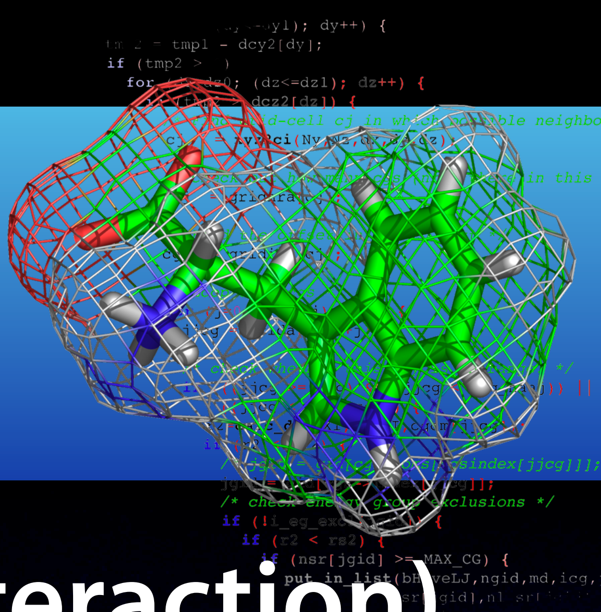
- Classical particles
- Parameterize to *reproduce experiments*
- Some properties like charge, bond parameters can be obtained from QM (Semi-empirical models)
- Atomic simulation is *not* always the best choice
    - coarse-grained methods (Marrink)
    - QSAR, chemoinformatics

Example properties



- Free energy (of binding, solvation, interaction)
- Diffusion coefficients, viscosity
- Reaction rates, phase transition properties
- Protein folding times
- Structure refinement
- All of these are *ensemble averages* over huge numbers of molecules ($N_{avogadro}$ ~6E23)

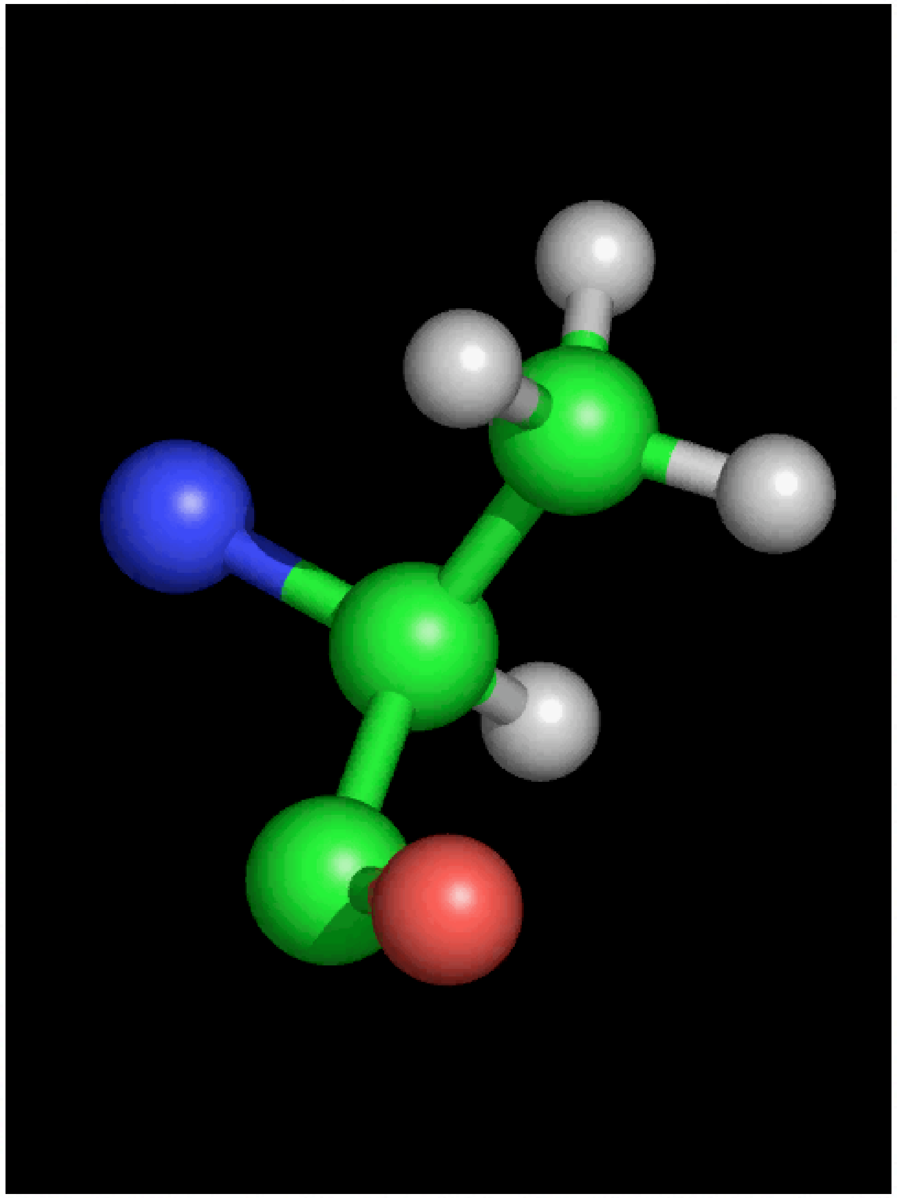
Example properties



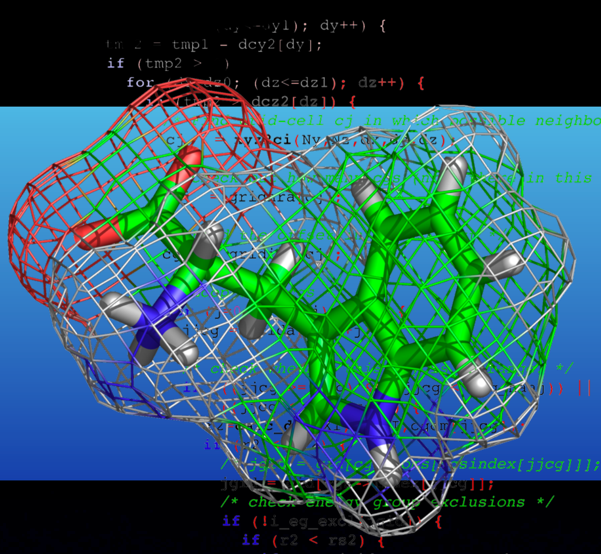
- Free energy (of binding, solvation, interaction)
- Diffusion coefficients
- Reaction rates
- Protein folding times
- Structure refinement
- All of these are *ensemble averages* of many molecules ($N_{avogadro}$ ~6E23)
- Thus, our goal is to sample equilibrium ensembles, not simulate individual particles!

Sampling methods

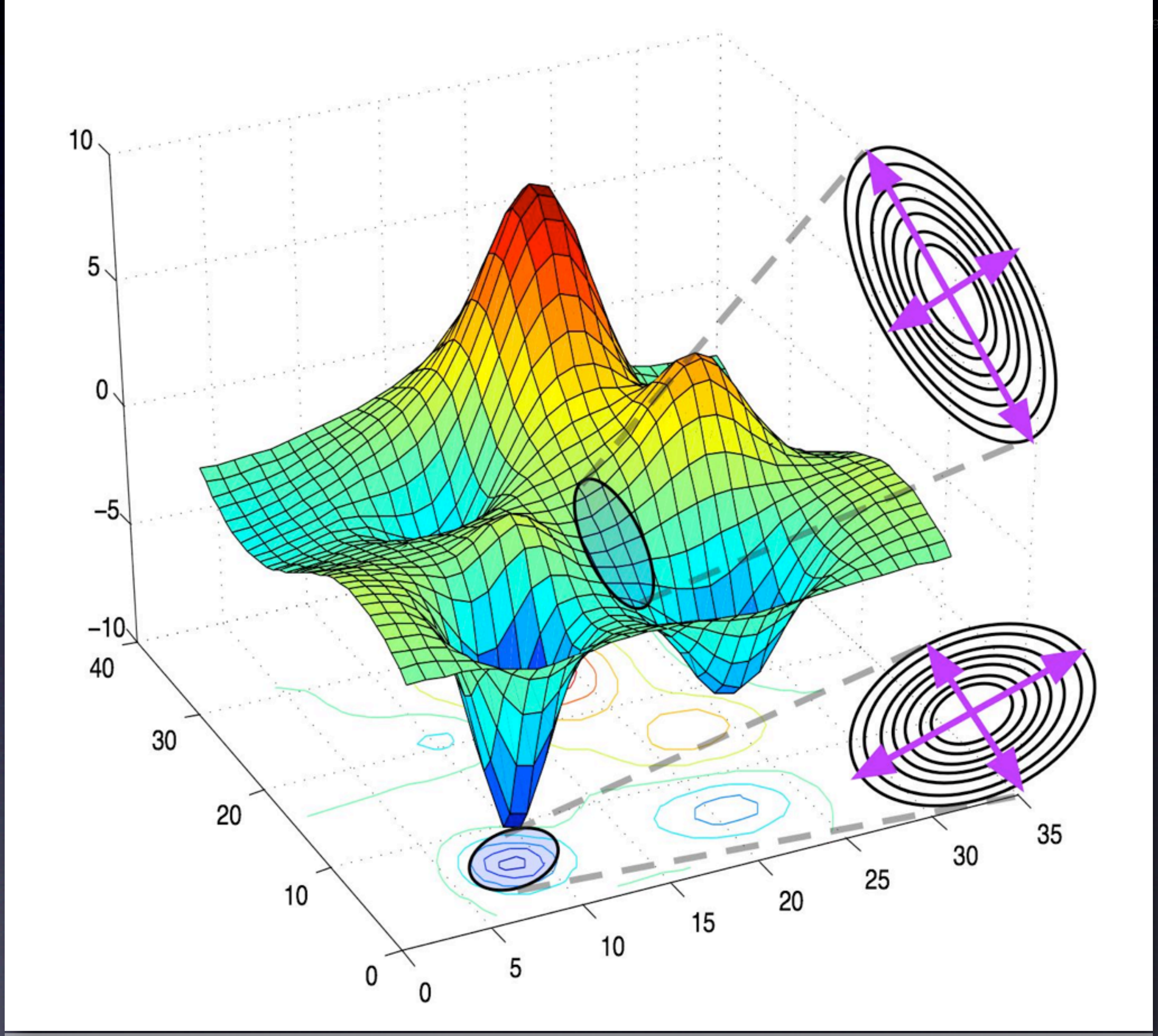
- Monte Carlo simulation
  - No time dependence
  - Requires intelligent moves
- Newton’s equations of motion
  - Time dependence
  - Dynamical events
- Stochastic / Brownian dynamics
  - Add noise to forces



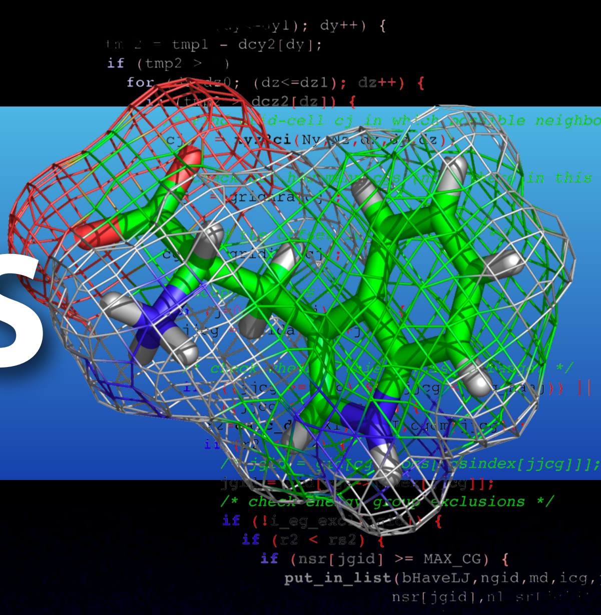
Energy landscapes



- 3N-dimensional space
- Native structure is the free energy minimum
- Ideally, we would sample all of phase space exhaustively
- In practice we have to make do with the most populated parts



