V10 Molecular dynamics simulations

There are "only" two real problems with MD simulations:

(1) the accuracy of force fields and

(2) sufficient conformational sampling.

Program for today:

- Force fields – no accuracy discussion though
- MD algorithms
- Applications of MD
- Thoughts on statistics

Molecular dynamics (MD) simulations

MD simulations are nowadays a major technique to study biomolecules in terms of

- structure
- dynamics,
- thermodynamics and
- their interaction with other biomolecules.

An MD simulation computes the behavior of a molecular system over time.

Trajectories of MD simulations contain detailed information on the fluctuations and conformational changes of proteins and nucleic acids.

MD simulations are also used by X-ray crystallographers and NMR experimentalists to **determine structures,** i.e. to generate conformations that match the experimental data.

Dynamic processes in biomolecules

Local movements (spatial dimensions 0.01 to 5 Å, time scales 10^{-15} to 10^{-1} s)

- atomic fluctuations
- movements of side chains
- movements of loops

Movements of rigid bodies (1 to 10 Å, 10^{-9} to 1s)

- movements of alpha-helices
- Domaim movements (hinge bending)
- movements of protein subunits

Large scale movements (> 5Å, 10^{-7} to 10^{4} s)

- Helix coil transitions
- Dissociation/association reactions
- Folding and unfolding processes

Molecular mechanics with empirical force fields

There exist **specialized force fields** for

Water (simple molecule, but yields very complicated solution phenomena)
and other solvents such as methanol, $CHCl_3$, etc

Proteins/peptides

Nucleic acids (DNA, RNA)

Phospholipids

Polymers, sugars

Metal atoms, inorganic and bioinorganic substances, …
(metals are difficult to model with force fields due to d-orbitals ->
spatial anisotropy is important for coordination)

Force field energy = steric energy

E_{steric} = covalent (bonded) interactions

+ non-covalent (nonbonded) interactions

= $E_{bonds} + E_{angles} + E_{torsions} + E_{bonds\text{-}angles} + \ldots +$

$E_{van\text{-}der\text{-}Waals} + E_{electrostatic} + E_{h\text{-}bonds} + E_{other} \ldots$

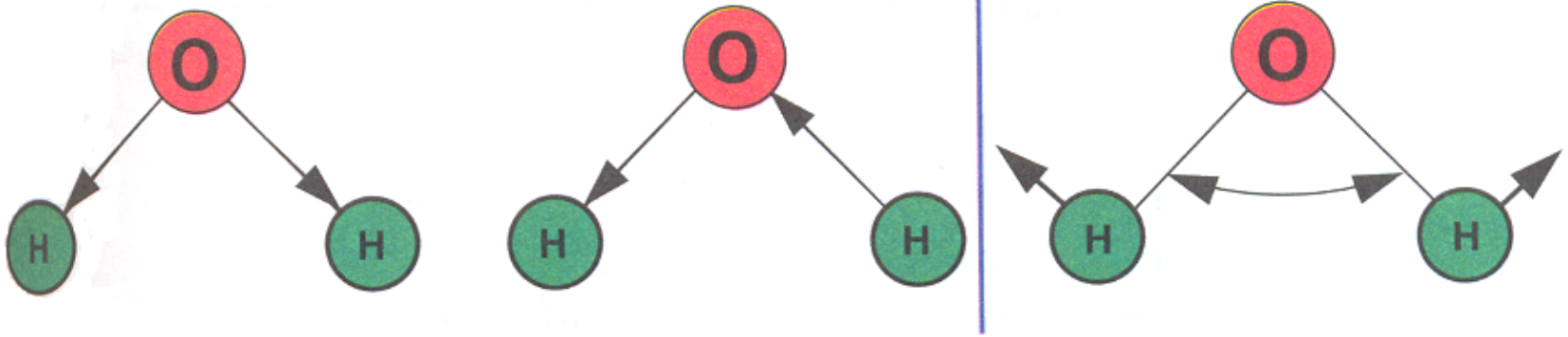
$$E = E_{stretch} + E_{bend} + E_{tors} + E_{vdW} + E_{ES}$$

$$= \sum_{bonds\,(ij)} \frac{k^{(ij)}}{2}\left(r_{ij} - r_0^{(ij)}\right)^2 + \sum_{angles\,(ijk)} \frac{k^{(ijk)}}{2}\left(\phi_{ij} - \phi_0^{(ijk)}\right)^2$$

$$+ \sum_{torsions\,(ijkl)} \frac{k^{(ijkl)}}{2}\left(1 + \cos(n^{(ijkl)}\tau - \tau_0^{(ijkl)}\right)^2$$

$$+ \sum_{pairs\,(ij)} \left(\frac{A_{(ij)}}{r_{ij}^{12}} - \frac{B_{(ij)}}{r_{ij}^{6}} \right) + \frac{1}{4\pi\varepsilon\varepsilon_0} \sum_{pairs\,(ij)} \frac{q_i q_j}{r_{ij}}$$

Bond stretching modes from vibrational spectroscopy



ν_1= 3657 cm^{-1}
symmetric
stretch vibration

ν_2 = 3776 cm^{-1}
asymmetric
stretch vibration

ν_3 = 1595 cm^{-1}
bending vibration

Normal modes of a water molecule [Schlick, Fig. 8.1]

Asymmetric stretch vibration is energetically more demanding than symmetric stretch vibration ($\rightarrow$ higher wave number).

Bending vibration requires least amount of energy.

Bond vibrations are only little coupled to remaining dynamics

$\rightarrow$ freezing bond lengths in simulations (SHAKE, LINCS) allows longer time steps

