Biomolecular Simulation: A Computational Microscope for Molecular Biology

Ron O. Dror,[1] Robert M. Dirks,[1] J.P. Grossman,[1] Huafeng Xu,[1] and David E. Shaw[1,2]

[1]D. E. Shaw Research, New York, New York 10036; email: Ron.Dror@DEShawResearch.com; David.Shaw@DEShawResearch.com

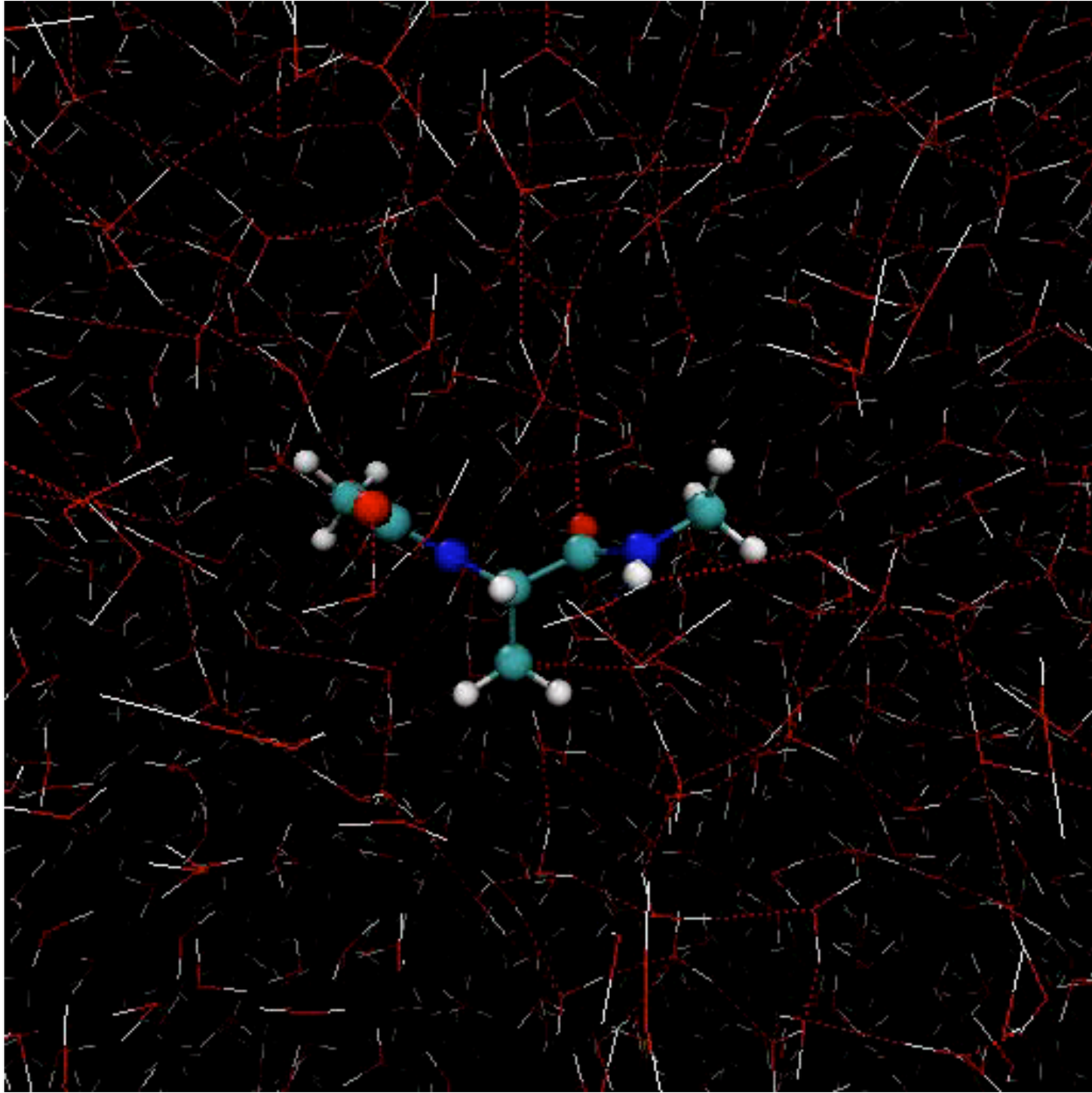
[2]Center for Computational Biology and Bioinformatics, Columbia University, New York, New York 10032

Molecular Dynamics Simulations

Structural Bioinformatics

Ideally, what should a computational microscope do?

- Observe the time evolution of molecules at high spatial and time resolution
- Observe them for very long time scales
- Be able to set different experimental conditions (Temperature, Pressure, solution conditions)
- Be accurate
- Be interpretable, that is find suitable macrostates to compare with experiments
- …

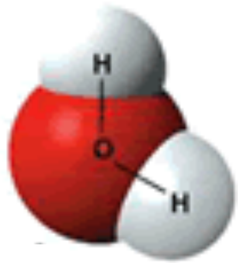


What is a good representation of a molecule?

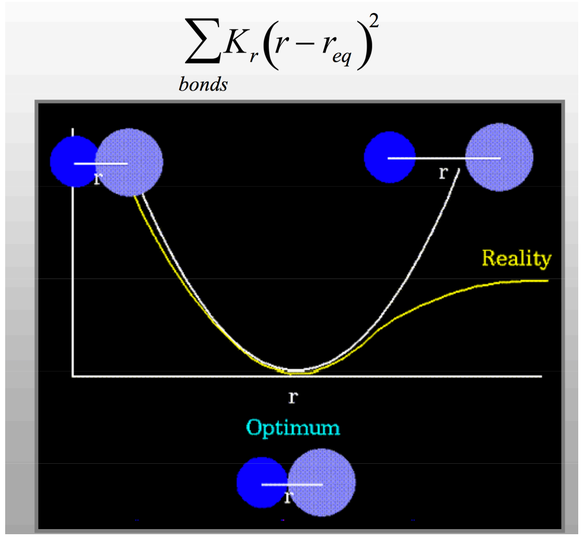
Structural experiments allow us to see molecules at atomic resolution and in a fixed chemical configuration, this is a molecular perspective (i.e. a molecule is given)

Molecules have covalent properties and non-covalent interactions

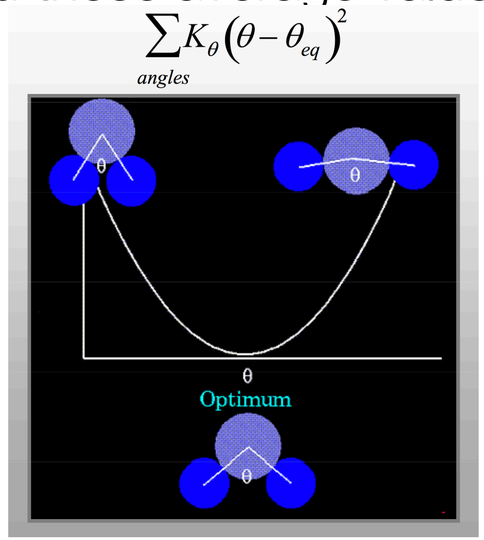
Covalent (Geometrical)
O-H distance: 0.9572 Å
H-O-H angle: 104.52
With fluctuations around these average values



=

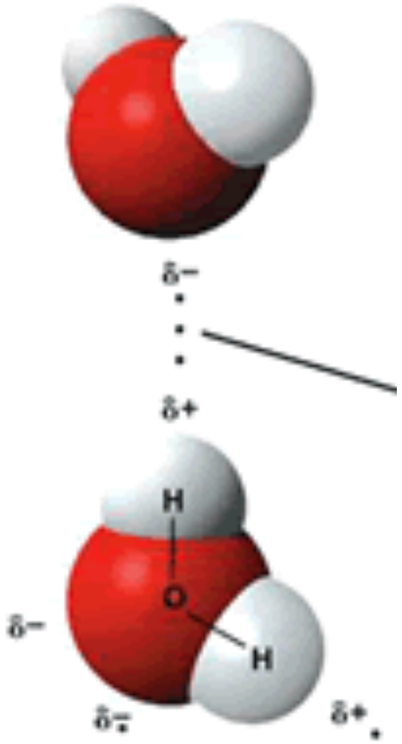


+



What is a good representation of a molecule?