Molecular Dynamics


if (tmp2 &gt;
for
/* check the group exclusions */
if (nsr[jgid] &gt;= MAX_CG) {
put_in_list(bHaveLJ,ngid,md,icg,
nsr[jgid],nl

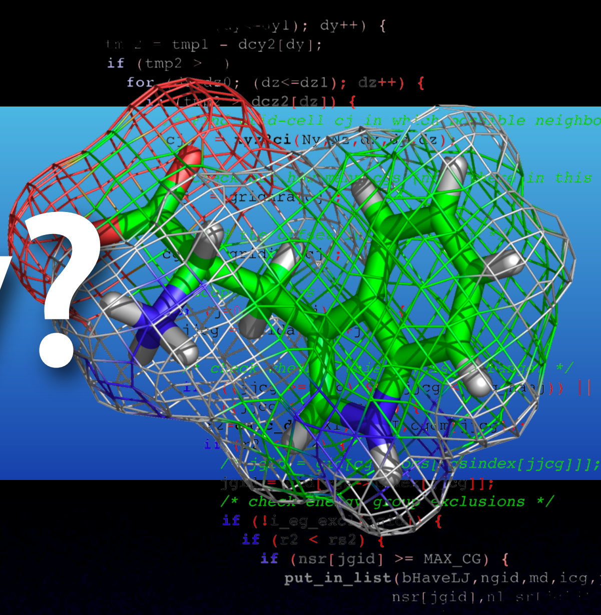
- Solve Newton's equations of motion:

$$m_i \frac{\partial^2 r_i}{\partial t^2} = F_i \quad i = 1..N$$

$$F_i = -\frac{\partial V(r)}{\partial r_i}$$

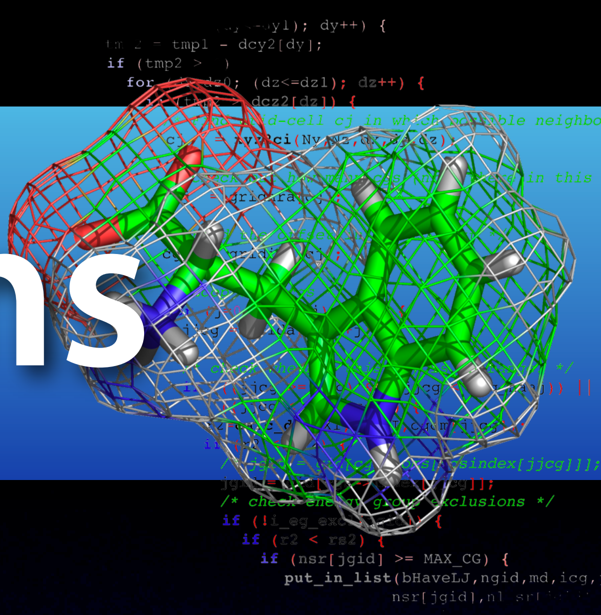
- Timestep has to be small (femtoseconds)
- Forces depend on all particle coordinates in the system (expensive to calculate)
- Generates a system *trajectory* over time

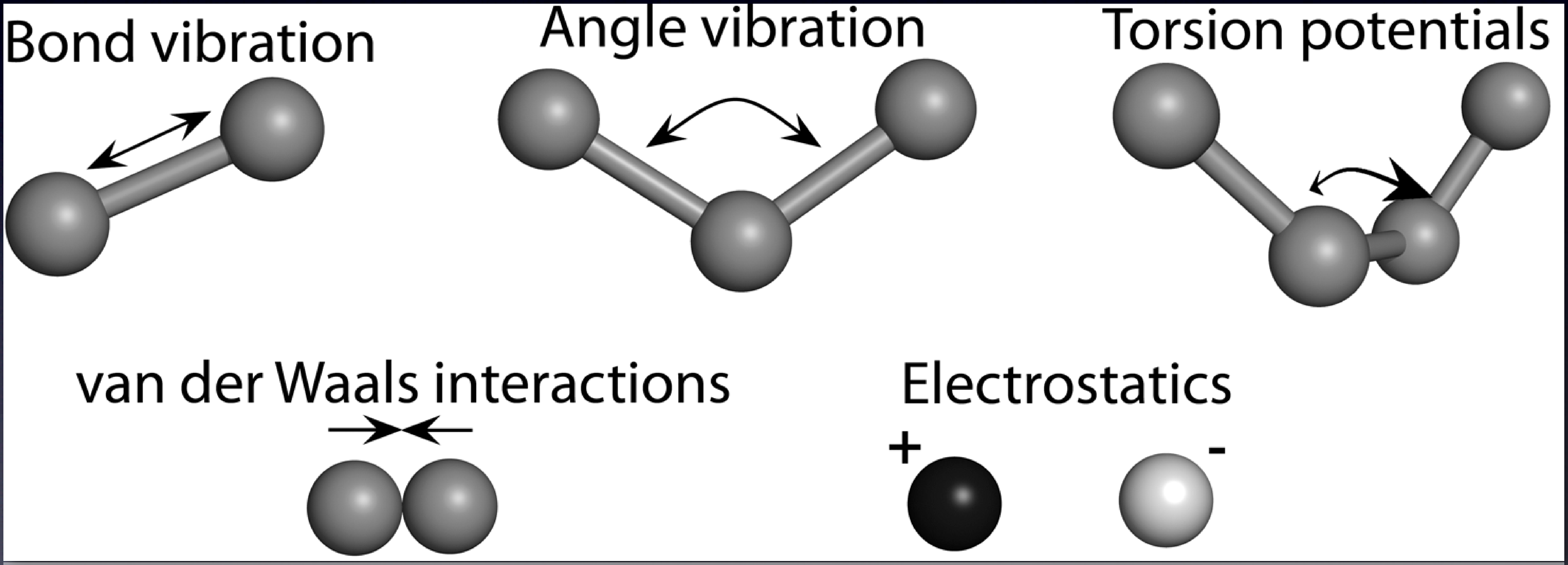
What is a trajectory?



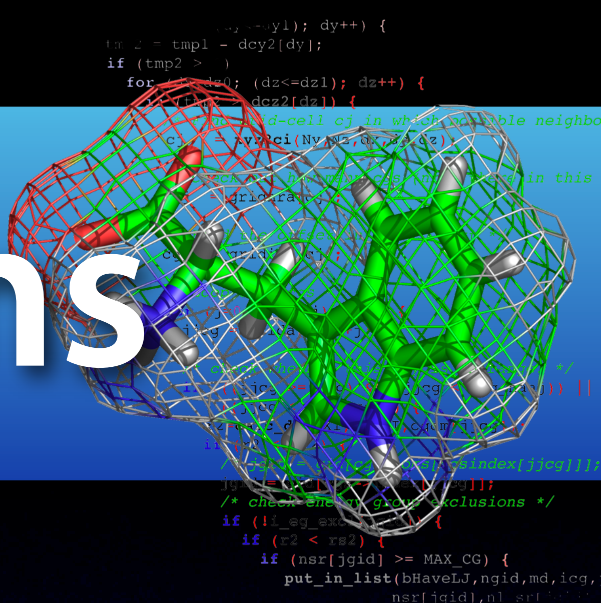
- *NOT* an accurate prediction of the motion of an individual particle
- Molecular dynamics is a chaotic process - differences grow exponentially
- But trajectories aren’t random either?
- *Shadow trajectory:* Simulated path stays close to (some) real path for shorter times

Empirical Interactions

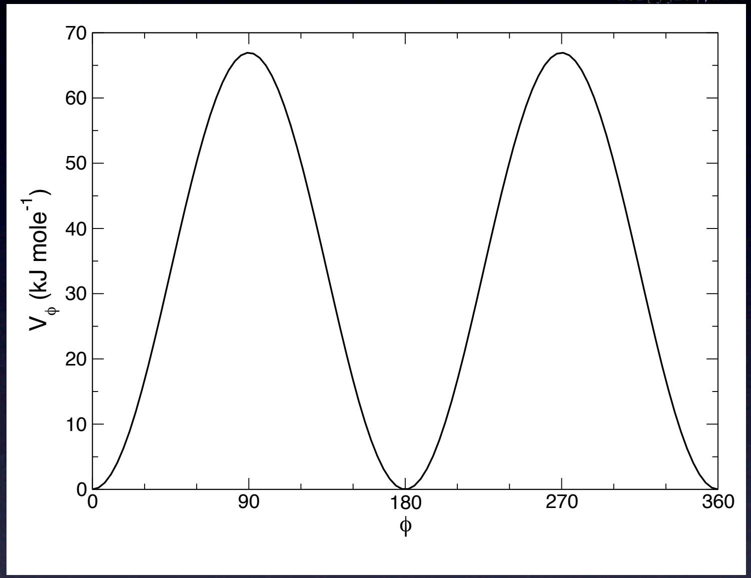


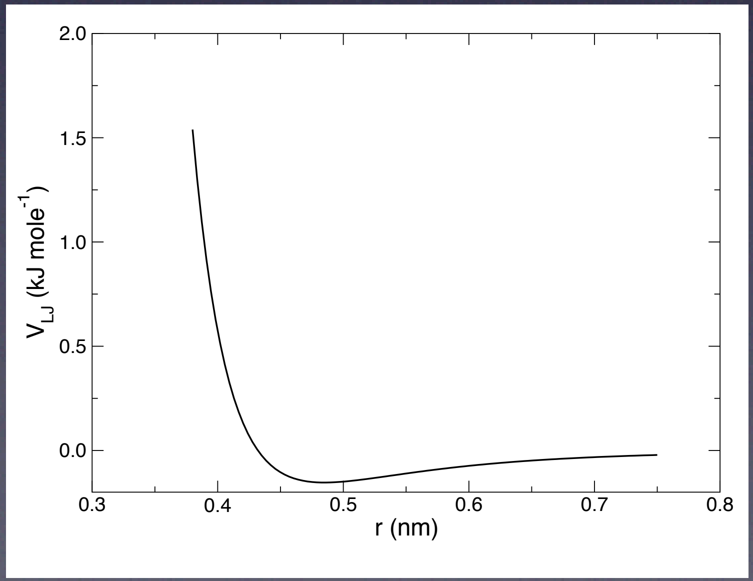


Empirical Interactions



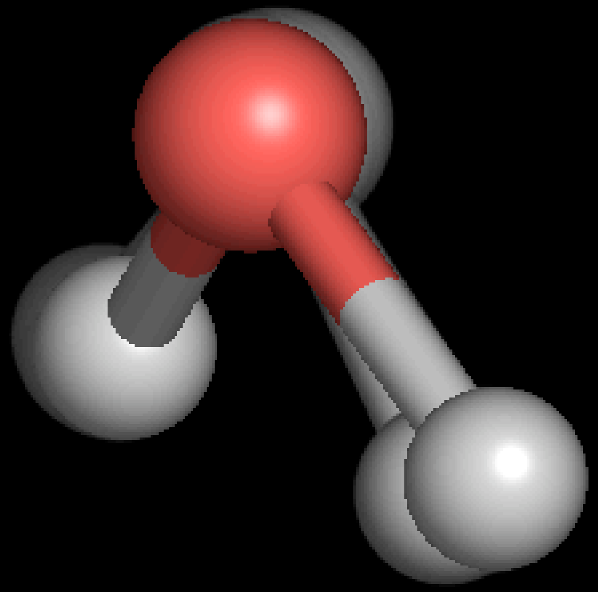
$$
\begin{aligned}
V(r) = & \sum_{bonds} \frac{1}{2} k^{b}_{ij} \left(r_{ij} - r^{0}_{ij}\right)^{2} \\
& + \sum_{angles} \frac{1}{2} k^{\theta}_{ijk} \left(\theta_{ijk} - \theta^{0}_{ijk}\right)^{2} \\
& + \sum_{torsions} \left\{ \sum_{n} k_{\phi} \left[1 + \cos\left(n\phi - \phi_{0}\right)\right] \right\} \\
& + \sum_{impropers} k_{\xi} \left(\xi_{ijkl} - \xi^{0}_{ijkl}\right) \\
& + \sum_{i,j} \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} \\
& + \sum_{i,j} \left[\frac{C_{12}}{r^{12}_{ij}} - \frac{C_6}{r^{6}_{ij}}\right]
\end{aligned}
$$