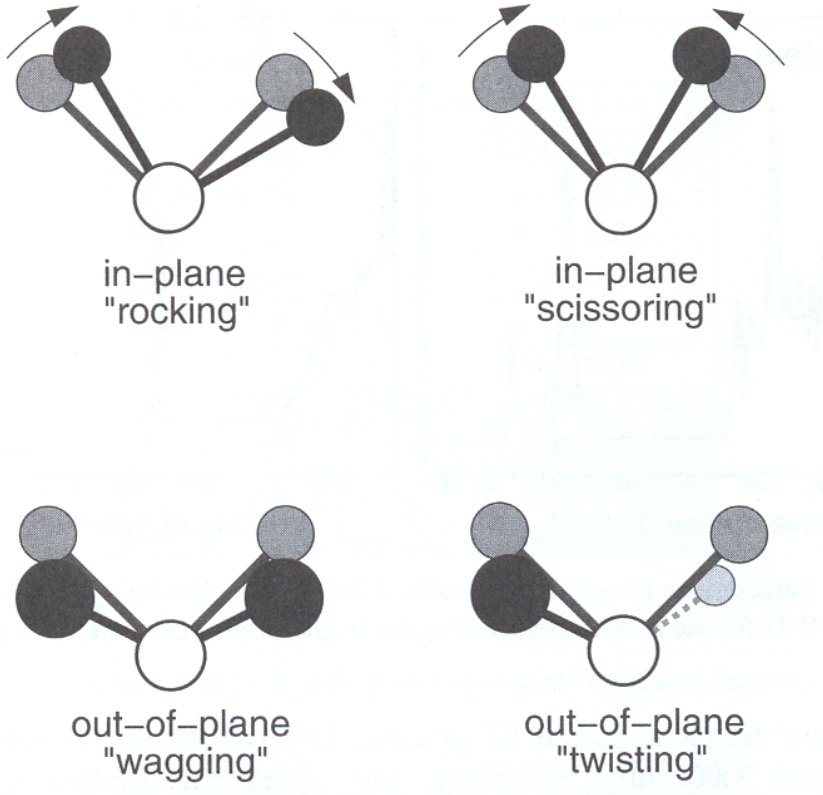
In-plane and out-of-plane bending

2 types of bending vibrations in the plane

2 types of bending vibrations out of the plane

grey atoms: reference position of the molecule.

black atoms: position after displacement.

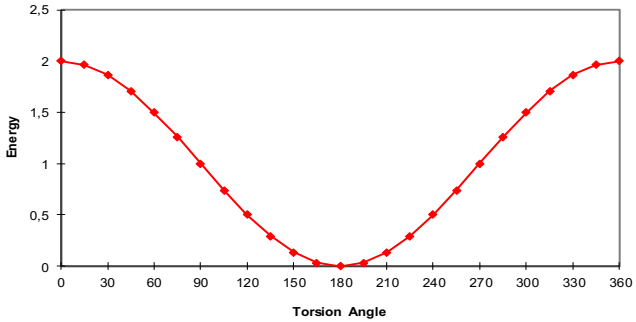


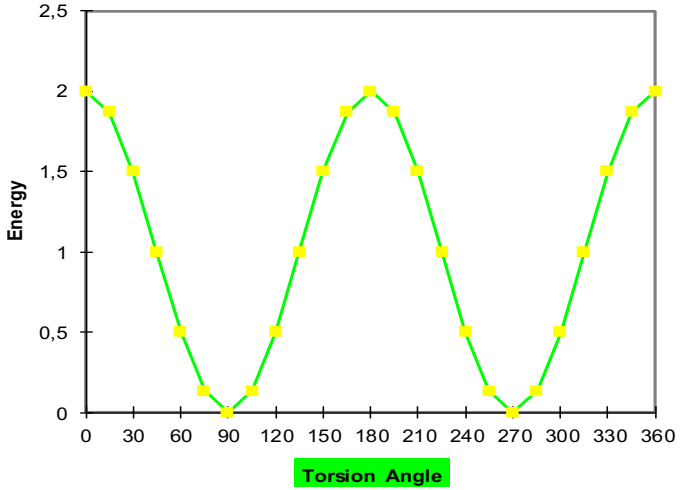
In the “wagging” mode (bottom left), both atoms move toward the observer.

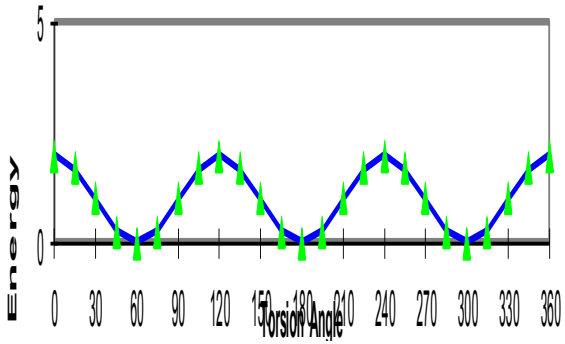
In the “twisting” mode (bottom right) one atom moves toward the observer, the other atom moves away from the observer.

Torsional energy

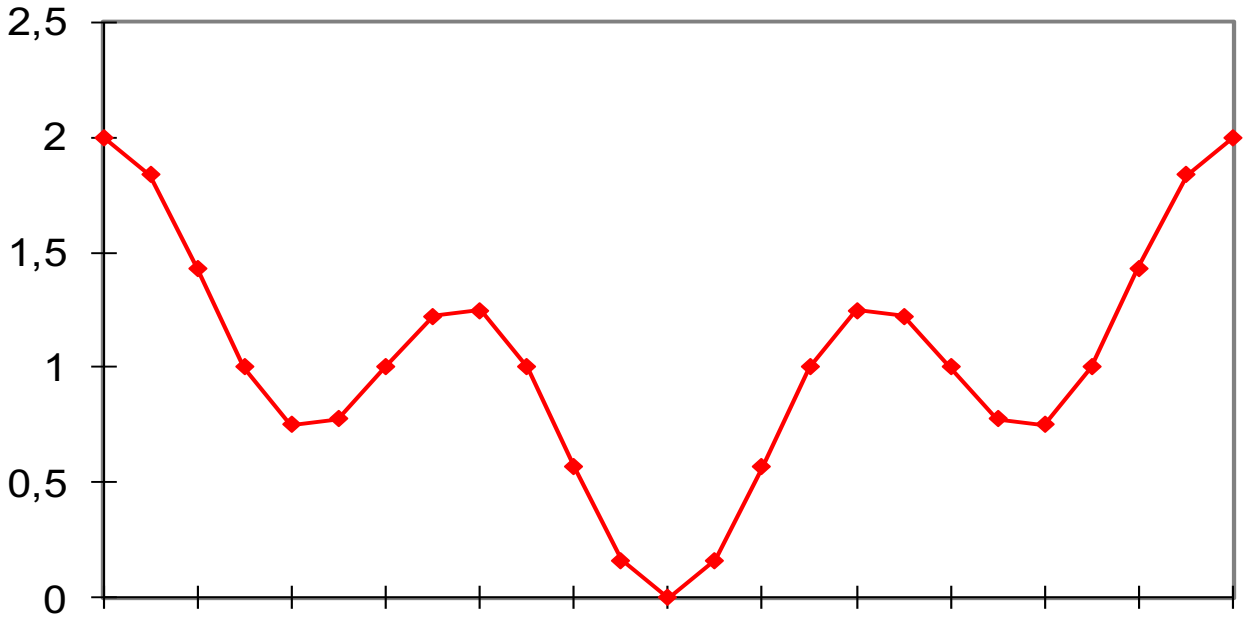
$$E_{torsion} = 0.5\ V_1\ (1 + \cos\phi) + 0.5\ V_2\ (1 + \cos 2\phi) + 0.5\ V_3\ (1 + \cos 3\phi)$$