Non-covalent interactions
Lone pair electrons that can form hydrogen bonds
The oxygen is negatively charged while hydrogens are positively charged



H-bond

=

$$+\sum_{i<j}\left[\frac{q_i q_j}{\varepsilon r_{ij}}\right]$$ Electrostatic

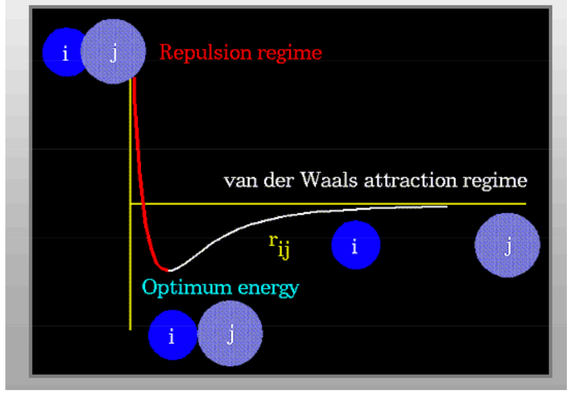
E

$q_i q_j > 0$ Repulsive potential

$q_i q_j < 0$ Attractive potential

r_{ij} +

$$\sum_{i<j}\left[\frac{A_{ij}}{r_{ij}^{12}}-\frac{B_{ij}}{r_{ij}^{6}}\right]$$



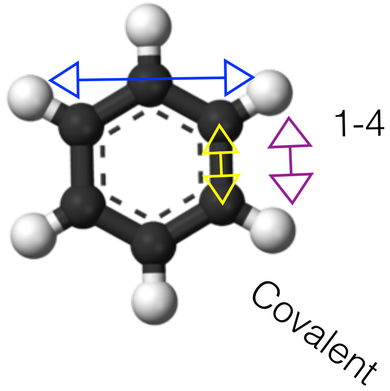
Otherwise an O can overlap with an H

What is a good representation of a molecule?

Are distances and angles enough to describe the geometry?

For example in the case of benzene this is not enough

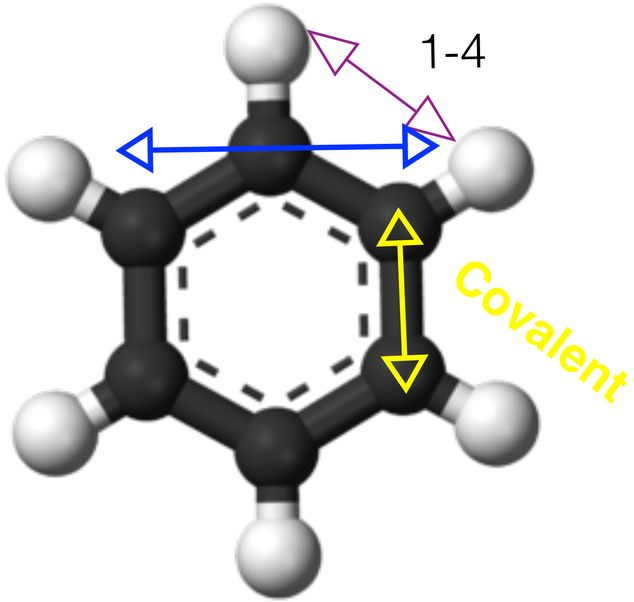
Non-covalent



We use dihedral interactions to add information about the relative orientation of two couples of atoms

$$\sum_{\text{dihedrals}} K_\phi \left(1 + \cos\left(n\phi\right)\right)$$

Molecular Mechanics Force Fields



Potential energy of a given configuration of atoms

first approx for vibrations (anharmonic potential can be used for more accurate vibrations)

geometrical consideration pi-bonds, etc

$$V(r) = \sum_{bonds} k_b(b - b_0)^2 + \sum_{angles} k_\theta(\theta - \theta_0)^2 + \sum_{torsions} k_\phi[\cos(n\phi + \delta) + 1]$$

$$+ \sum_{\substack{nonbond \\ pairs}} \left[\frac{q_i q_j}{r_{ij}} + \frac{A_{ij}}{r_{ij}^{12}} - \frac{C_{ij}}{r_{ij}^{6}} \right]$$

point charge Coulomb

excluded volume

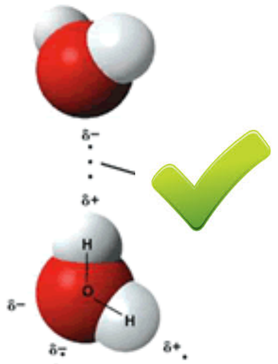
Dispersions (interactions of neutral molecules)

What is a good representation of a molecule?

Are Coulomb and Lennard-Jones interactions enough for non-covalent interactions?

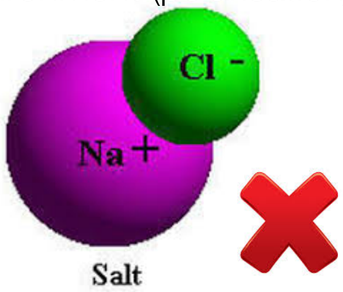
Neutral molecules

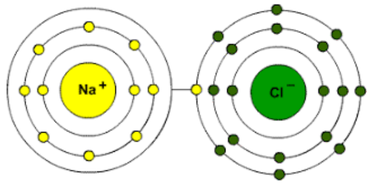
Yes, interaction energies between neutral molecules can be reproduced rather accurately (dipolar interaction)



Ionic interactions

No because the charge of an atom is fixed, while in reality it reacts to the environment (polarisation)

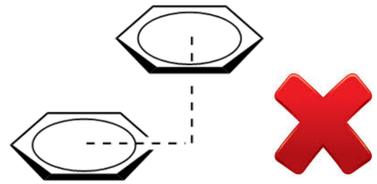




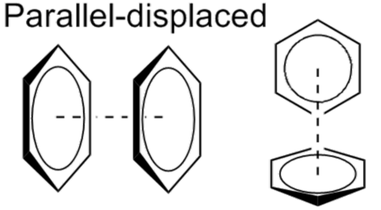
Almost OK for monovalent, bad for others

Pi-stacking

Weakly, because the charge of a ring is distributed over and below the ring (multipole)