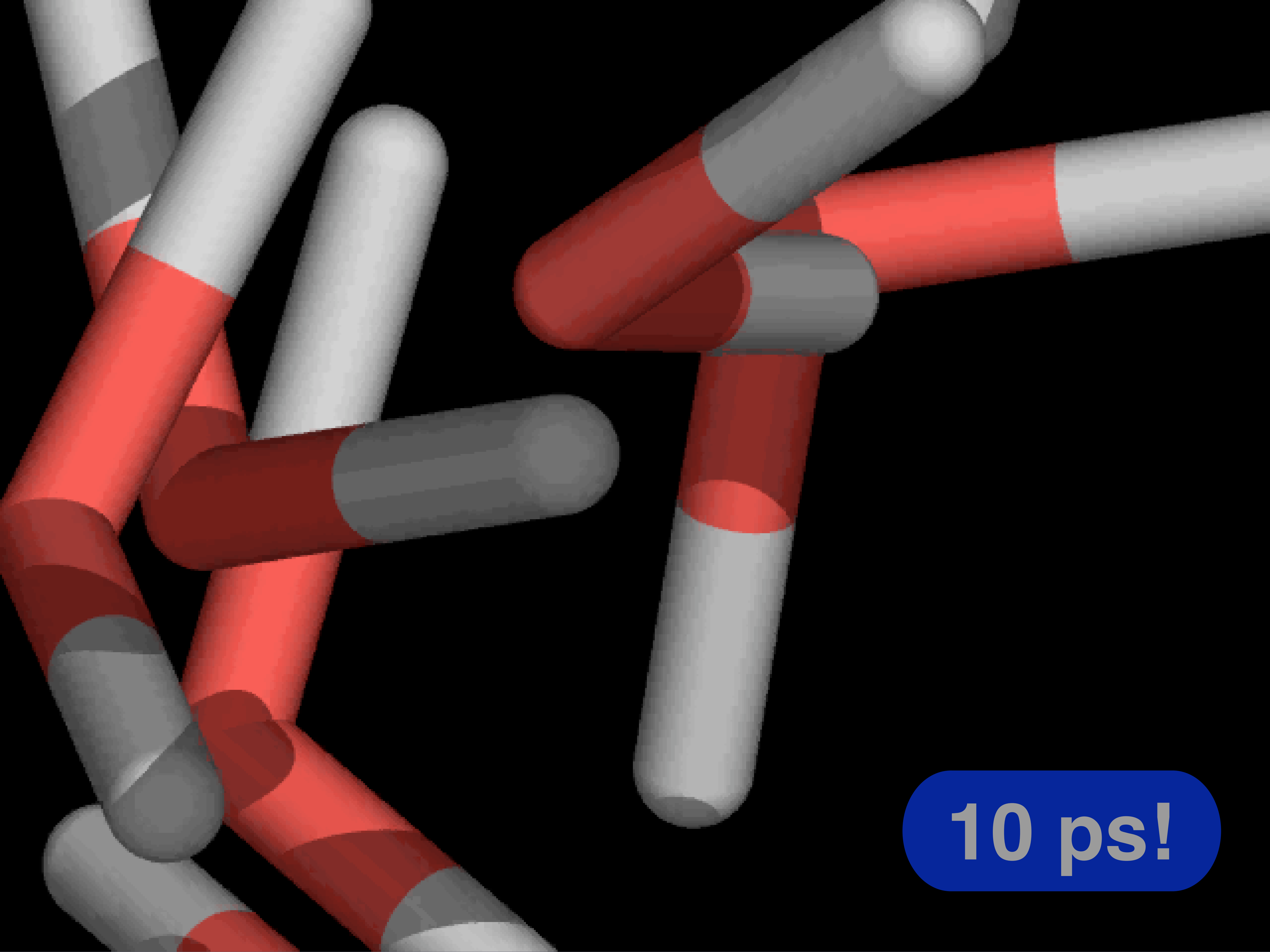




Taking a timestep

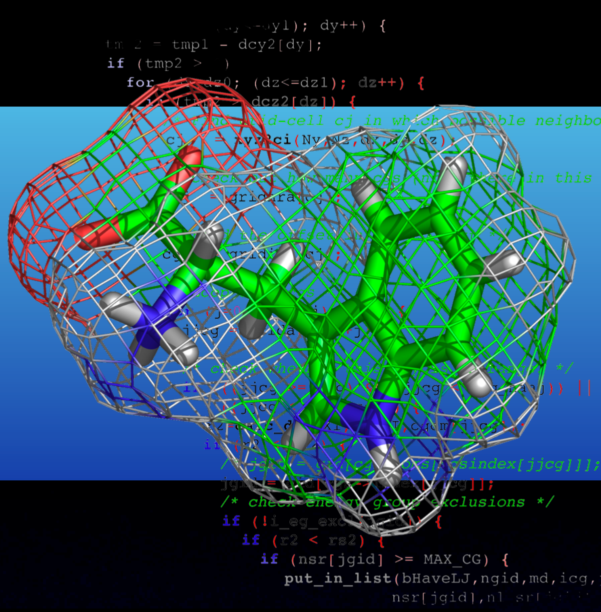
8 fs

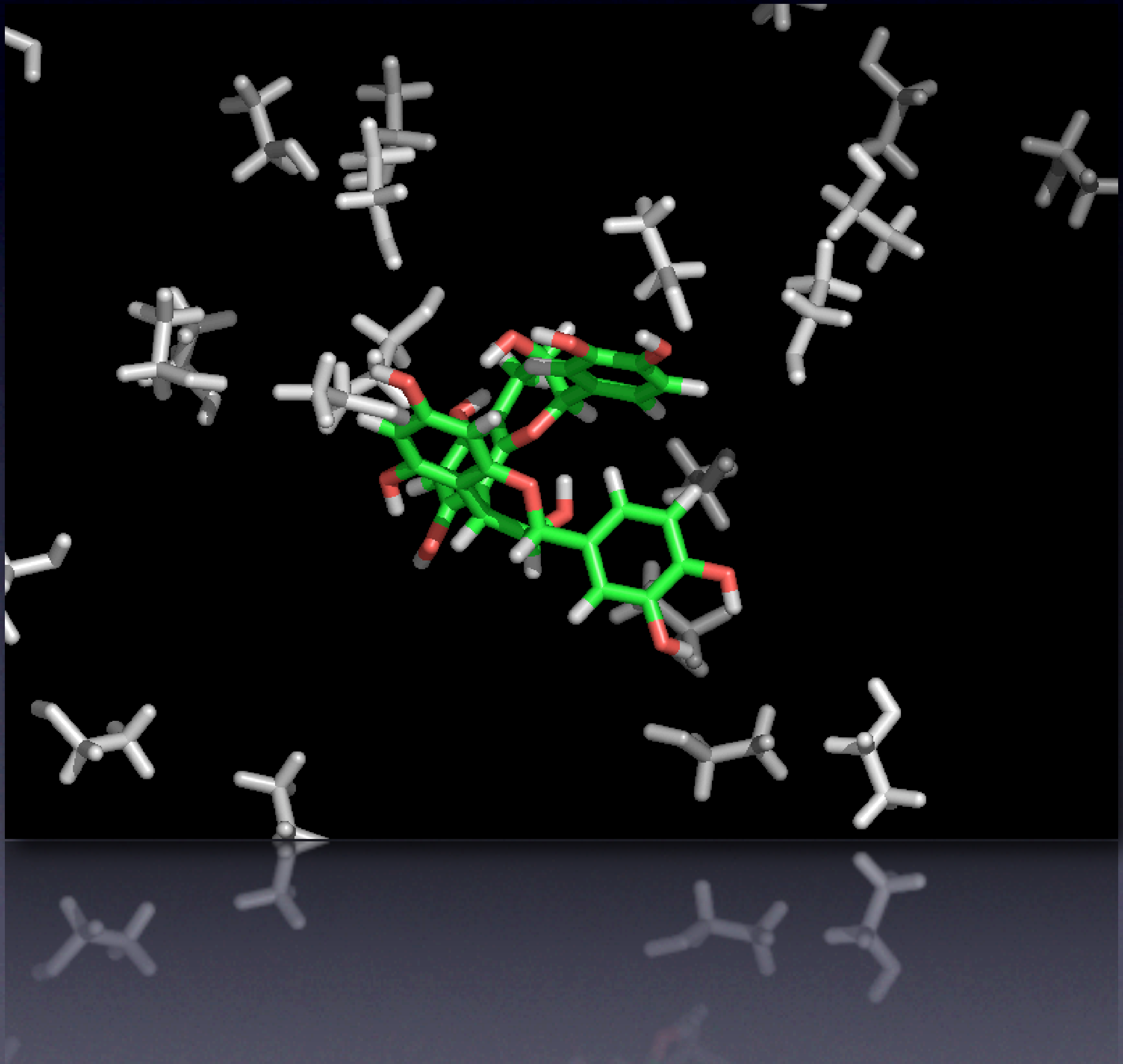




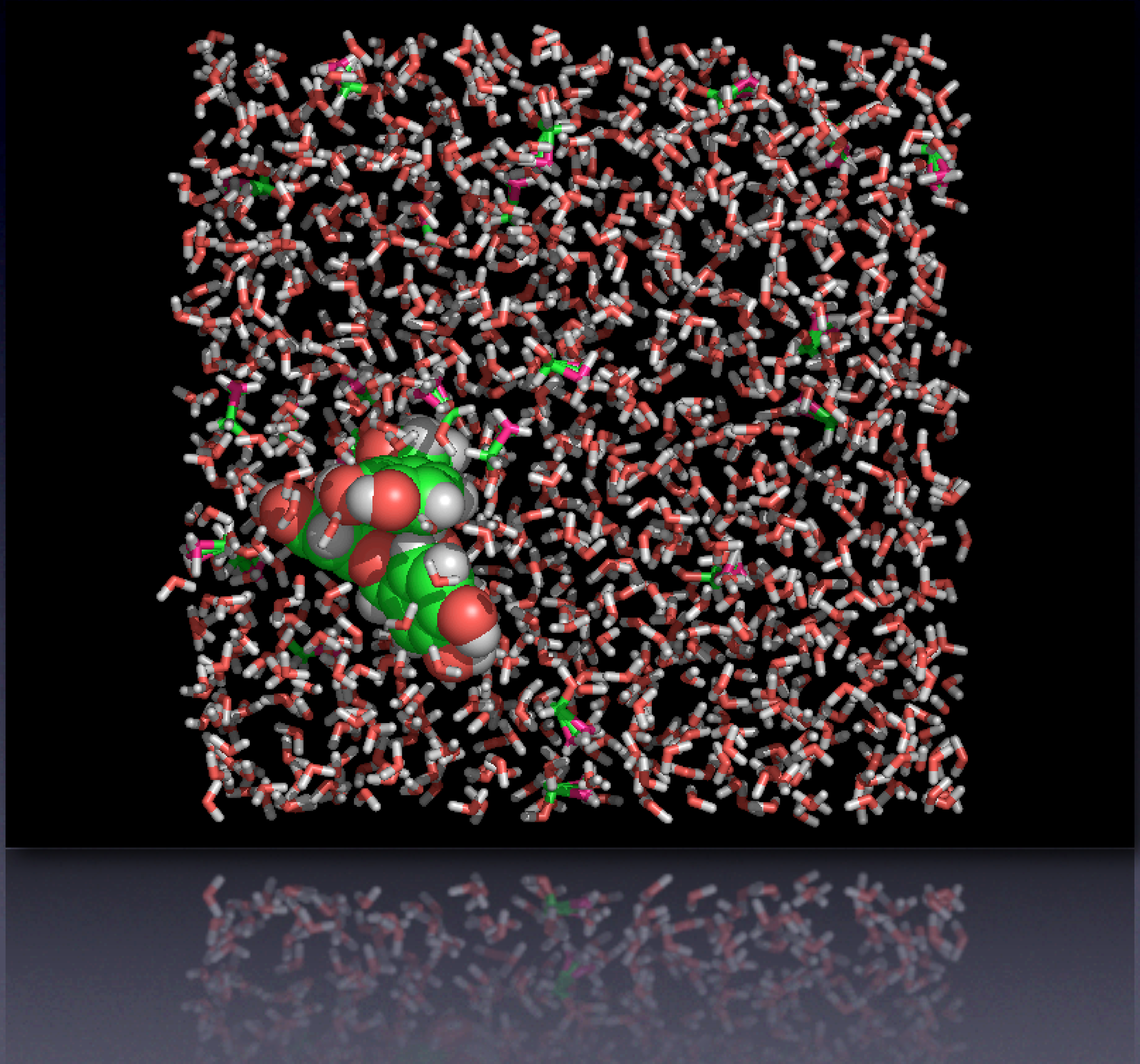
10 ps!