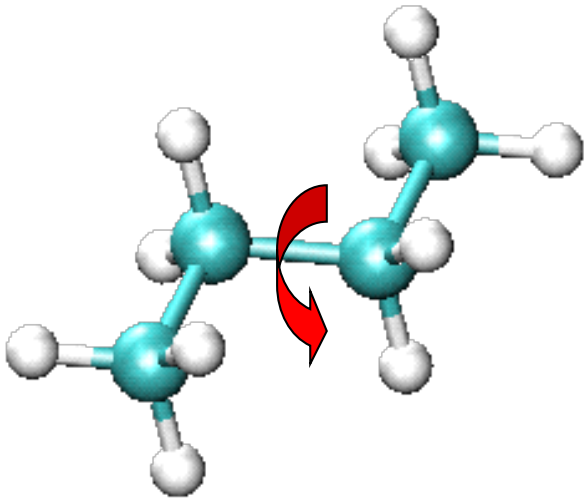






Rotational barrier of butane: sum of 2 terms (1+ cosϕ) + cos3ϕ





Non bonded interactions

Possibilities for parametrization:

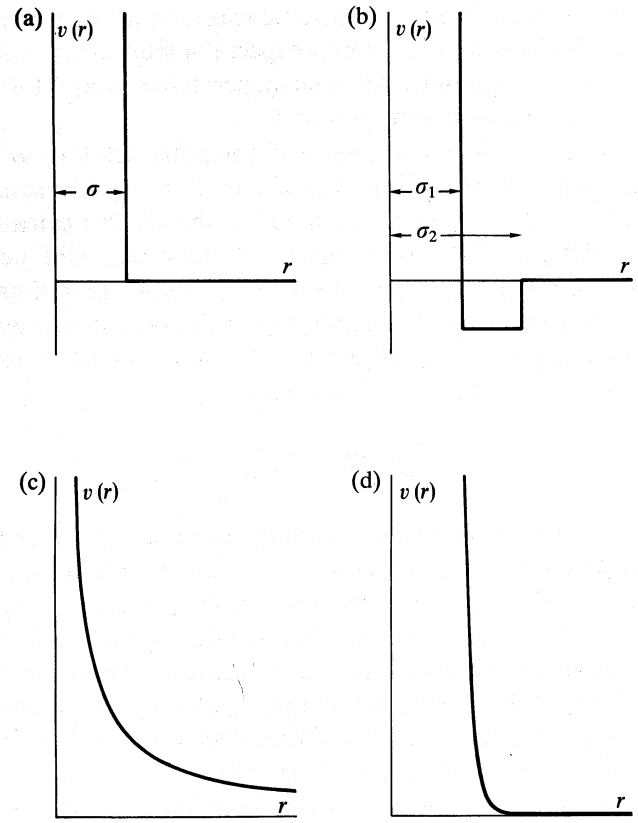


Fig. 1.4 Idealized pair potentials. (a) The hard-sphere potential; (b) The square-well potential; (c) The soft-sphere potential with repulsion parameter $\nu = 1$; (d) The soft-sphere potential with repulsion parameter $\nu = 12$.

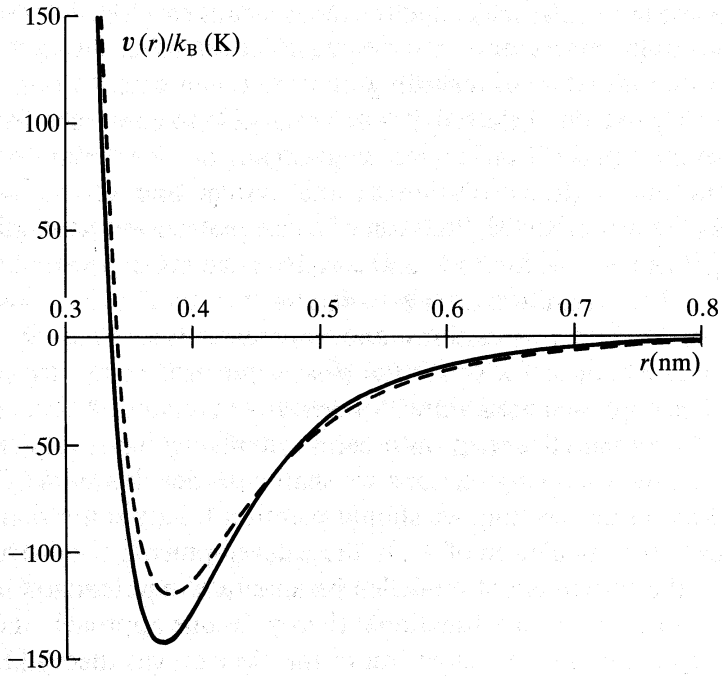


Fig. 1.3 Argon pair potentials. We illustrate the BBMS pair potential for argon (solid line) [Maitland and Smith 1971]. The BFW potential [Barker *et al.* 1971] is numerically very similar. Also shown is the Lennard-Jones 12–6 effective pair potential (dashed line) used in computer simulations of liquid argon.

Uncharged noble atoms interact only via van der Waals (vdW) forces.