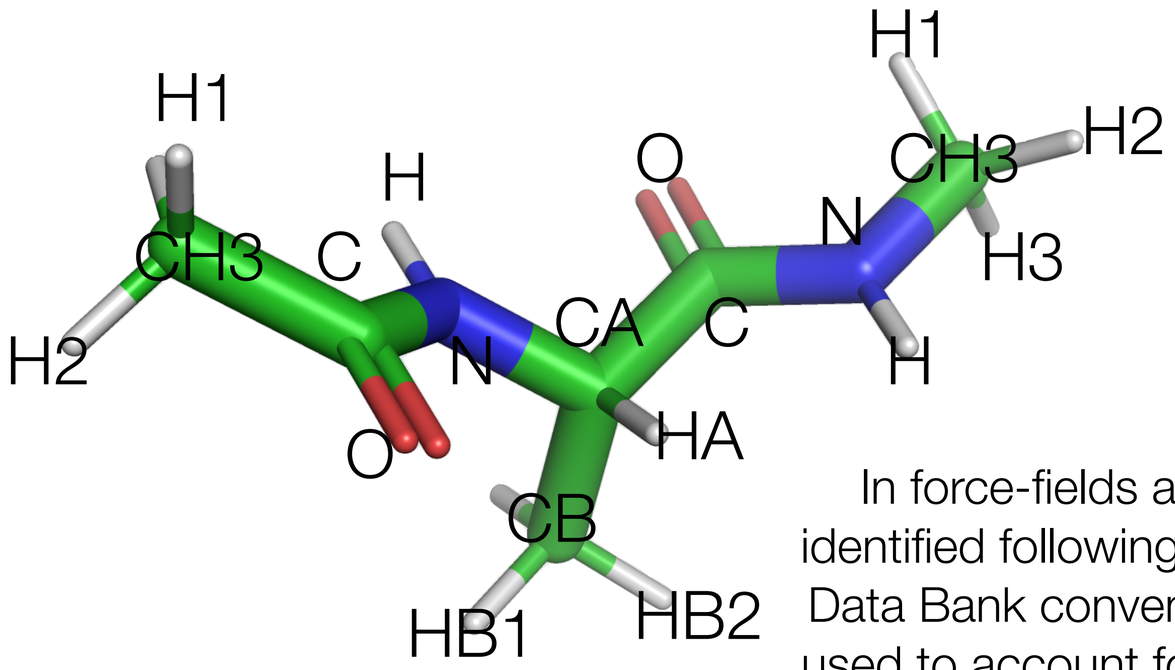
Parallel-displaced



Sandwich T-shaped

Force Fields

a force field has ~5000-10000 parameters to describe all the possible bonds, angles, torsion, Coulomb and Van der Waals interactions between atoms.



In force-fields atoms are identified following the Protein Data Bank convention, this is used to account for their local chemical environment and thus allow to define different force-field terms:

```
[ atoms ]
;   nr       type  resnr residue  atom   cgnr     charge       mass  typeB    char
; residue   1 ACE rtp ACE  q  0.0
     1         CT      1    ACE    CH3      1          0      12.01
     2         HC      1    ACE   HH31      2       0.02      1.008
     3         HC      1    ACE   HH32      3       0.02      1.008
     4         HC      1    ACE   HH33      4       0.02      1.008
     5          C      1    ACE      C      5   0.533742      12.01
     6          O      1    ACE      O      6  -0.593742         16   ; qtot 0
; residue   2 ALA rtp ALA  q  0.0
     7          N      2    ALA      N      7  -0.278592      14.01
     8          H      2    ALA      H      8   0.305959      1.008
     9         AA      2    ALA     CA      9     0.0337      12.01
    10      H1_H1B      2    ALA     HA     10     0.0823      1.008
    11      CT_CT       2    ALA     CB     11    -0.1825      12.01
    12         HC      2    ALA    HB1     12     0.0603      1.008
    13         HC      2    ALA    HB2     13     0.0603      1.008
    14         HC      2    ALA    HB3     14     0.0603      1.008
    15          C      2    ALA      C     15   0.451975      12.01
    16          O      2    ALA      O     16  -0.593742         16   ; qtot 0
; residue   3 NME rtp NME  q  0.0
    17          N      3    NME      N     17  -0.278592      14.01
    18          H      3    NME      H     18   0.305959      1.008
    19         CT      3    NME    CH3     19  -0.087367      12.01
    20         H1      3    NME   HH31     20       0.02      1.008
    21         H1      3    NME   HH32     21       0.02      1.008
    22         H1      3    NME   HH33     22       0.02      1.008   ; qtot 0
```

Force Fields

a force field has ~5000-10000 parameters to describe all the possible bonds, angles, torsion, Coulomb and Van der Waals interactions between atoms.

```
[ bonds ]
;  ai    aj  funct
    1     2      1
    1     3      1
    1     4      1
    1     5      1
    5     6      1
    5     7      1
    7     8      1
    7     9      1
    9    10      1
    9    11      1
    9    15      1
   11    12      1
   11    13      1
   11    14      1
   15    16      1
   15    17      1
   17    18      1
   17    19      1
   19    20      1
   19    21      1
   19    22      1
```

```
[ pairs ]
;  ai    aj funct
    1     8     1
    1     9     1
    2     6     1
    2     7     1
    3     6     1
    3     7     1
    4     6     1
    4     7     1
    5    10     1
    5    11     1
    5    15     1
    6     8     1
    6     9     1
    7    12     1
    7    13     1
    7    14     1
    7    16     1
    7    17     1
    8    10     1
    8    11     1
    8    15     1
    9    18     1
    9    19     1
   10    12     1
   10    13     1
   10    14     1
   10    16     1
   10    17     1
   11    16     1
   11    17     1
   12    15     1
   13    15     1
   14    15     1
   15    20     1
   15    21     1
   15    22     1
   16    18     1
   16    19     1
   18    20     1
   18    21     1
   18    22     1
```

```
[ angles ]
;  ai    aj    ak funct
    2     1     3     1
    2     1     4     1
    2     1     5     1
    3     1     4     1
    3     1     5     1
    4     1     5     1
    1     5     6     1
    1     5     7     1
    6     5     7     1
    5     7     8     1
    5     7     9     1
    8     7     9     1
    7     9    10     1
    7     9    11     1
    7     9    15     1
   10     9    11     1
   10     9    15     1
   11     9    15     1
    9    11    12     1
    9    11    13     1
    9    11    14     1
   12    11    13     1
   12    11    14     1
   13    11    14     1
    9    15    16     1
    9    15    17     1
   16    15    17     1
   15    17    18     1
   15    17    19     1
   18    17    19     1
   17    19    20     1
   17    19    21     1
   17    19    22     1
   20    19    21     1
   20    19    22     1
   21    19    22     1
```

```
[ dihedrals ]
;  ai    aj    ak    al funct
    2     1     5     6     9
    2     1     5     7     9
    3     1     5     6     9
    3     1     5     7     9
    4     1     5     6     9
    4     1     5     7     9
    1     5     7     8     9
    1     5     7     9     9
    6     5     7     8     9
    6     5     7     9     9
    5     7     9    10     9
    5     7     9    11     9
    5     7     9    15     9
    8     7     9    10     9
    8     7     9    11     9
    8     7     9    15     9
    7     9    11    12     9
    7     9    11    13     9
    7     9    11    14     9
   10     9    11    12     9
   10     9    11    13     9
   10     9    11    14     9
   15     9    11    12     9
   15     9    11    13     9
   15     9    11    14     9
    7     9    15    16     9
    7     9    15    17     9
   10     9    15    16     9
   10     9    15    17     9
   11     9    15    16     9
   11     9    15    17     9
    9    15    17    18     9
    9    15    17    19     9
   16    15    17    18     9
   16    15    17    19     9
   15    17    19    20     9
   15    17    19    21     9
   15    17    19    22     9
   18    17    19    20     9
   18    17    19    21     9
   18    17    19    22     9
```

```
[ bondtypes ]
; i        j       func       b0          kb
YY        CT         1    0.1526000   259408.00 ;
C         YY         1    0.1522000   265265.60 ;
YY        H1         1    0.1090000   284512.00 ;
YY        HP         1    0.1090000   284512.00 ;
YY        N          1    0.1449000   282001.60 ;
YY        N3         1    0.1471000   307105.60 ;
WW        CT         1    0.1526000   259408.00 ;
C         WW         1    0.1522000   265265.60 ;
WW        H1         1    0.1090000   284512.00 ;
WW        HP         1    0.1090000   284512.00 ;
WW        N          1    0.1449000   282001.60 ;
WW        N3         1    0.1471000   307105.60 ;
VV        CT         1    0.1526000   259408.00 ;
C         VV         1    0.1522000   265265.60 ;
VV        H1         1    0.1090000   284512.00 ;
VV        HP         1    0.1090000   284512.00 ;
VV        N          1    0.1449000   282001.60 ;
VV        N3         1    0.1471000   307105.60 ;
TT        CT         1    0.1526000   259408.00 ;
C         TT         1    0.1522000   265265.60 ;
TT        H1         1    0.1090000   284512.00 ;
TT        HP         1    0.1090000   284512.00 ;
TT        N          1    0.1449000   282001.60 ;
TT        N3         1    0.1471000   307105.60 ;
SS        CT         1    0.1526000   259408.00 ;
C         SS         1    0.1522000   265265.60 ;
SS        H1         1    0.1090000   284512.00 ;
SS        HP         1    0.1090000   284512.00 ;
SS        N          1    0.1449000   282001.60 ;
SS        N3         1    0.1471000   307105.60 ;
RR        CT         1    0.1526000   259408.00 ;
C         RR         1    0.1522000   265265.60 ;
RR        H1         1    0.1090000   284512.00 ;
RR        HP         1    0.1090000   284512.00 ;
RR        N          1    0.1449000   282001.60 ;
RR        N3         1    0.1471000   307105.60 ;
QQ        CT         1    0.1526000   259408.00 ;
```

```
[ angletypes ]
; i        j         k       func       th0          cth
YY        C         OH         1     110.000     669.440 ; Junmei et a
YY        C         O          1     120.400     669.440 ;
YY        C         O2         1     117.000     585.760 ;
YY        C         N          1     116.600     585.760 ;
H1        YY        H1         1     109.500     292.880 ;
H1        YY        N          1     109.500     418.400 ;
HP        YY        HP         1     109.500     292.880 ;
HP        YY        N3         1     109.500     418.400 ;
H1        YY        N3         1     109.500     418.400 ;
C         YY        H1         1     109.500     418.400 ;
C         YY        HP         1     109.500     418.400 ;
C         YY        N          1     110.100     527.184 ;
C         YY        N3         1     111.200     669.440 ;
C         N         YY         1     121.900     418.400 ;
YY        N         H          1     118.040     418.400 ;
YY        N3        H          1     109.500     418.400 ;
YY        N         CT         1     118.000     418.400 ;
YY        N3        CT         1     109.500     418.400 ;
CT        YY        N3         1     111.200     669.440 ;
CT        YY        HP         1     109.500     418.400 ;
YY        CT        CT         1     109.500     334.720 ;
YY        CT        HC         1     109.500     418.400 ;
CT        YY        N          1     109.700     669.440 ;
C         YY        CT         1     111.100     527.184 ;
```