Environments



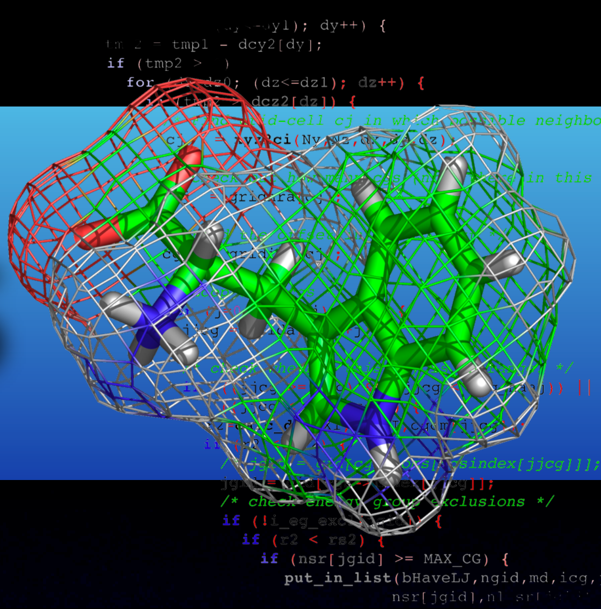


Unrealistic

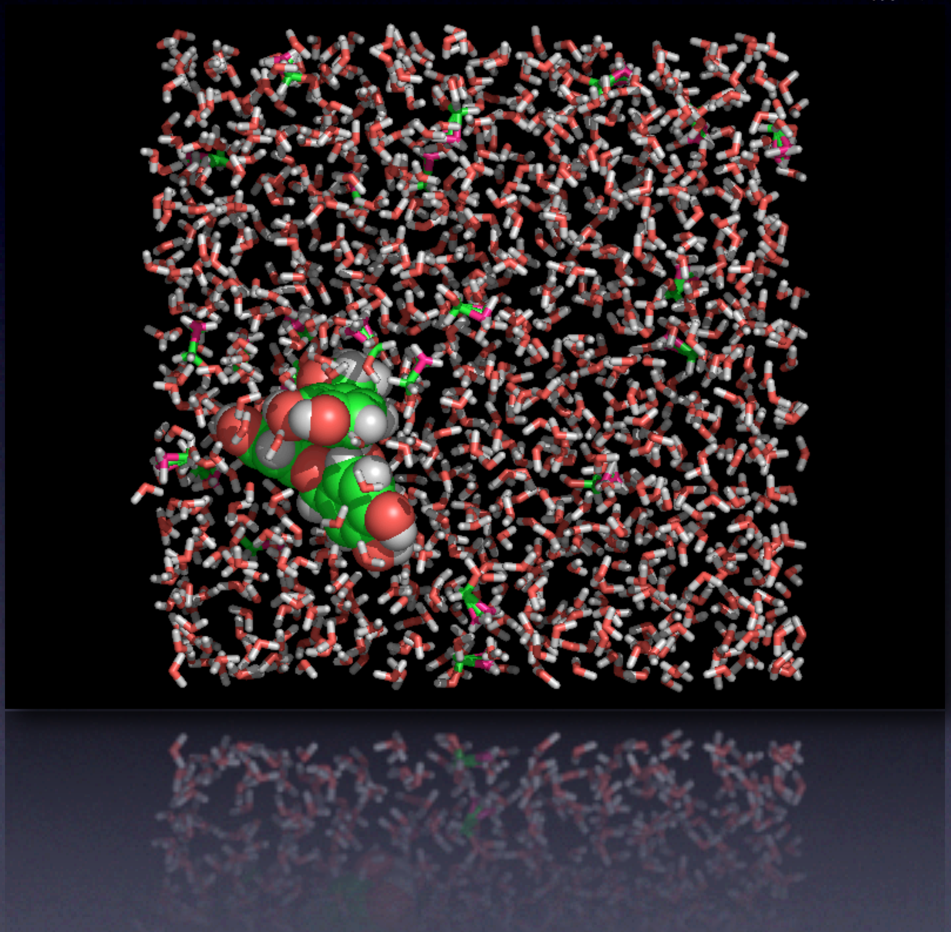


Realistic

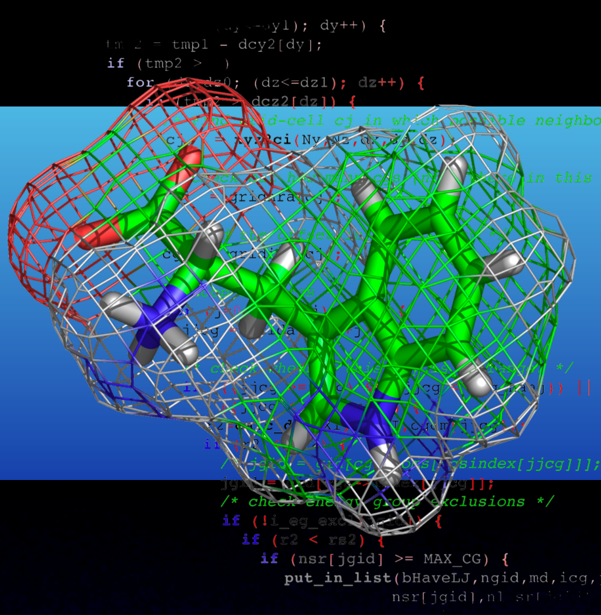
Boundary conditions



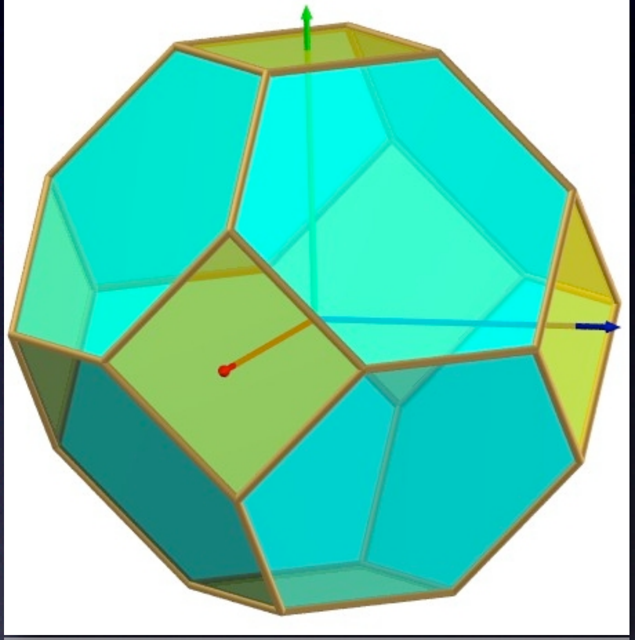
- Vacuum: No solvent
- Droplet: Spherical layer of water around protein, solvent molecules constrained with random forces (not in Gromacs)
- Periodic Boundary Conditions (PBC) - a water that exits on the right reappears on the left



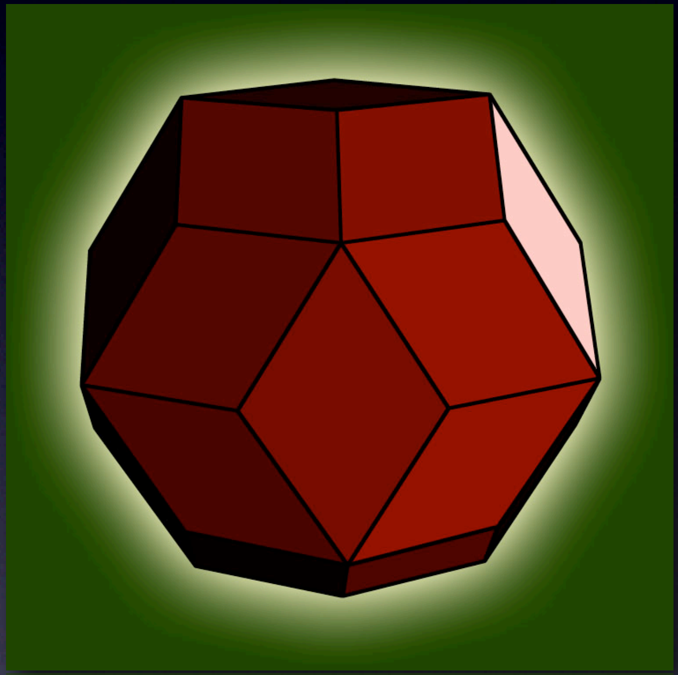
Periodic cell shapes



- Cubic / rectangular
- Truncated octahedron (more spherical)
- Rhombic dodecahedron (most spherical cell)
- Octahedron volume is only 77% of cube, and dodecahedron 71% at same periodic distance!