[Leach]

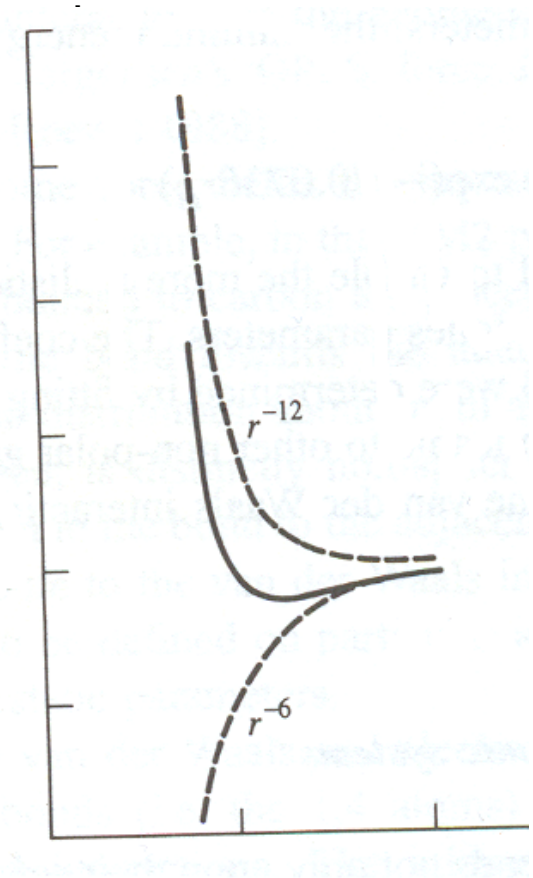
Van der Waals interactions

vdW interactions are based on the overlap of the electron clouds of 2 atoms (**electronic correlation**). One can think of this as an induced dipole-dipole interaction.

At close distances, both electrons repel eachother st
At medium distances, the interaction is attractive.
At large distances, it decays with $1/r^6$.

It can be modelled as: $$E_{vdW} = \frac{A_{(ij)}}{r_{ij}^{12}} - \frac{B_{(ij)}}{r_{ij}^{6}}$$

In force fields, A_{ii} and B_{ii} are fitted against some experimental data for each atom type.

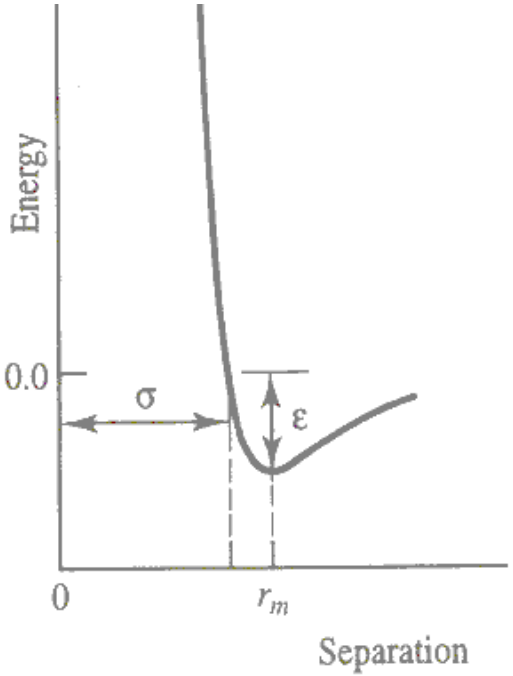


Van der Waals interaction

Analogous representation:

$$E_{vdW} = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

with the collision diameter σ and the depth of the energy minimum ε.



(One can also represent the potential by $r_m = r^*$.

Taking $\frac{\partial E_{vdW}}{\partial r} = 0$

yields $r_m = 2^{1/6}\sigma$

By comparison with $E_{vdW} = \frac{A_{(ij)}}{r_{ij}^{12}} - \frac{B_{(ij)}}{r_{ij}^{6}}$

one obtains:

$$A = 4\varepsilon\sigma^{12} = \varepsilon r_m^{12}$$

$$B = 4\varepsilon\sigma^{6} = 2\varepsilon r_m^{6}$$

Combination rules

To describe arbitrary interactions E_{ij} between two different atom types one often uses the **combination rule** named after Lorentz-Berthelot:

Geometric: $r^*_{AB} = ½ (r^*_{AA} + r^*_{BB})$ or $\sigma_{AB} = ½ (\sigma_{AA} + \sigma_{BB})$

$$\varepsilon^{AB} = \sqrt{\varepsilon^{A}\varepsilon^{B}}$$

Electrostatic interactions: Coulomb law

$$E_{ES} = \frac{1}{4\pi\varepsilon_r\varepsilon_0} \sum_{pairs\,(ij)} \frac{q_i q_j}{r_{ij}}$$

- ε_0 = 8,854 $\times$ 10^{-12} As/Vm : electrical field constant
- ε_r : **relative dielectric constant** of the medium
- e.g. 2 in the interior of proteins and in membranes
- 80 in water

- In molecular mechanics one typically uses ε_r = 1 for the interaction between all atoms that are explicitly modelled
- Neglecting solvent molecules saves a lot computing time. Then, the solvent effect can be modelled by a distance-dependent dielectric constant $\varepsilon_r = r$.
- The idea of this is: if solvent molecules were present, they would orient themselves to screen all pairwise interactions.
- q_i and q_j are suitable **atomic point charges**

Force field parametrization

Covalent terms

- Bond lengths and -angles	X-ray structures of small molecules (CSD)
- Dihedral (torsional) angles	stereo chemistry
- force constants	e.g. from infrared vibrational spectroscopy

This approach has only difficulties if **cross terms** are used in the force field.

Most accurate force field: **Merck force field MMFF**, that was parametrized against accurate quantum chemical calculations.

Non covalent terms

Are tuned so that conformations are correctly sampled and energy differences of ΔE und ΔG are correctly computed.

Take home message 1: force fields are „horror“

Molecular mechanics method is a **serious simplification**.
This is only historically justified because computing resources were severly limited „in the old days“.

- Developers of different empirical force fields split up energy terms in very different ways:
- Some implement any possible sort of interaction,
- others use only 5-10 terms. Missing terms need to be compensated by ad hoc **parametrisions.**

- Values of the potential function have **no physical meaning**, but depend on potential function and parameters.

Take home message 2: force fields are practically useful

Relative energy difference of isomers is predicted quite well by MM force fields

→ MM force fields are well suited for **conformational analyses**

(though peptides are particularly tricky …)

Force fields enable simple calculation **intermolecular interactions.**

Force field calculations are **very efficient**

→ Only force fields allow treating **large systems** with 100.000 – 1.00.000 atoms, and even their **dynamics**.

History of MD simlations

1957/1959: First MD simulation of hard spheres mimicking fluids (Alder & Wainwright)

1964 first simulation with realistic potential for liquid argon (Rahman, 1964)

1974 first MD simulation of water (Stillinger and Rahman, 1974)

1977 first MD simulation of the protein BPTI (McCammon, *et al*, 1977).