Force Fields

In the end once you have a configuration of your system and you choose a force-field you can calculate the energy of that configuration as parameterised for the specific force-field you are using, then you can get a different configuration and again calculate its force-field energy.

Over time there have been multiple force-fields that have been parameterised and now we have four main force-field families:

AMBER

CHARMM

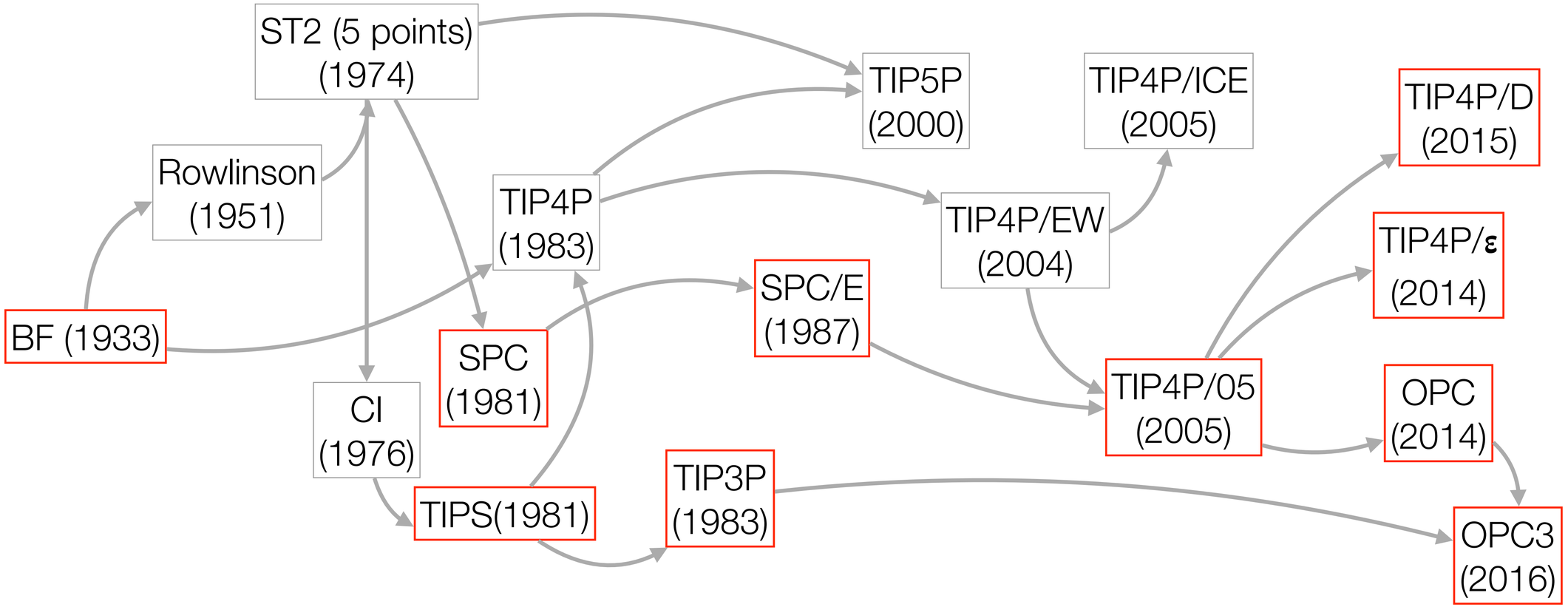
GROMOS

OPLS

And for each family there are multiple generations and variants.

A force-field is then an approximation of the energy of all chemically reasonable configurations of atoms in a biomolecule.

Digression: a brief history of rigid models for water



A Theory of Water and Ionic Solution, with Particular Reference to Hydrogen and Hydroxyl Ions

J. D. Bernal and R. H. Fowler, *University of Cambridge, England*

(Received April 29, 1933)

In 1933 Bernal and Fowler tried to made a model of water to explain the tetrahedral structure of ice and other properties as observed by scattering. They observe that:

"If we take the spectroscopic model and place a charge e at the H positions and -2e on the O nucleus we arrive at a molecule of dipole moment 5.6 D instead of the observed 1.87 D. It is clear that the discrepancy is due to having made no allowance of the effect of the two H+ on the O2- ion. This will be twofold. The centre of the negative electronic charge will be moved from the O nucleus to some position intermediate between it and the H nuclei, and a certain concentration of negative charge will screen the H charges. The simplest assumption is to assume effective charges e' (<e) at the H positions and a charge -2e' situated at x Å from the O nucleus on the bisector of the HÔH angle. With an electric dipole of 2D these are connected by the relation:

$$(0.58-x)\epsilon' = 0.21\epsilon.$$

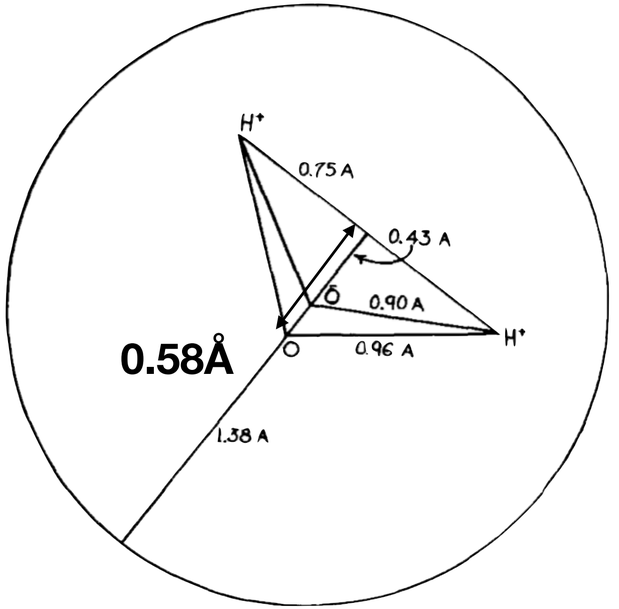


FIG. 10. The water molecule model. H^+H^+ are the hydrogen nuclei; O, the oxygen nucleus; $\bar{O}$ is the centre of negative charge and of the molecule.