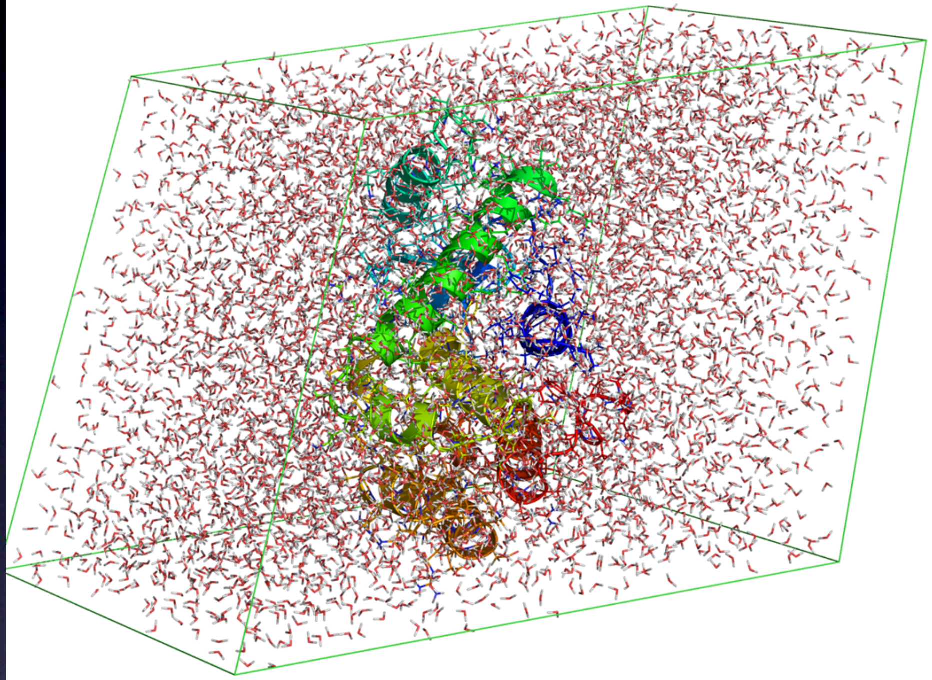
Truncated octahedron

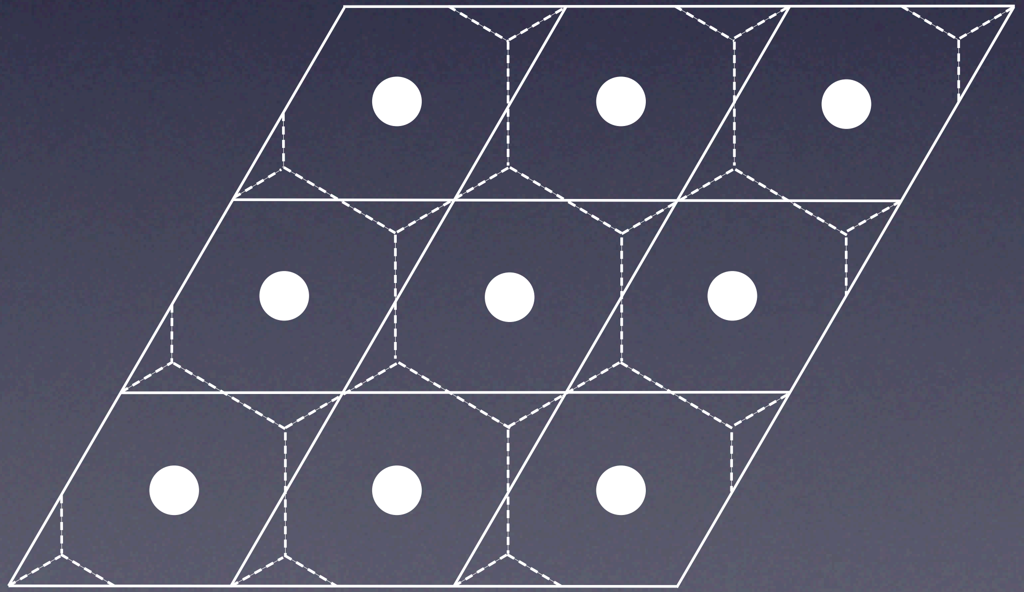


Rhombic dodecahedron

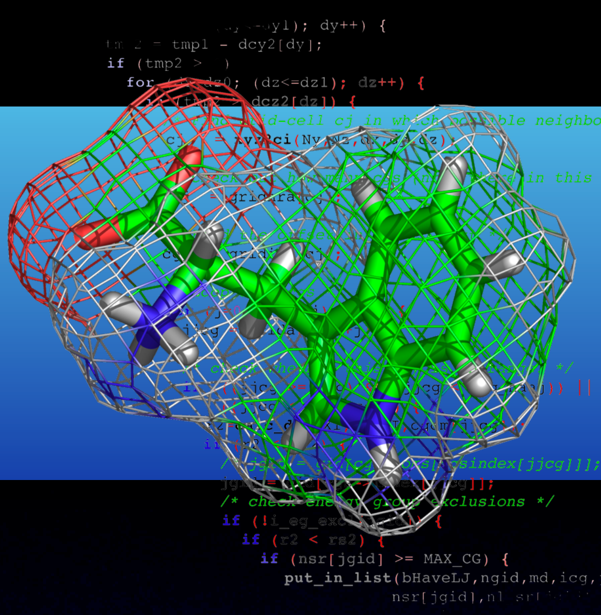
Triclinic cells

- Gromacs uses triclinic cells internally
- These can represent any of the cells mentioned, and also e.g. hexagonal boxes
- Not trivial to see in 3D, but see the hexagonal example in 2D to the right!

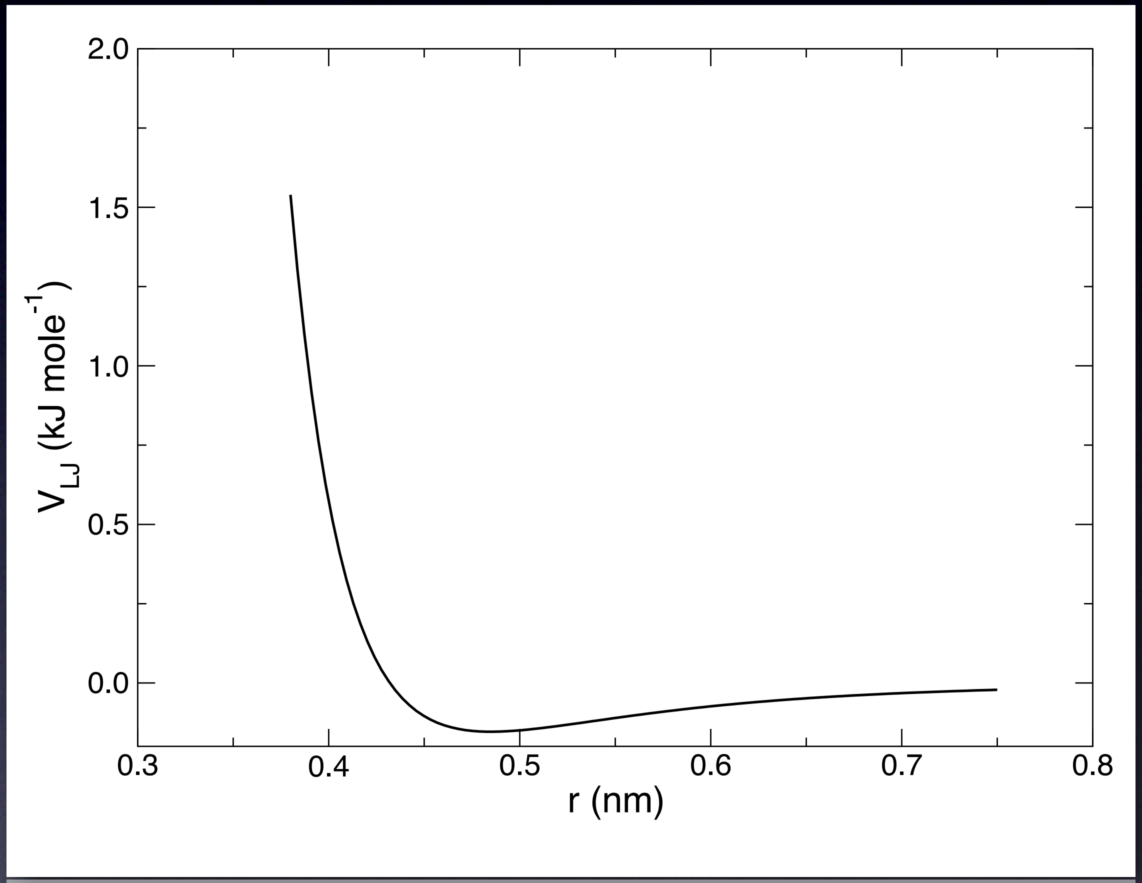




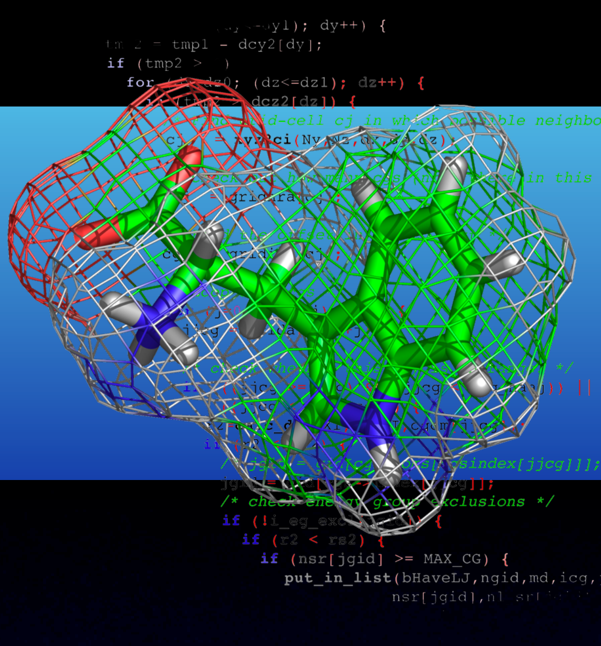
Lennard-Jones



- Interactions between all particles in a system
- Long range dipole dispersions: $1/r^6$
- Short range repulsion: Approx. $1/r^{12}$ (Really exp)
- Decays fast - cutoff fine



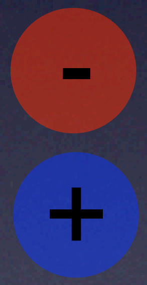
$$\frac{C_{12}}{r_{ij}^{12}} - \frac{C_6}{r_{ij}^{6}} \quad \text{or} \quad 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

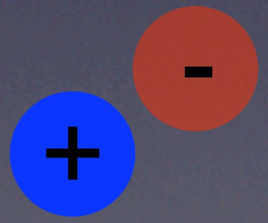


Electrostatics

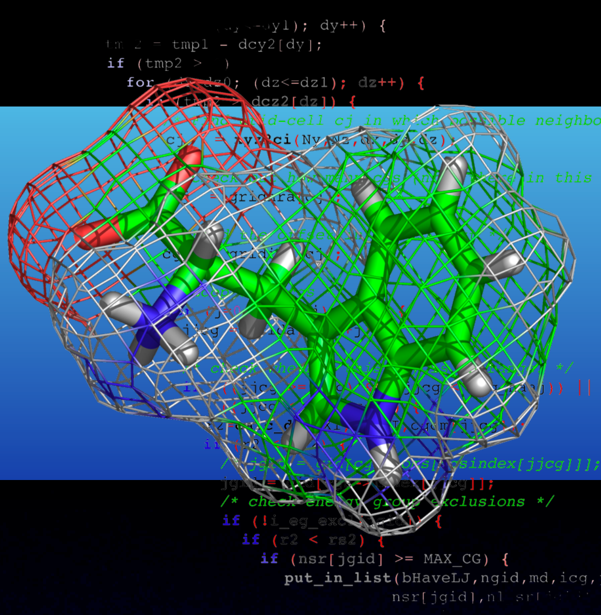
- Interactions between charged particles
- Strong interaction - sensitive to partial charges!
- Decays very slowly; cut-offs do not even always converge!
- Charge groups can be used to guarantee convergence

$$\frac{1}{4\pi\varepsilon_0}\frac{q_i q_j}{r_{ij}}$$

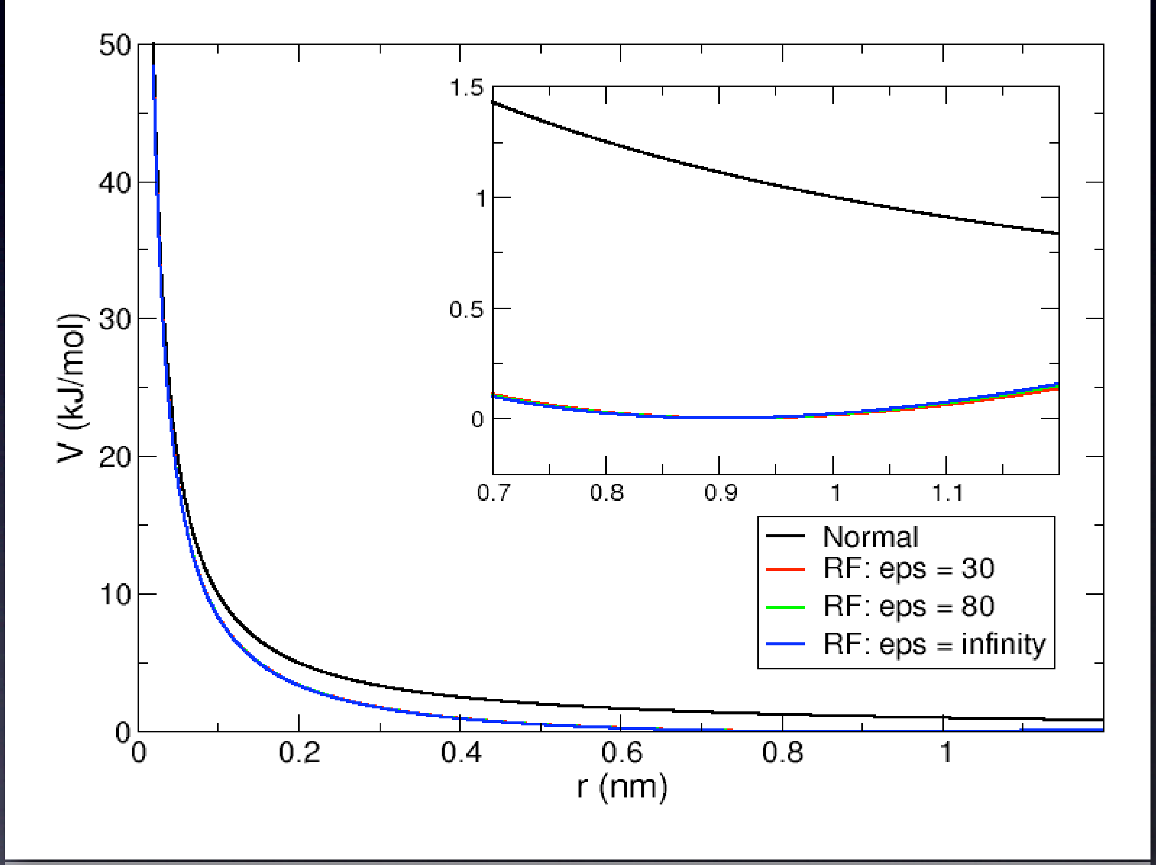




Reaction-field

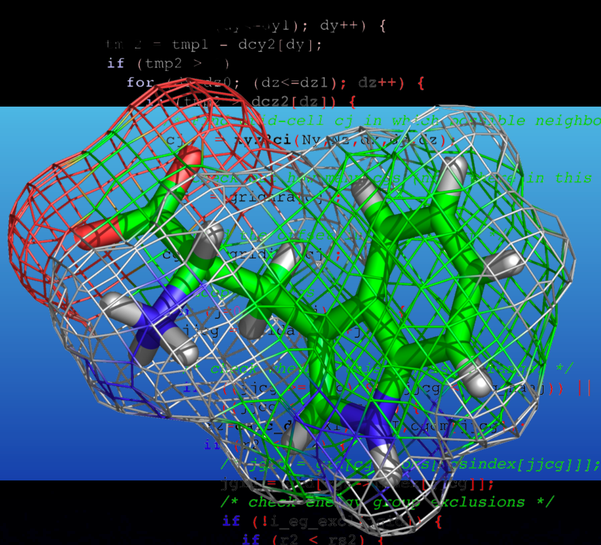


- Assume interactions are damped by ε_r beyond cut-off
- Homogeneous & isotropic
- Really cheap to calculate
- Good for dipoles, not ions