1977 SHAKE algorithm (to freeze bond lengths)

1983 CHARMM, AMBER force fields

1984 free energy calculations

1985 Car-Parrinello MD

1987 Gromos87 force fields

since 1990 QM/MM methods, e.g. of enzymatic reactions (Noble prize 2013)

1992 Ewald method (efficient FFT computation of long-range electrostatics)

1990ies: MD simulation on parallel processos $\rightarrow$ larger systems

Alder, B. J. and Wainwright, T. E. *J. Chem. Phys.* **27**, 1208 (1957)

Alder, B. J. and Wainwright, T. E. *J. Chem. Phys.* **31**, 459 (1959)

Rahman, A. *Phys. Rev.* **A136**, 405 (1964)

Stillinger, F. H. and Rahman, A. *J. Chem. Phys.* **60**, 1545 (1974)

McCammon, J. A., Gelin, B. R., and Karplus, M. *Nature (Lond.)* **267**, 585 (1977)

A simple MD package

```
program md

call init                          Initialization
t = 0
do while (t.lt.tmax)               MD loop
       call force(f,en)            compute forces
       call integrate(f,en)        integrate equations of motion
       t = t + delta
       call sample                 compute averages
enddo
stop
end
```

after [Frenkel, Smit 1996]

Initialization in an MD program

```
subroutine init                    Initialization
sumv=0
sumv2=0
do i=1,npart
   x(i)= lattice_pos(i)            place particles on lattice (or read starting coordinates)
   v(i)=(ranf()-0.5)               assign random velocities
   sumv=sumv+v(i)                  velocity of the center of mass (CMS)
   sumv2=sumv2+v(i)**2             kinetic energy
enddo
sum=sumv/npart                     velocity of the CMS
sumv2=sumv2/npart                  mean squared velocity
fs=sqrt(3*temp/sumv2)              scaling factor for particle velocities
do i=1,npart                       set desired kinetic energy
   v(i)=(v(i)-sumv)*fs             - set CMS velocity to zero
   xm(i)=x(i)-v(i)*dt              position at previous time step
enddo                              - (is needed for integration algorithm)
return
end
```

after [Frenkel, Smit 1996]

periodic boundary conditions (PBC)

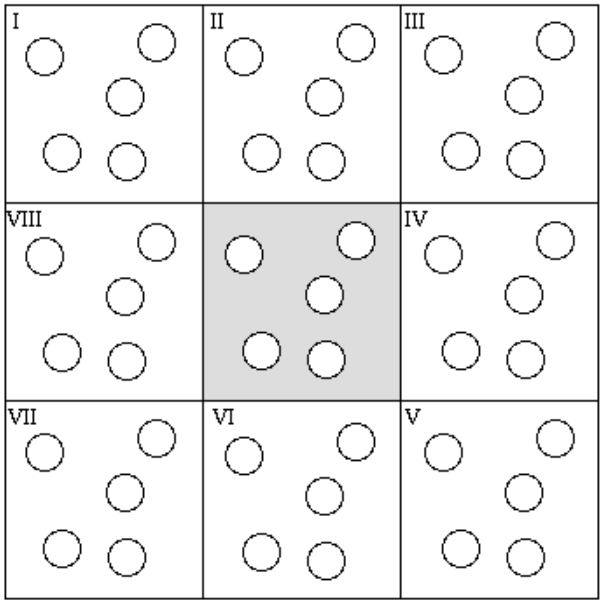
Using PBC enables simulations with fairly small particle numbers, whereby particles are subject to forces like in solution.

The coordinates of the periodic „**copies**“ are obtained by simply **translating** the coordinates of the particles in the central box.

Central particles feel interactions with other central particles or with their periodic copies.

A suitable **cut-off** ensures that a central particle cannot interact both with another central particle and with one of its copies.

For a cubic box: `cut-off < box/2`



2-dimensional box.

central box is surrounded by 8 neighbors

nach [Frenkel, Smit 1996]

Computation of forces

```
subroutine force(f,en)                    Initialization
en=0
do i=1,npart
    f(i)= 0                               set forces to 0
enddo
do i=1,npart-1
    do j=i+1,npart                        Loop over all pairs of particles (atoms)
        xr=x(i)-x(j)                      compute distance of particles i and j
        xr=xr-box*nint(xr/box)            periodic boundary conditions
        r2=xr**2
        if (r2.lt.rc2) then               check whether distance < cut-off
            r2i=1/r2
            r6i=r2i**3
            ff=48*r2i*r6i*(r6i-0.5)       in this example:  Lennard-Jones potential
            f(i)=f(i)+ff*xr               Update of forces
            f(j)=f(j)-ff*xr
            en=en+4*r6i*(r6i-1)-ecut              Update of energy
        endif
    enddo
enddo
return
end
```

nach [Frenkel, Smit 1996]

Taylor expansion

Let function *f* be n+1 times differentiable in an interval $[x_0-\alpha, x_0+\alpha]$.

Then, the **Taylor expansion** applies in this interval:

$$f(x)=\sum_{\nu=0}^{n}\frac{1}{\nu!}\frac{\partial^{\nu} f(x_0)}{\partial x^{\nu}}(x-x_0)^{\nu}+R_n(x)$$

mit

$$R_n(x)\overset{def}{=}\frac{1}{(n+1)!}\frac{\partial^{n+1} f(x_0+\vartheta(x-x_0))}{\partial x^{n+1}}(x-x_0)^{n+1}, \qquad \vartheta\in[0,1]$$

Integration of the equations of motion

Taylor expansion of the particle coordinates **r**(t)

$$\mathbf{r}(t+\Delta t)=\mathbf{r}(t)+\mathbf{v}(t)\Delta t+\frac{\mathbf{F}(t)}{2m}\Delta t^2+\frac{\Delta t^3}{3!}\dddot{\mathbf{r}}+O(\Delta t^4),$$

$$\mathbf{r}(t-\Delta t)=\mathbf{r}(t)-\mathbf{v}(t)\Delta t+\frac{\mathbf{F}(t)}{2m}\Delta t^2-\frac{\Delta t^3}{3!}\dddot{\mathbf{r}}+O(\Delta t^4)$$

Summation gives

$$\mathbf{r}(t+\Delta t)+\mathbf{r}(t-\Delta t)=2\mathbf{r}(t)+\frac{\mathbf{F}(t)}{m}\Delta t^2+O(\Delta t^4)$$

or

$$\mathbf{r}(t+\Delta t)\approx 2\mathbf{r}(t)-\mathbf{r}(t-\Delta t)+\frac{\mathbf{F}(t)}{m}\Delta t^2$$

Subtraction yields

$$\mathbf{r}(t+\Delta t)-\mathbf{r}(t-\Delta t)=2\mathbf{v}(t)\Delta t+O(\Delta t^3)$$

and

$$\mathbf{v}(t)=\frac{\mathbf{r}(t+\Delta t)-\mathbf{r}(t-\Delta t)}{2\Delta t}+O(\Delta t^2)$$

This is the **Verlet algorithm**.