p=qd=2e0.58 (dipole e=4.803 10^{-18}(cgs))
(0.58-x)2e'=(2/4.803)e
(0.58-x)=1/4.803 e = 0.21e

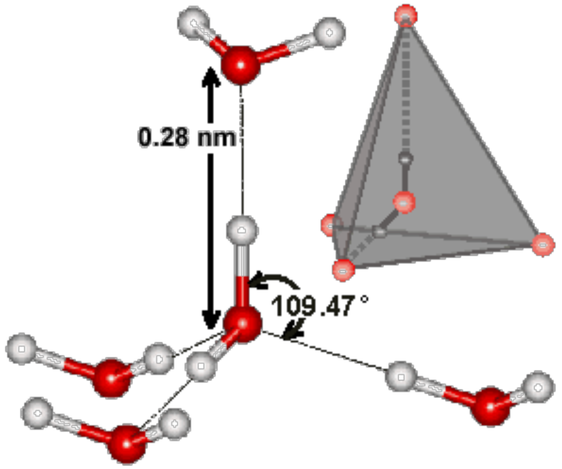
A Theory of Water and Ionic Solution, with Particular Reference to Hydrogen and Hydroxyl Ions

J. D. Bernal and R. H. Fowler, *University of Cambridge, England*

(Received April 29, 1933)

$$(0.58-x)\epsilon' = 0.21\epsilon.$$

...This leaves only the choice of x arbitrary between 0 and 0.37Å **(otherwise e' becomes >1)*. It is plainly nearer the former because for 6 of the 10 electrons of the system the negative centre must be close to the O nucleus. A value 0.15 for x, giving e'=0.49e would not be unreasonable and has the advantage for calculation of giving a tetrahedral arrangement of + and - charges".***



HÔH = 102.75
HÔ'H = 109.9

TIPS & TIP3P (Jorgensen et al, 1981 - 1983)

J. Am. Chem. Soc. **1981**, *103*, 335–340

Transferable Intermolecular Potential Functions for Water, Alcohols, and Ethers. Application to Liquid Water[1]

William L. Jorgensen[2]

Contribution from the Department of Chemistry, Purdue University, West Lafayette, Inidana 47907. *Received May 20, 1980*

Rotation-Vibration Spectra of Deuterated Water Vapor*

W. S. Benedict, *The Johns Hopkins University, Baltimore, Maryland*

AND

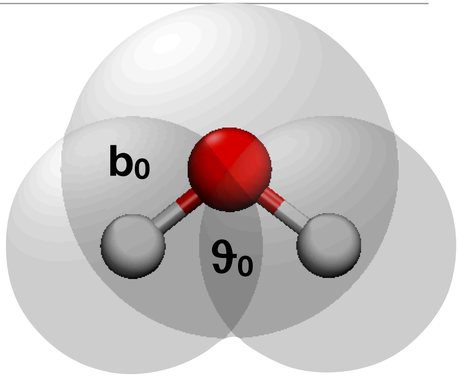
N. Gailar and Earle K. Plyler, *National Bureau of Standards, Washington 25, D. C.*

(Received August 5, 1955)

Spectra of heavy water have been obtained under high resolution between 1.25–4.1μ (2400–8000 cm^{-1}). Approximately 4500 lines have been measured, and the majority of them analyzed into ten bands of D_2O and nine bands of HDO. The analysis is described in some detail, spectra of all bands are shown and a partial table of lines and a complete table of energy levels are presented. The vibration-rotation constants are derived and compared with those of H_2O.

Step 1: Internal geometry

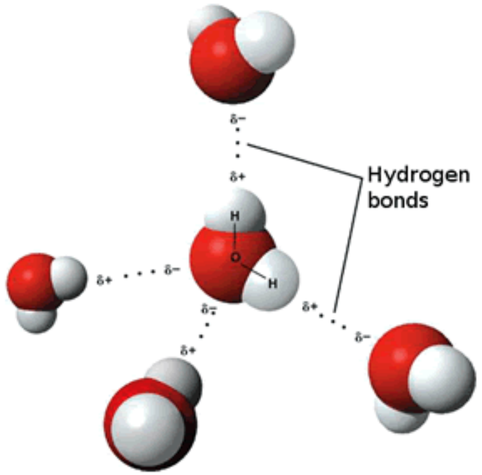
b_0=0.9572 Å
ϑ_0=104.52
rigid



Step 2: Intermolecular interactions

Key idea: reproduce the energy of a water dimer.

1. **Basic interacting unit of water**
2. **Hydrogen bonding**
3. **Recent (1973) measures of the structure of the dimer**



TIPS & TIP3P (Jorgensen et al, 1981 - 1983)

$$E_{m,n} = \sum_{i \in m} \sum_{j \in n} \frac{q_i q_j e^2}{r_{ij}} + \frac{A}{r_{OO}^{12}} - \frac{C}{r_{OO}^{6}}$$

$q_O = -2q_H$

LJ only for oxygen

3 parameters

$R_{OO} = 2.98 \pm 0.04$ Å

$\Theta = 60° \pm 10°$

Gas Dipole: 1.85D

$q_O = -0.8, -0.834$

$A^2 = 580 \times 10^3, 582 \times 10^3$

$C^2 = 525, 595$

Microwave spectrum and structure of hydrogen bonded water dimer

Thomas R. Dyke and J. S. Muenter

Department of Chemistry, University of Rochester, Rochester, New York 14627
(Received 27 December 1973)

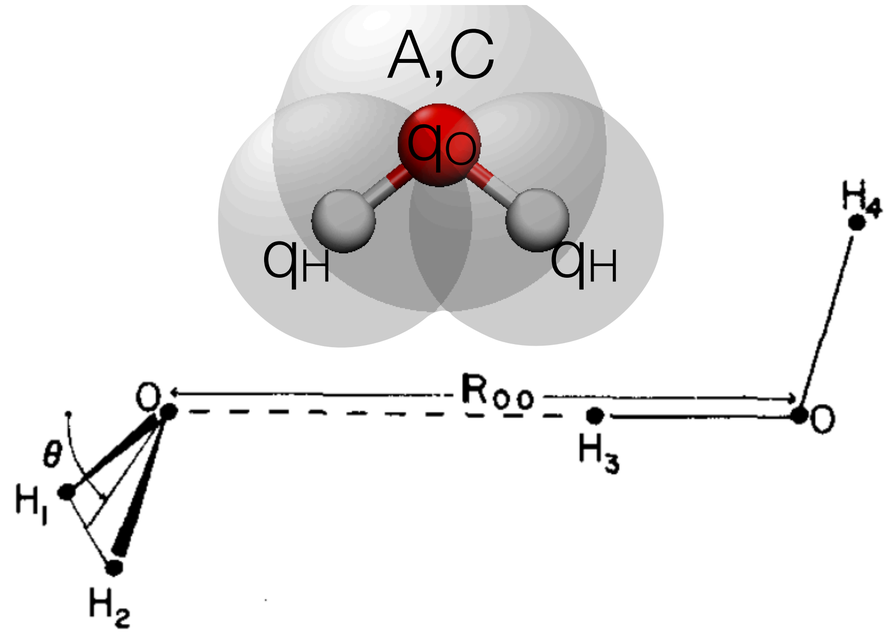


FIG. 1. Structure of water dimer.