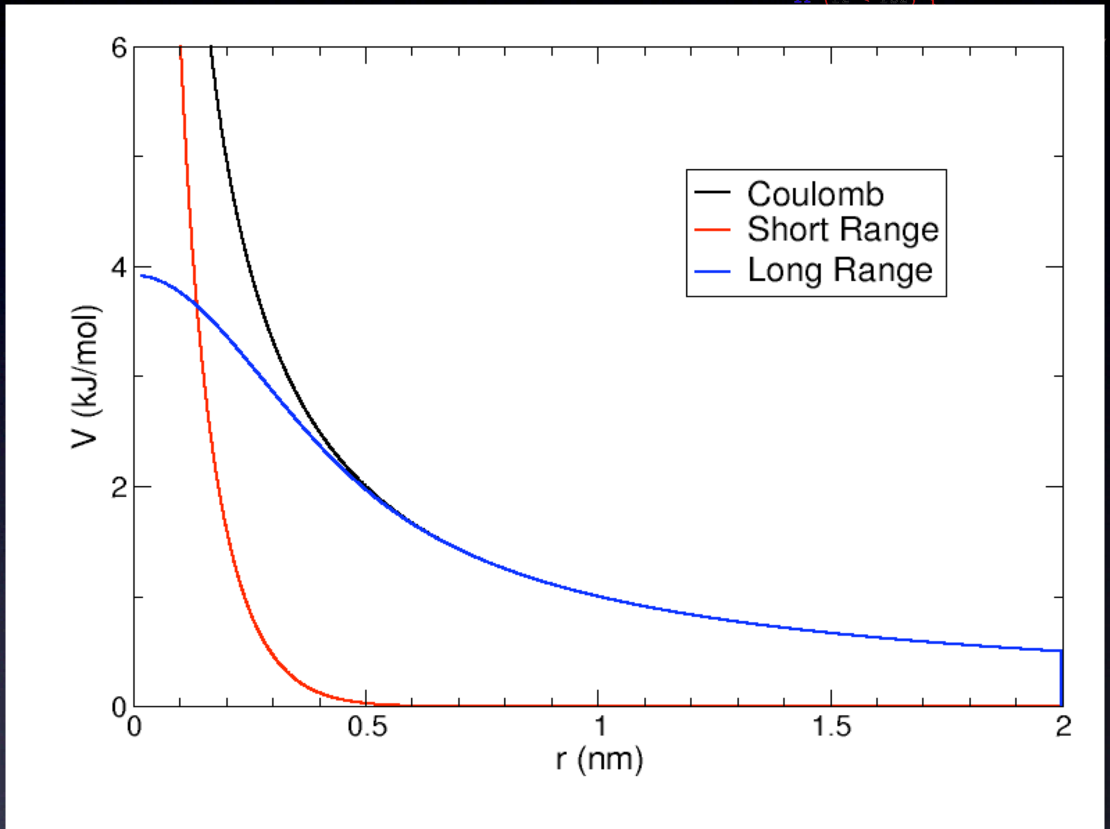


$$V = V_{coulomb} * \left(1 + \frac{\varepsilon_{rf}-1}{2\varepsilon_{rf}+1}\right)\frac{r^3}{r_{cut}^3}$$

Ewald summation



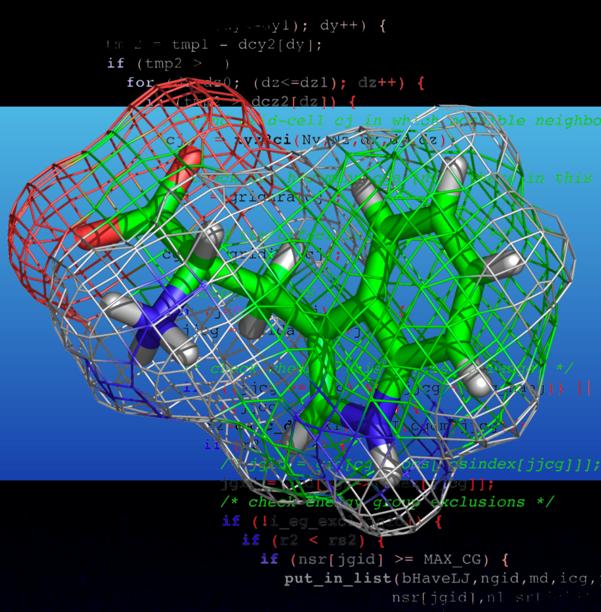
- Separate electrostatics into long & short range parts
- Short range decays fast (cut-off can be applied)
- Long range solved in reciprocal space, just as for crystals - corresponds to sum over the infinite periodic copies!



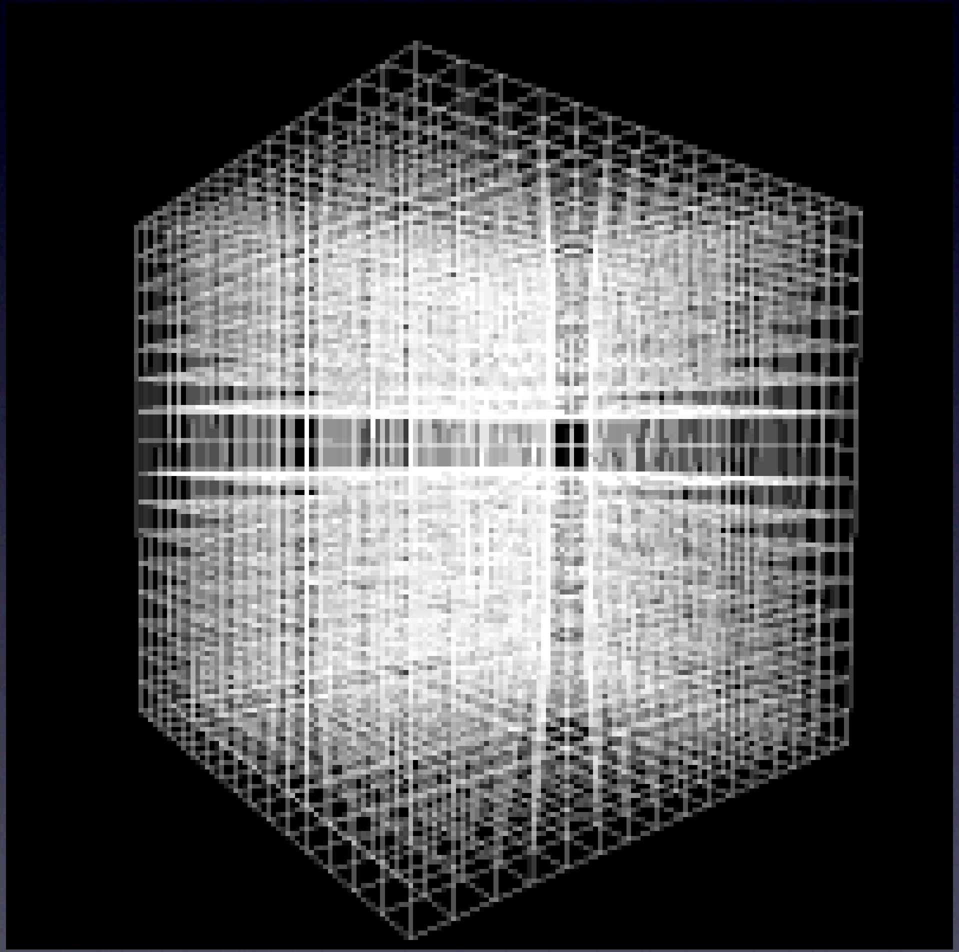
$$V_{sr} = V_{coulomb} * erfc(r/\beta)$$