Velocities do not show up explicitly, but can be computed from the trajectory.

Leap-frog algorithm

Leap-frog ("Bocksprung-") **algorithm**

using

$$\mathbf{v}\left(t-\frac{\Delta t}{2}\right)\equiv\frac{\mathbf{r}(t)-\mathbf{r}(t-\Delta t)}{\Delta t}$$

$$\mathbf{v}\left(t+\frac{\Delta t}{2}\right)\equiv\frac{\mathbf{r}(t+\Delta t)-\mathbf{r}(t)}{\Delta t}$$

yields

$$\mathbf{r}(t+\Delta t)=\mathbf{r}(t)+\mathbf{v}\left(t+\frac{\Delta t}{2}\right)\Delta t$$

The *update* of the velocities is:

$$\mathbf{v}\left(t+\frac{\Delta t}{2}\right)=\frac{\mathbf{r}(t+\Delta t)-\mathbf{r}(t)}{\Delta t}$$

$$=\mathbf{v}\left(t-\frac{\Delta t}{2}\right)-\frac{\mathbf{r}(t)-\mathbf{r}(t-\Delta t)}{\Delta t}+\frac{\mathbf{r}(t+\Delta t)-\mathbf{r}(t)}{\Delta t}$$

$$=\mathbf{v}\left(t-\frac{\Delta t}{2}\right)+\Delta t\,\frac{\mathbf{F}(t)}{m}$$

as the Verlet algorithm yields

$$\Delta t\,\frac{\mathbf{F}(t)}{m}=\frac{\mathbf{r}(t+\Delta t)-2\mathbf{r}(t)+\mathbf{r}(t-\Delta t)}{\Delta t}$$

Integrate equations of motion

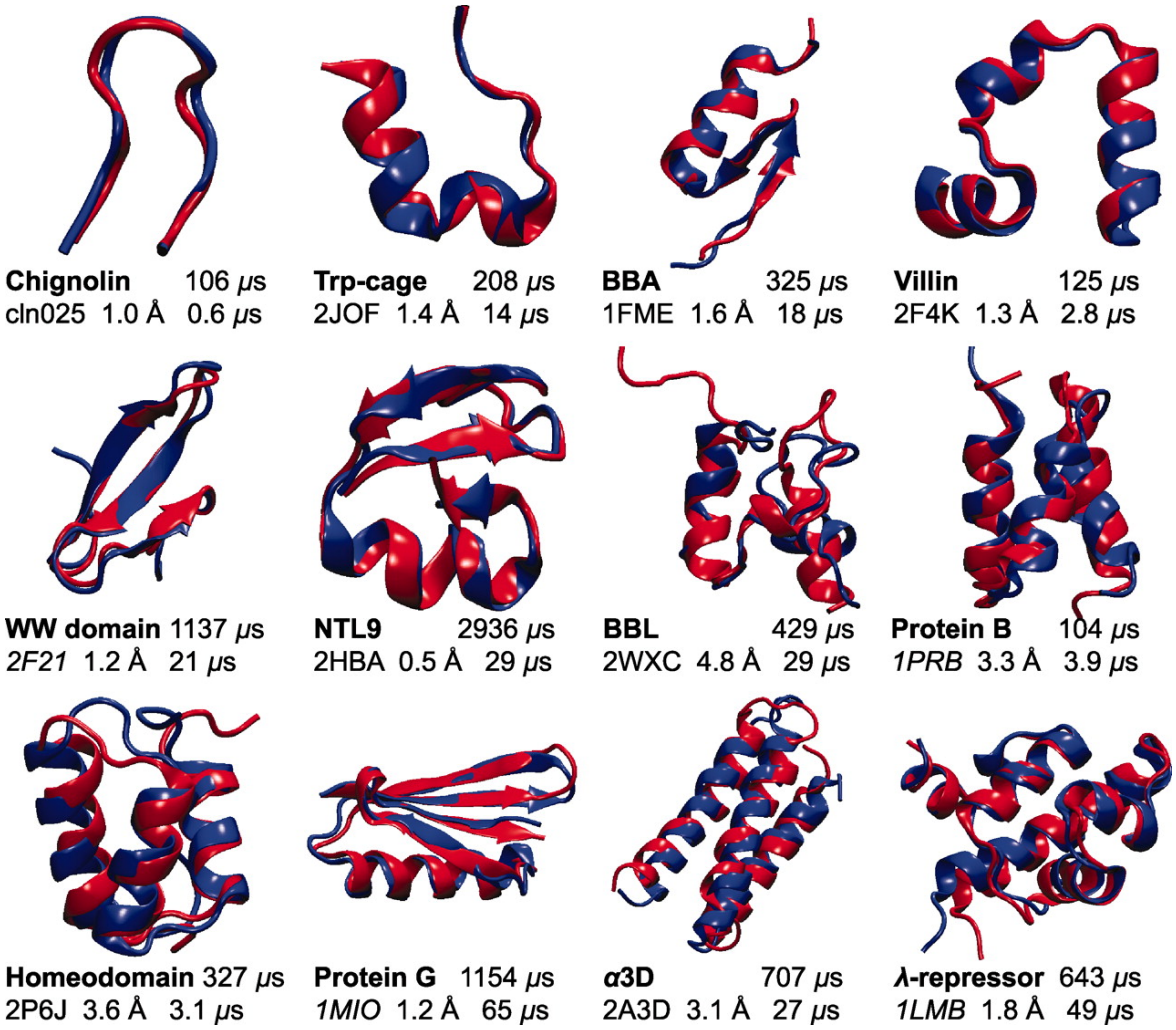
```
subroutine integrate(f,en)      integrate equations of motion
sumv=0
sumv2=0
do i=1,npart                    MD loop
   xx=2*x(i)-xm(i)+delt**2*f(i)        Verlet algorithm
   v(i)=(xx-xm(i))/(2*delt)     velocities
   sumv=sumv+v(i)               CMS velocity
   sumv2=sumv2*v(i)**2          total kinetic energy
   xx(i)=xx                     position update
enddo
temp=sumv2/(3*npart)            current temperature
etot=(en+sumv2)/(2*npart)       total energy per particle
return
end
```

nach [Frenkel, Smit 1996]

Example 1:
Fast-folding proteins fold/unfold in MD simulations

- 12 protein domains (10 to 80 amino acids) without disulfide bonds or prosthetic groups.
- Different topologies: α-helical, β sheet and mixed α/β.
- MD simulations used a single force field and included explicitly represented solvent molecules.