TIPS & TIP3P

(Jorgensen et al, 1981 - 1983)

Dimer	TIPS	EXPT
R_{OO}	2,78	2.98±0.04 Å
θ	27	60°±10°
Energy	5,70	5.44±0.7 kcal/mol

Are the results transferable to liquid water?

Table IV. Computed and Experimental Properties of Liquids at 25 °C[a]

water	TIP	exptl[b]
ΔH_V°	8.9	10.7
C_V	15.0	17.9

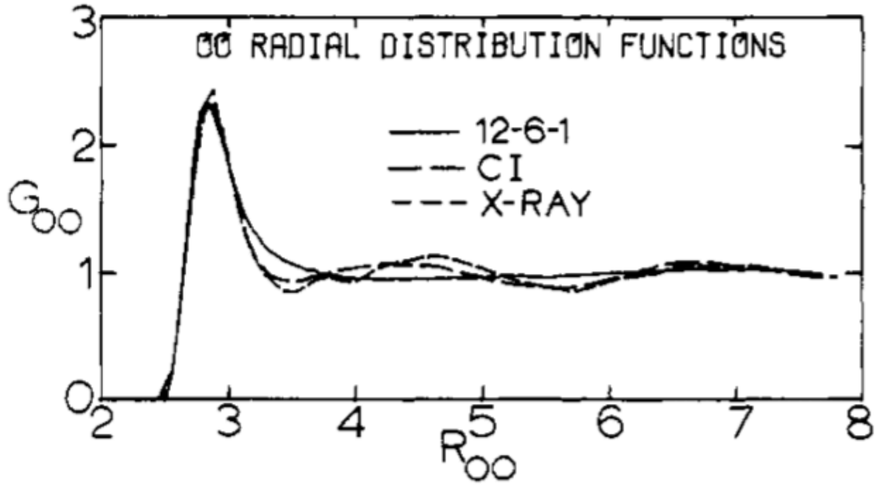


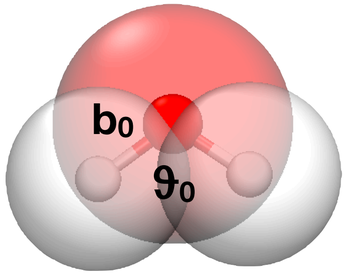
Figure 7. OO radial distribution functions for liquid water at 25 °C from X-ray data (ref 33) and the TIP and CI potential functions.

SPC: Berendsen et al 1981

SPC, key idea: try to reproduce the second peak in the g(r)
and reproduce the Enthalpy of Vaporisation, 2 restraints for 2 parameters.

Step 1: Internal geometry

b_0=1.0 Å
ϑ_0=109.28 **rigid**

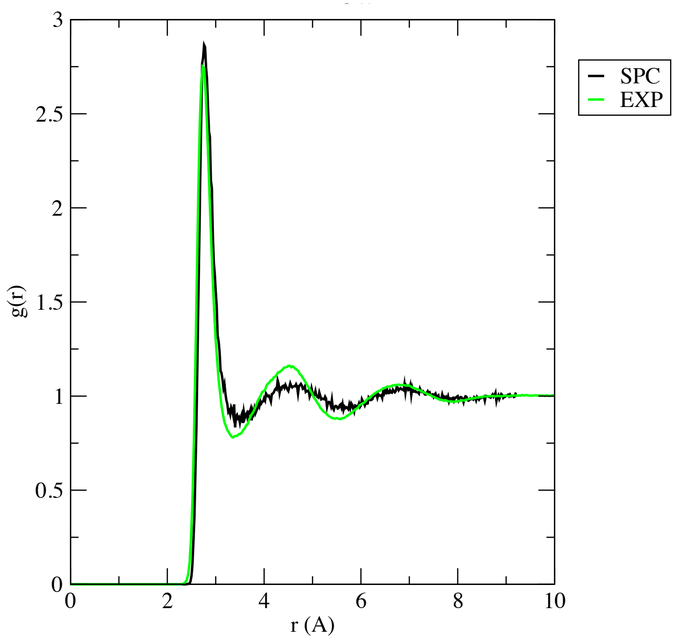


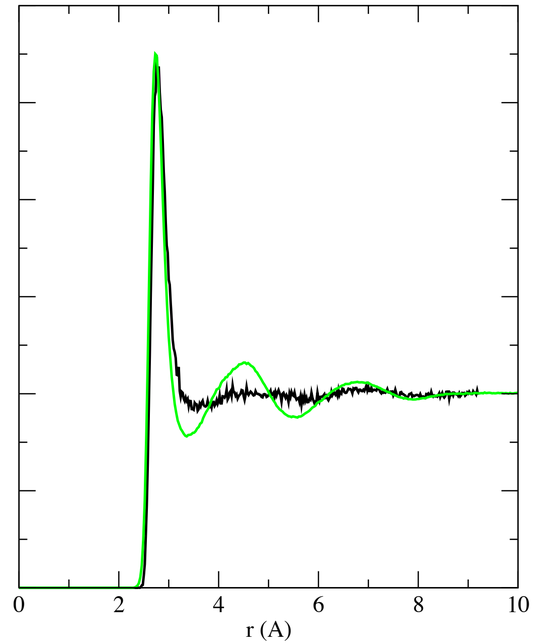
Step 2: Intermolecular interactions

C=0.37122 nm (kJ.mol^{-1})$^{1/6}$ This was derived by London in ~1920

$$E_{m,n} = \sum_{i \in m} \sum_{j \in n} \frac{q_i q_j e^2}{r_{ij}} + \frac{A}{r_{OO}^{12}} - \frac{C}{r_{OO}^{6}}$$

q_O=-0.82
A=0.3428





SPC water can indeed better reproduce the structure of water in solution

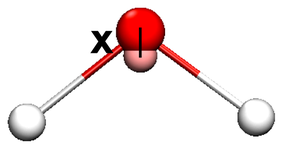
THE JOURNAL OF CHEMICAL PHYSICS **123**, 234505 (2005)

A general purpose model for the condensed phases of water: TIP4P/2005

J. L. F. Abascal[a)] and C. Vega
Departamento de Química Física, Facultad de Ciencias Químicas, Universidad Complutense, 28040 Madrid, Spain