$$V_{lr} = V_{coulomb} * (1 - erfc(r/\beta))$$

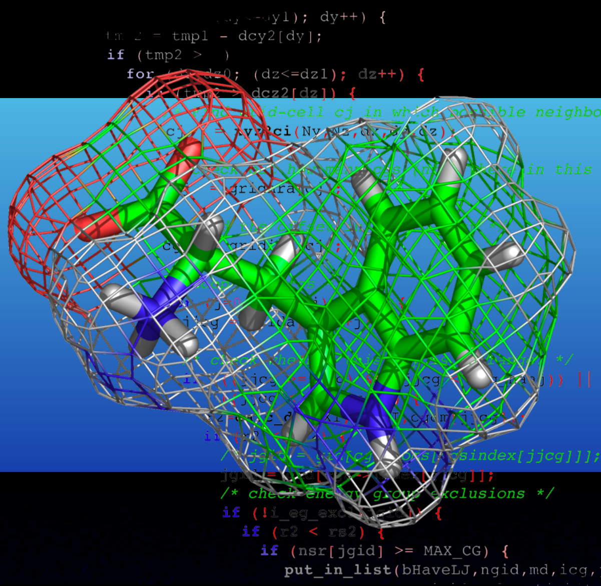
PME

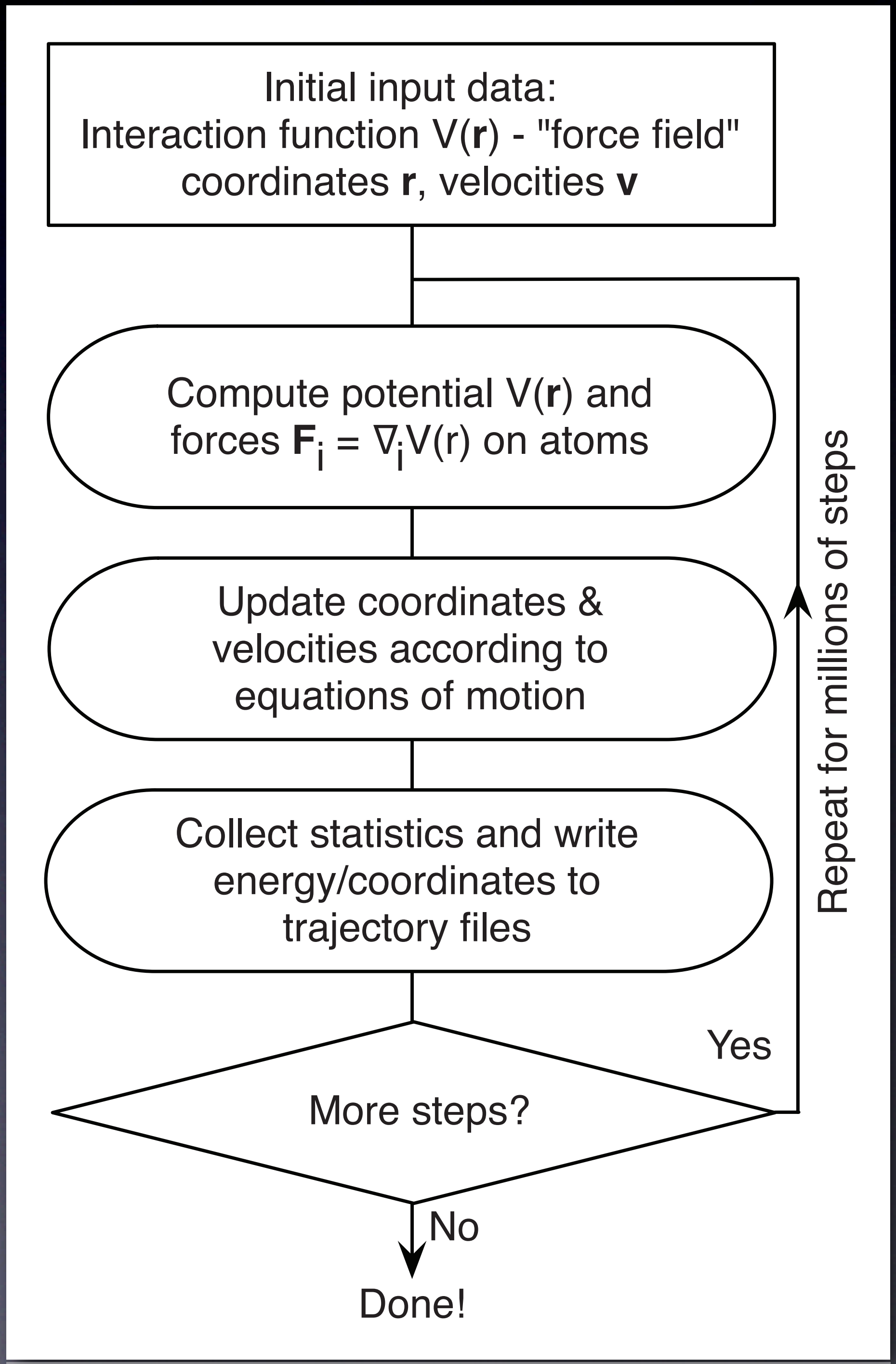


- Particle-Mesh Ewald
- Modern, fast, way to perform Ewald summation
- Charge spreading on 3D grid
- Solve Poisson eq. on grid (convolution in recprocal space)
- Transform back, interpolate to get forces



MD algorithm





Initial input data:
Interaction function V(**r**) - "force field"
coordinates **r**, velocities **v**

Compute potential V(**r**) and forces $\mathbf{F}_i = \nabla_i V(r)$ on atoms

Update coordinates & velocities according to equations of motion

Collect statistics and write energy/coordinates to trajectory files

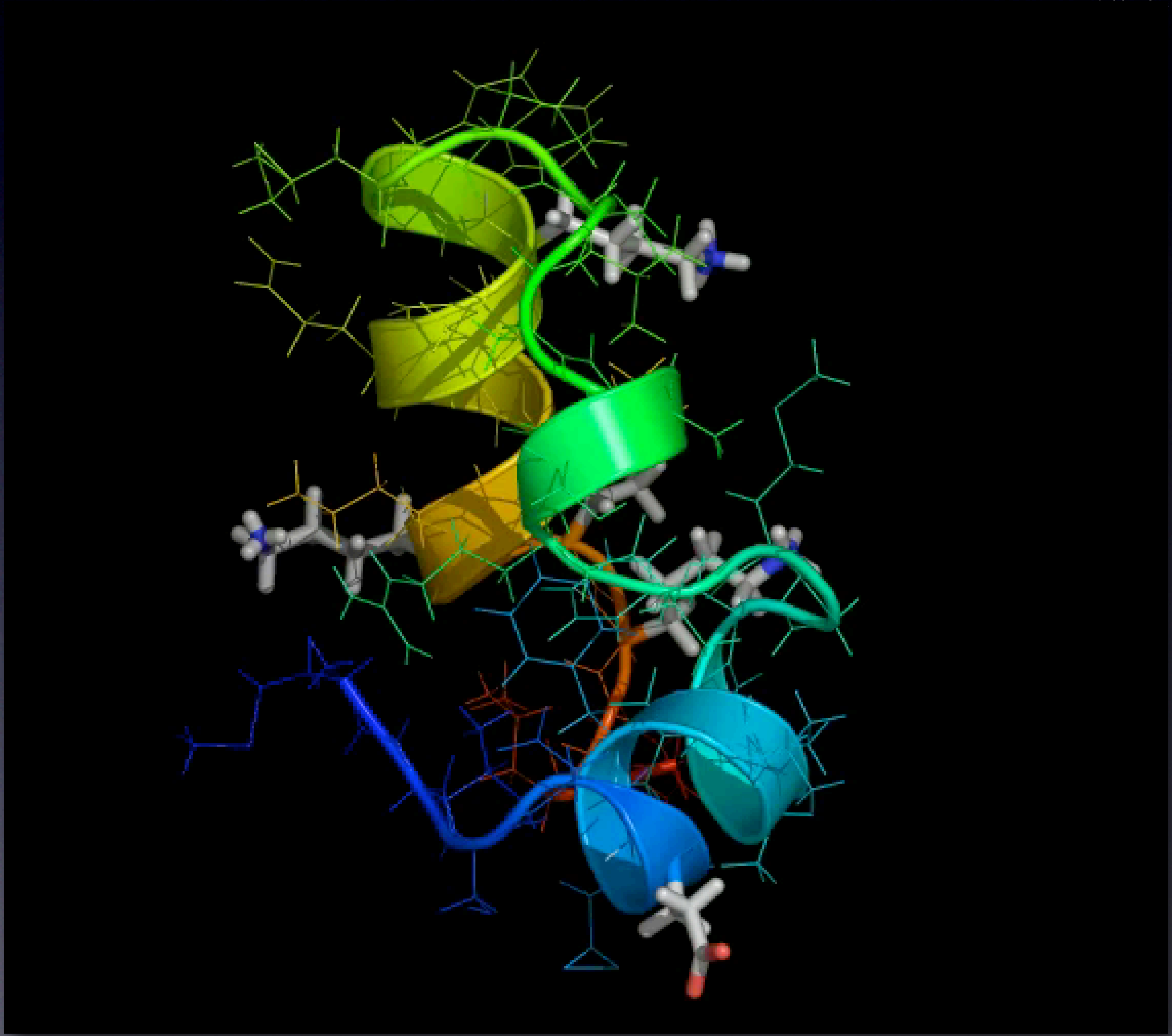
More steps?

Yes

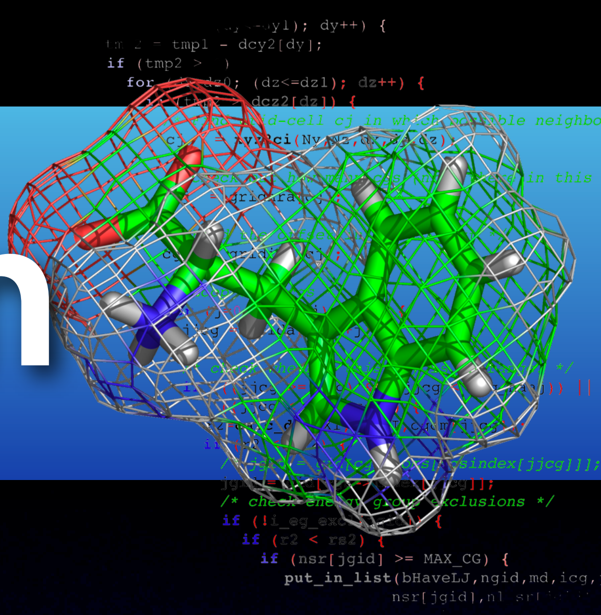
No

Repeat for millions of steps

Done!

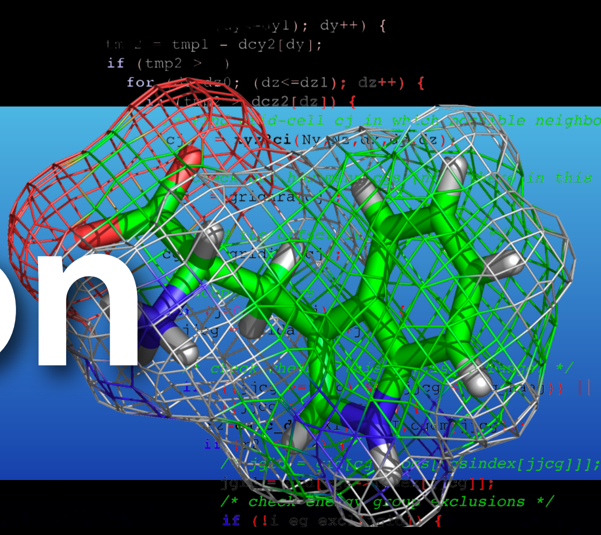


A typical simulation

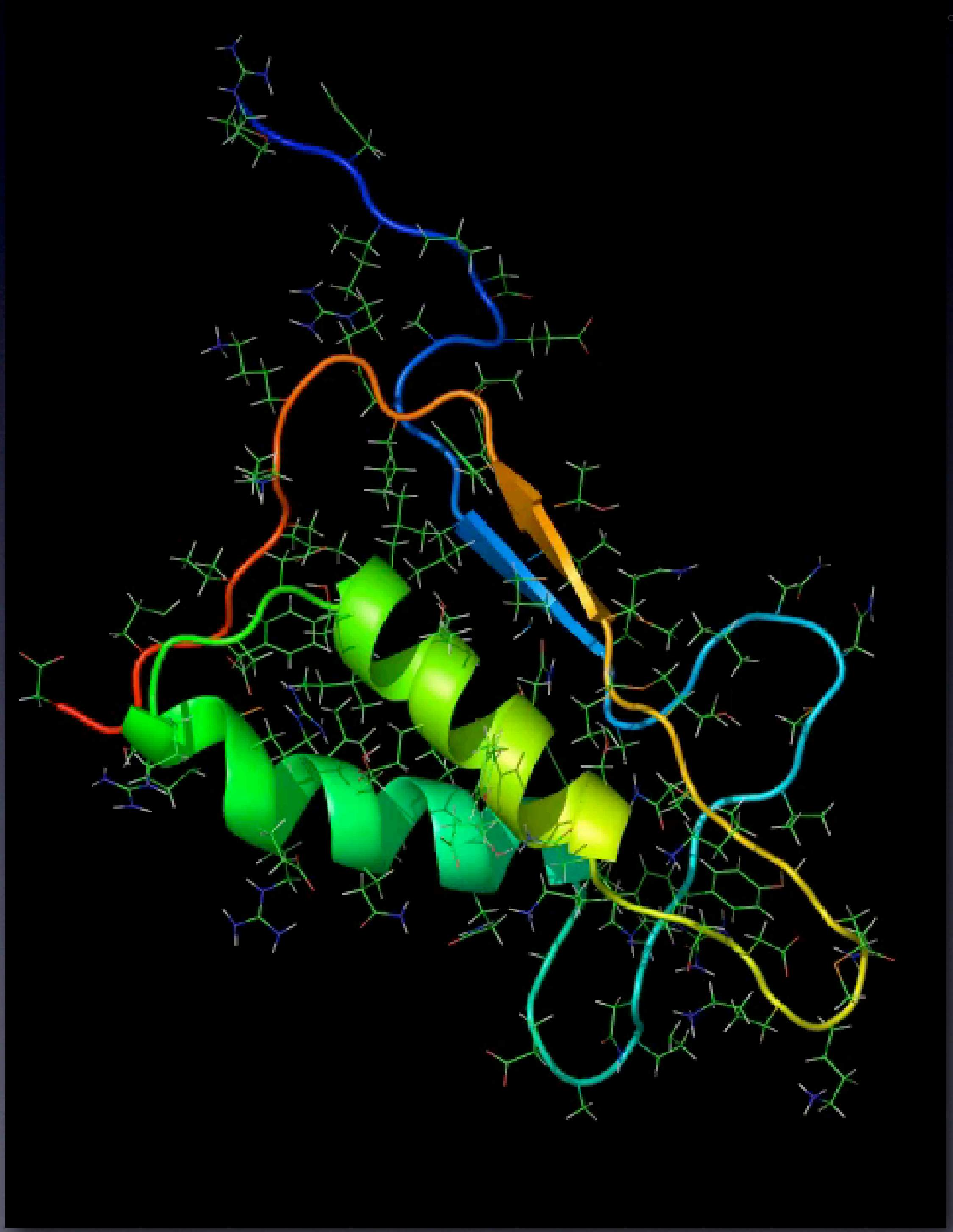


- Get a structure (usually from PDB)
- Fix missing segments, sidechains, etc.
- Prepare a Gromacs topology (parameters)
- Add solvent water
- Energy minimization
- Equilibration simulation
- Run production simulation
- Analyze the output trajectory data