Fig shows folded structure obtained from simulation (blue) superimposed on the experimentally determined structure (red),



Lindorff-Larsen, …, David E. Shaw
Science (2011) 334, 517-520

Fast-folding proteins fold/unfold in MD simulations

For each of the 12 proteins, 1 to 4 simulations were performed, each between 100 μs and 1 ms long.

For each protein, at least 10 folding and 10 unfolding events were observed.
In ~8 ms of simulation, more than 400 folding or unfolding events were observed.

For 8 of the 12 proteins, the most representative structure of the folded state fell within 2 Å root mean square deviation (RMSD) of the experimental structure.

Lindorff-Larsen, …, David E. Shaw
Science (2011) 334, 517-520

Example 2:
How do voltage-dependent K^+ channels work?

Hodgkin and Huxley discovered that **voltage-regulated ion flow** underlies **nerve conduction**.

Voltage-sensing domains (VSD) control the activity of voltage-gated K^+, Na^+, and Ca^{2+} channels, shifting these proteins between activated and deactivated states in response to changes in transmembrane voltage.

Different mechanistic models have been proposed to describe how conserved arginine and lysine gating-charge residues on a VSD transmembrane helix (**S4**) couple with the electric field to gate ion channel conduction.

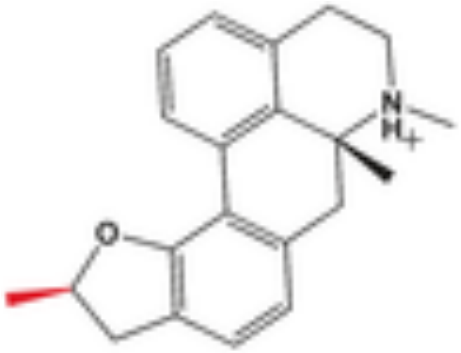
Show **movies 1 and 7** at
http://science.sciencemag.org/content/suppl/2012/04/11/336.6078.229.DC1

Jensen, … David E. Shaw,
Science (2012) 336, 229–233.

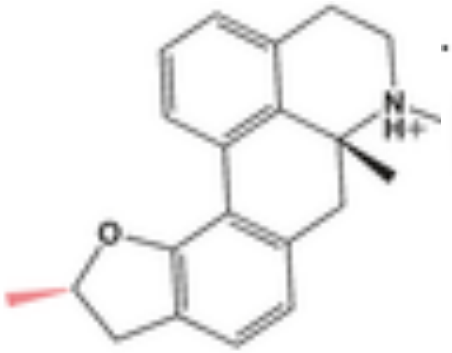
Example 3:
Stereoselectivity of the Serotonin 5-HT_{1A} Receptor

Q: Why do certain molecules act as activators (**agonists**) whereas others, with nearly identical structures, act as blockers (**antagonists**) of GPCRs?

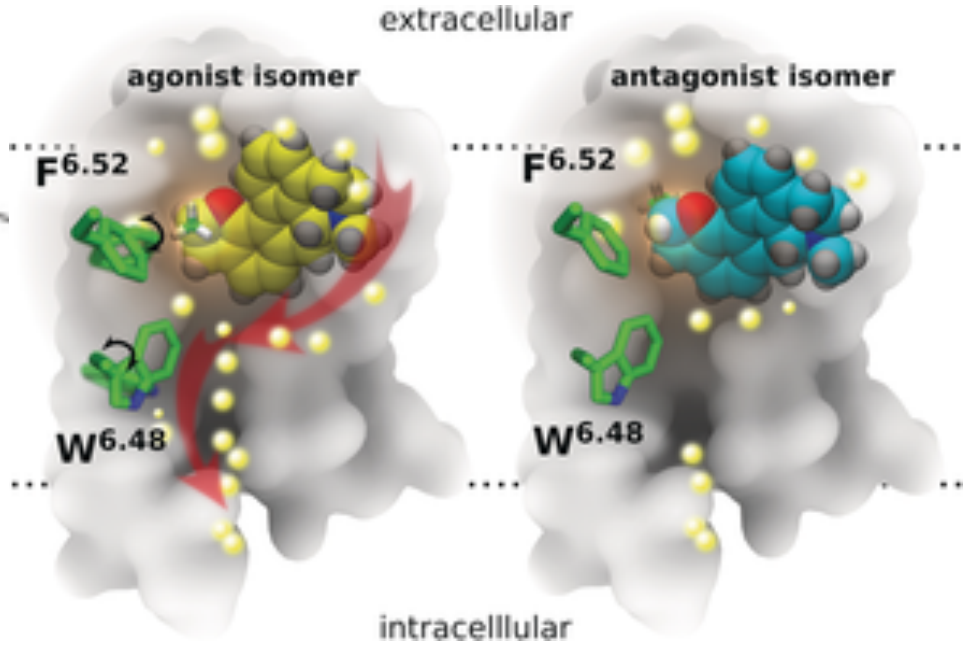
Approach: study by MD simulations how two diastereomers (epimers) of dihydrofuroaporphine bind to the serotonin 5-HT_{1A} receptor and exert opposite effects.



agonist



antagonist



Yuan et al. (2016) Angew Chemie Int Ed 55, 8661

Example 3:
Stereoselectivity of the Serotonin 5-HT_{1A} Receptor

(1) The methyl group at the chiral center of the agonist molecule contacts Phe362$^{6.52}$ of the receptor through hydrophobic interactions, thereby resulting in a **rotamer switch** of the phenyl group of this residue.

(2) This first movement induces **structural changes** in the **transmission switch**, including the central residue in this switch, the highly conserved Trp358$^{6.48}$.

(3) This opens a **gate**, followed by opening of the 3–7 lock of the receptor, thereby

(4) eventually allowing **diffusion** of **water** molecules from the bulk extracellular phase towards the central cytoplasmic internal space of the receptor.

(5) The successive movement of water molecules into the receptor induces **structural changes** in TM5, TM6, and TM7, first **bending** and then **rotation**,

(6) This finally enables the binding and activation of a G protein at the intracellular site of the receptor.

Are the observations of the MD simulations statistically significant?