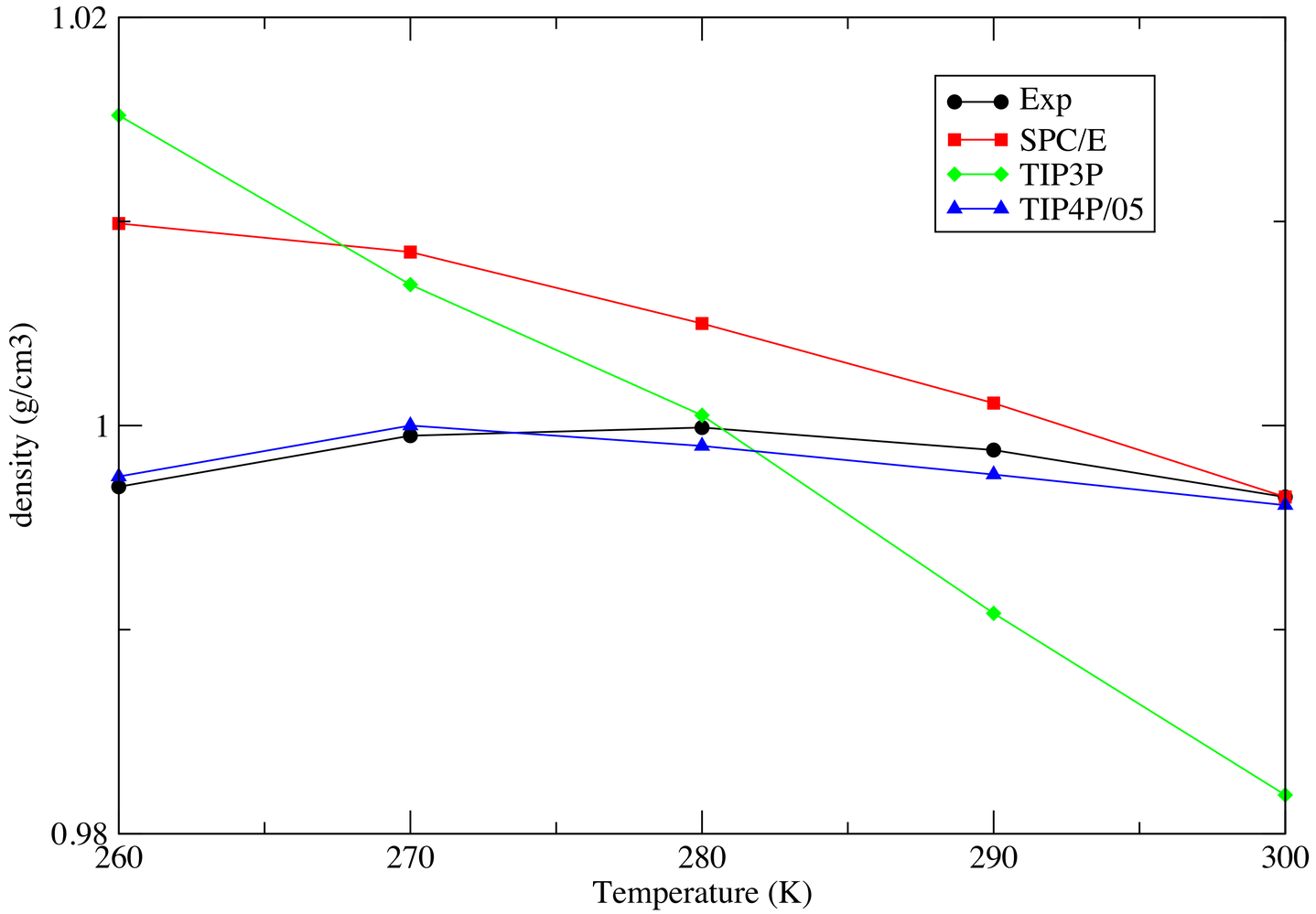
Key ideas: 1) use the water geometry (as in TIPS) 2) use a dislocated positive charge following Bernal - Fowler to better account for the electronic density. 3) use the correction for the enthalpy of vaporisation as in SPC/E; 4) use also properties from ice to train the model.

Systematic optimisation: parameters (q_O, A, C, and x -> this the displacement of the additional particle with respect to the symmetry axis) and properties (T_{md}, Hv, rho(298K), density of ice II at 123 K and 0 MPa, and of ice V at 223 K and 530 MPa, and the range of temperatures at which ice III is the thermodynamically stable ice at a pressure of 300 MPa



b_0=0.967 Å q_O=-1.1128
ϑ_0=104.52 A=3.06010e+06
x=0.1546 Å C=307978

Water Density as a function of the Temperature



Some comparisons of water models properties

Property (298K, 1bar)	TIP3P	SPC/E	TIP4P/05	TIP4P/ε	EXP
Hv (kcal/mol)	10.2 (9.0)	11.6 (10.4)	11.9 (10.8)	11.8 (**10.7**)	**10,7**
Cp (cal/mol/K)	**18,5**	19,0	21	21	**17,9**
Density (g/cm^3)	0,984	**0,998**	**0,996**	0,995	**0,997**
Diffusion (cm^2/s)	5,6	2,7	**2,3**	**2,3**	**2,3**
Dielectric constant	94	68	58	**78,5**	**78,5**

Force-Fields: from energy to forces

In physics forces are defined as the negative gradient of the energy, so if we know an energy function, a function that given an atomic configuration gives us an energy, we can calculate the forces on the atoms for that configuration:

$$\boldsymbol{F}(\boldsymbol{R}) = -\nabla V(\boldsymbol{R})$$

Here **bold** characters denote a Vector or more general a list of vectors, for example in the case of a force-field applied to alanine dipeptide it will be a list of 22 vectors, each one pointing to an atom, while standard characters denote a Scalar (a number)

How to:

$$\nabla f = \frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j} + \frac{\partial f}{\partial z}\mathbf{k},$$

$$f(x, y, z) = 2x + 3y^2 - \sin(z)$$

$$\nabla f = 2\mathbf{i} + 6y\mathbf{j} - \cos(z)\mathbf{k}.$$

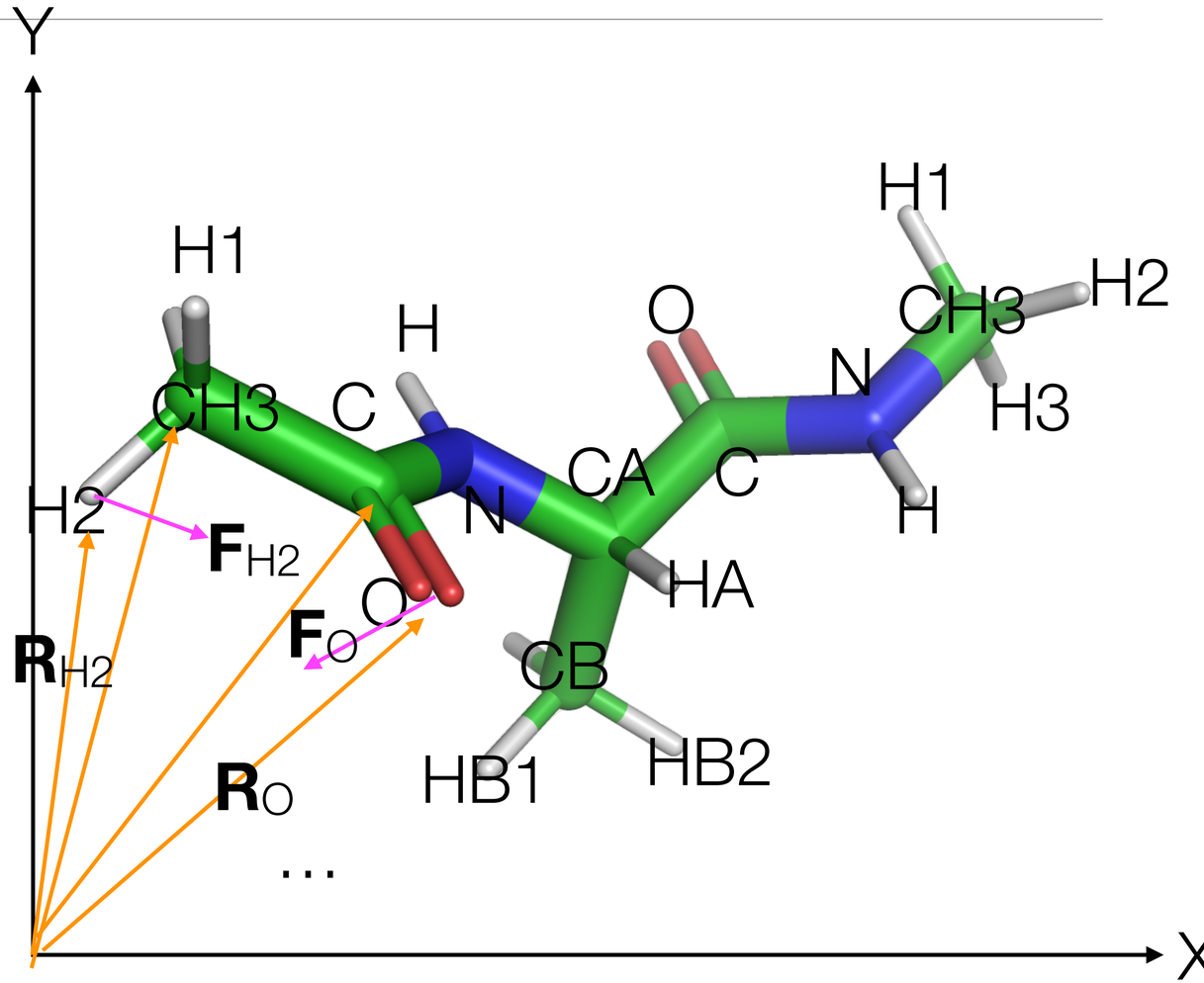
The gradient is the direction of maximum growth of the curve.