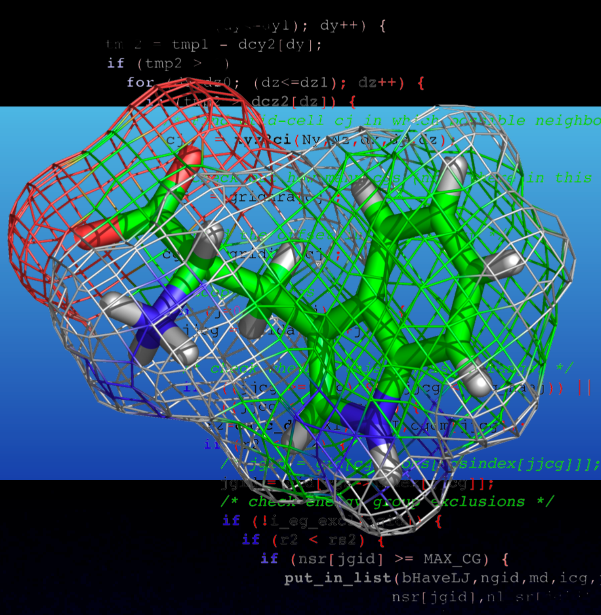
Energy Minimization



- Find local energy minima
- ...or rather: avoid maxima
- Forces are gradient of V
- Example algorithms:
  Steepest descent
  Conjugate gradients
  L-BFGS
- Normally small changes

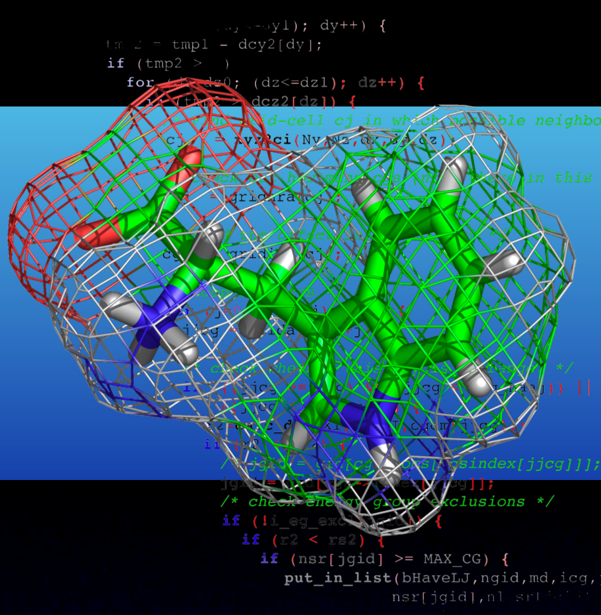


Ensembles



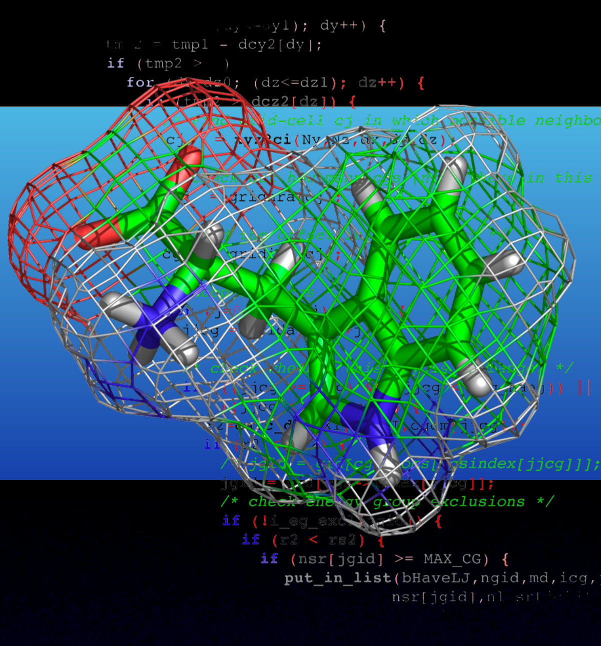
- How do we reproduce experimental conditions like temperature & pressure?
  - Temperature = Kinetic energy
    Set from Maxwell distribution
  - Has to be controlled during simulation, since we lose energy, e.g. due to cut-offs
  - Berk will cover this in detail next hour!
- Pressure: Control by adjusting cell size
- Chemical potential: Add/remove particles

Water models



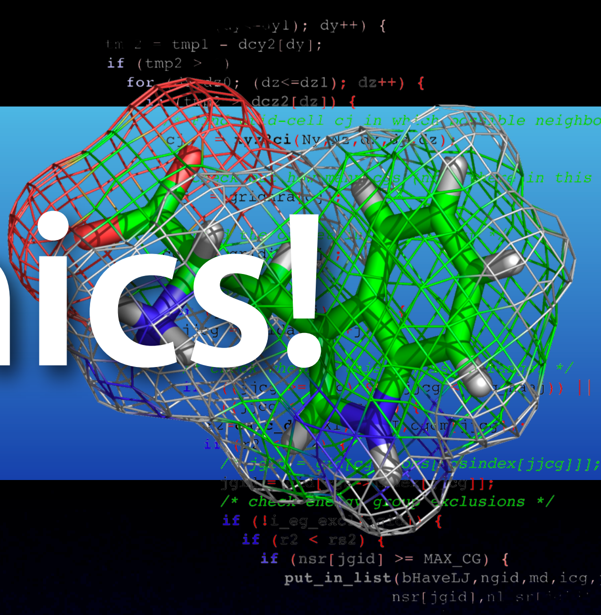
- To speed up simulation we normally use simplified models for water
- Assume water molecules to be rigid to avoid hydrogen vibrations
- Common models: SPC, TIP3P, TIP4P
- There is a wealth of advanced models when you are more interested in water properties *per se* (also polarization)

Limitations of MD



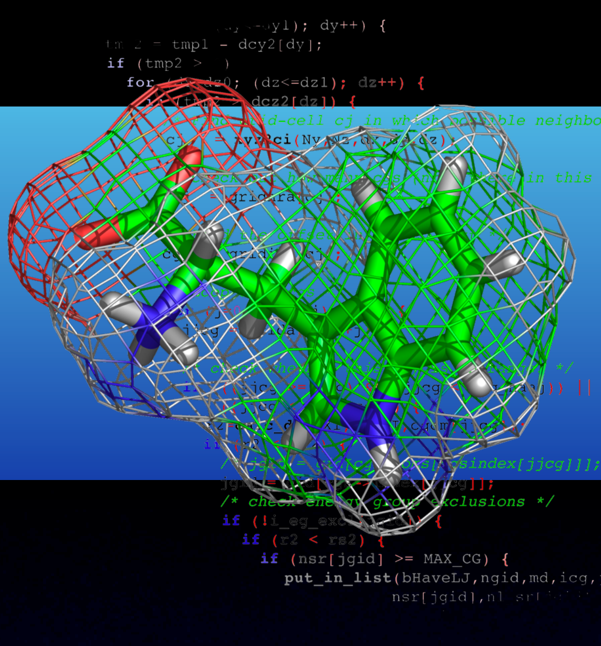
- Parameters are imperfect
- Phase space is not sampled exhaustively
- Example: Free energies of solvation for amino acids often have errors ~1kJ/mol
  - Likely impossible to calculate binding free energies more accurately than this
- Limited polarization effects; waters can reorient, but partial charges are fixed

It’s statistical mechanics!



- Remember:
  Just because you see something in a simulation does NOT mean it is real
- Everything is about statistics
- When you’ve seen it 10 times it’s significant - a single event is not!
- You should always try to calculate error estimates for predicted properties

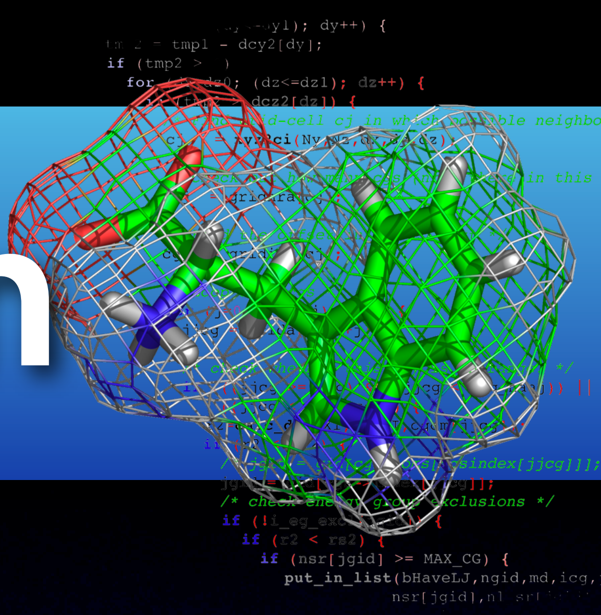
MD Programs



```
                    y1); dy++) {
tmp2 = tmp1 - dcy2[dy];
if (tmp2 >  )
  for (dz=dz0; (dz<=dz1); dz++) {
    if (tmp2 > dcz2[dz]) {
            /* check energy group exclusions */
            if (!i_eg_excl
              if (r2 < rs2) {
                if (nsr[jgid] >= MAX_CG) {
                  put_in_list(bHaveLJ,ngid,md,icg,
                        nsr[jgid],nl
```

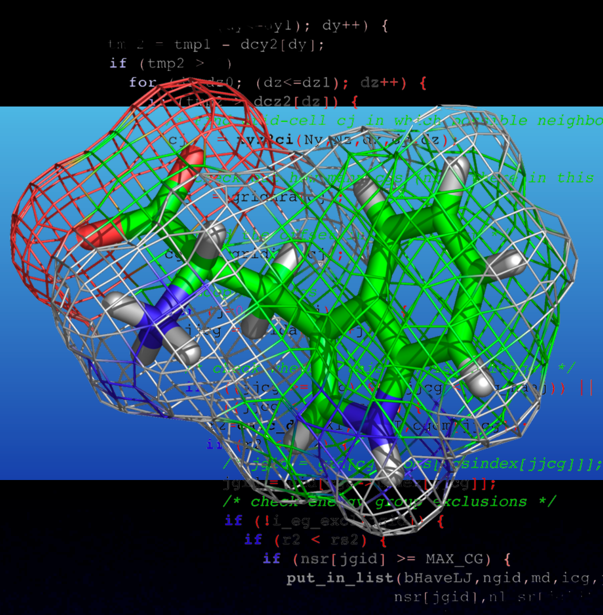
- GROMACS, Amber, Charmm, NAMD, ESPRESSO, Encad, BOSS, LAMMPS
- Programs are often intimately tied to a force field (Amber99, Charmm19, OPLS)
- Force fields supported by Gromacs: GROMOS96, Gromacs, OPLS-AA/L, Amber, Encad, Charmm (beta), etc.
- Try to stick to 1-2 programs, and learn them in detail - you need to motivate your choice of algorithms when publishing!

A typical simulation



- Get a structure (usually from PDB)
- Fix missing segments, sidechains, determine protonation states, etc.
- Prepare a Gromacs topology (parameters)
- Add solvent water
- Energy minimization
- Equilibration simulation
- Run production simulation
- Analyze the output trajectory data

Summary



- Think first, then simulate
- Timescales & limitations
- Empirical classical models
- Interaction forms
- Sampling equilibrium distributions
- Algorithms, approximations, quality
- Flowcharts of typical simulations