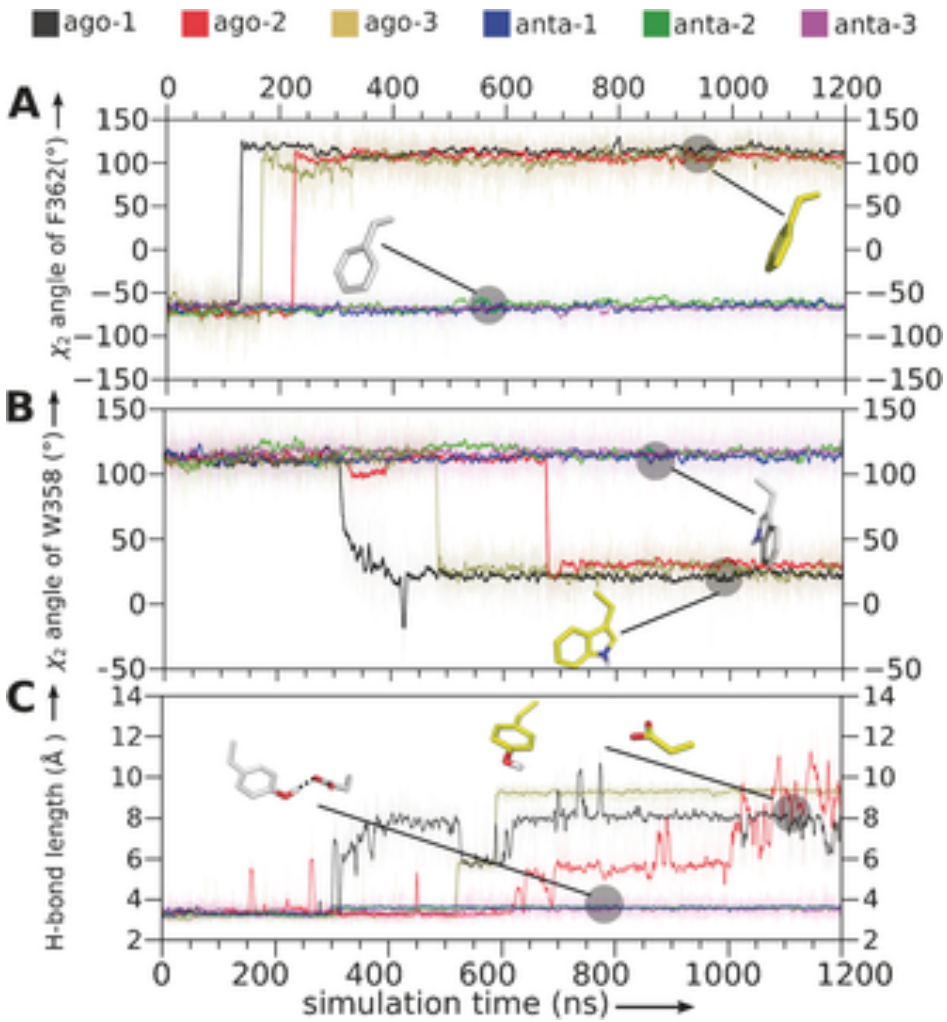
Yuan et al. (2016) Angew Chemie Int Ed 55, 8661

Example 3:
Stereoselectivity of the Serotonin 5-HT_{1A} Receptor

3 × 1.2 µs all-atom MD simulations were performed for both the agonist-bound and the antagonist-bound 5-HT_{1A} receptor.

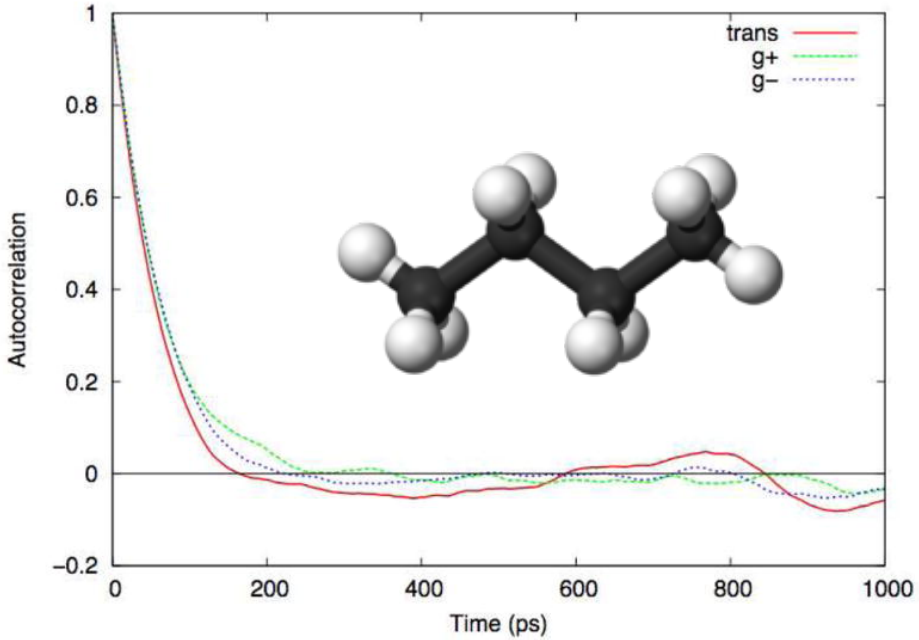
All 3 agonist-simulations showed the conformational transitions (feature π), but none of the antagonist-simulations.

According to the hypergeometric test,
N = 6, K_π = 3
n = 3, k_π = 3
this is statistically significant (p = 0.05).



Yuan et al. (2016) Angew Chemie Int Ed 55, 8661

Example 4:
Autocorrelation of observables in trajectory



State autocorrelations computed from 100 ns butane simulations.
The central torsion was labeled as either trans, g^+, or g^-.

$$c_f(t') = \frac{\langle [f(t) - \langle f \rangle][f(t+t') - \langle f \rangle] \rangle}{\sigma_f^2}$$

Grosfield (2009) Annu Rev Comput Chem 5, 23

All 3 correlation functions drop smoothly to zero within 200 ps → a 100 ns simulation contains many independent samples.

BUT: more complicated systems/variables have much longer autocorrelation times!

The populations for the 3 states are 0.78, 0.10, and 0.13 for trans, g+, and g−.

BUT g+ and g− are physically identical. They should have the same population!
→ even a very long simulation of a very simple system is incapable of estimating populations with high precision.