Force-Fields: from energy to forces

$$\boldsymbol{F}(\boldsymbol{R}) = -\nabla V(\boldsymbol{R})$$

The gradient is the direction of maximum growth of the curve, so negative gradient will results in one vector per atom, centered on the atom and pointing in the direction of maximum decrease of the energy. The forces will indicate in which direction the atom should move to lower the overall energy of the complete system.

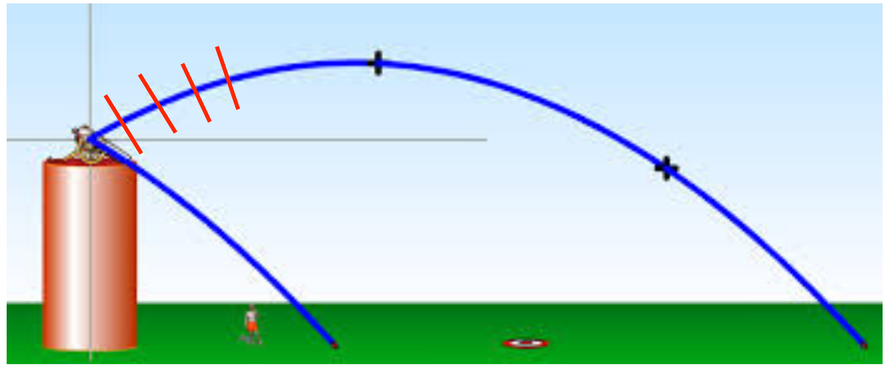
In the figure **orange arrows** represent **positions** vectors for some of the atoms, while **magenta arrows** represent **force** vectors for some of the atoms.



Molecular Dynamics: from forces to motion

Second Newton's Law: the action of the total force **F** on a body produces an acceleration **a** equal to the total force divided by the mass *m* of the body: **a**=**F**/m

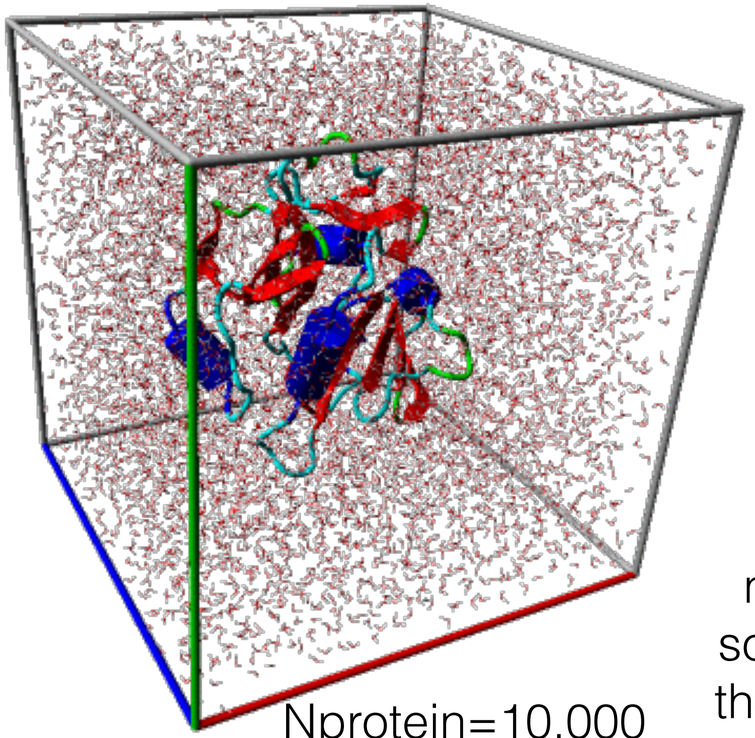
In principle my forces will change continuously with the motions of the atoms, but we want to solve an iterative problem where atoms are moved by small variations, or alternatively, we want to move atoms forward in time only for a very small time step, how small? Somehow compatible with the time scale of atom motions ~fs (10^{-15} s), on this time scale we can consider the forces as constant.



Would be like solving simple physics problems for small steps

When forces are constant we know how bodies move: **uniformly accelerated motion**

The simulation box determines the size so the speed of your MD



Nprotein=10,000
Nsolvent=55,000

$$V(r) = \sum_{bonds} k_b(b - b_0)^2 + \sum_{angles} k_\theta(\theta - \theta_0)^2 + \sum_{torsions} k_\phi[\cos(n\phi + \delta) + 1]$$

$$+ \sum_{\substack{nonbond \\ pairs}} \left[\frac{q_i q_j}{r_{ij}} + \frac{A_{ij}}{r_{ij}^{12}} - \frac{C_{ij}}{r_{ij}^{6}} \right]$$

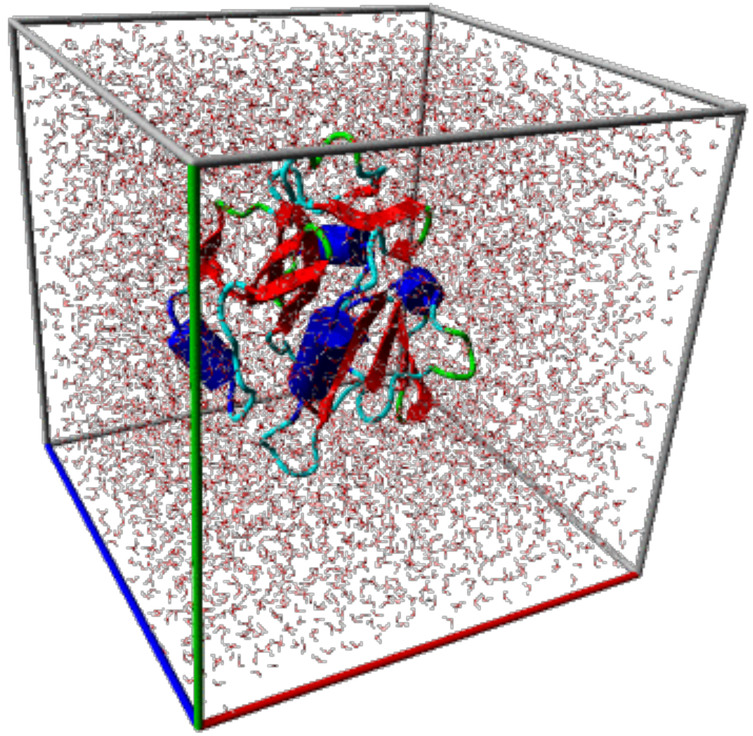
To calculate the energy and thus the forces we need to calculate distances between all pairs of atoms (N^2 calculations)

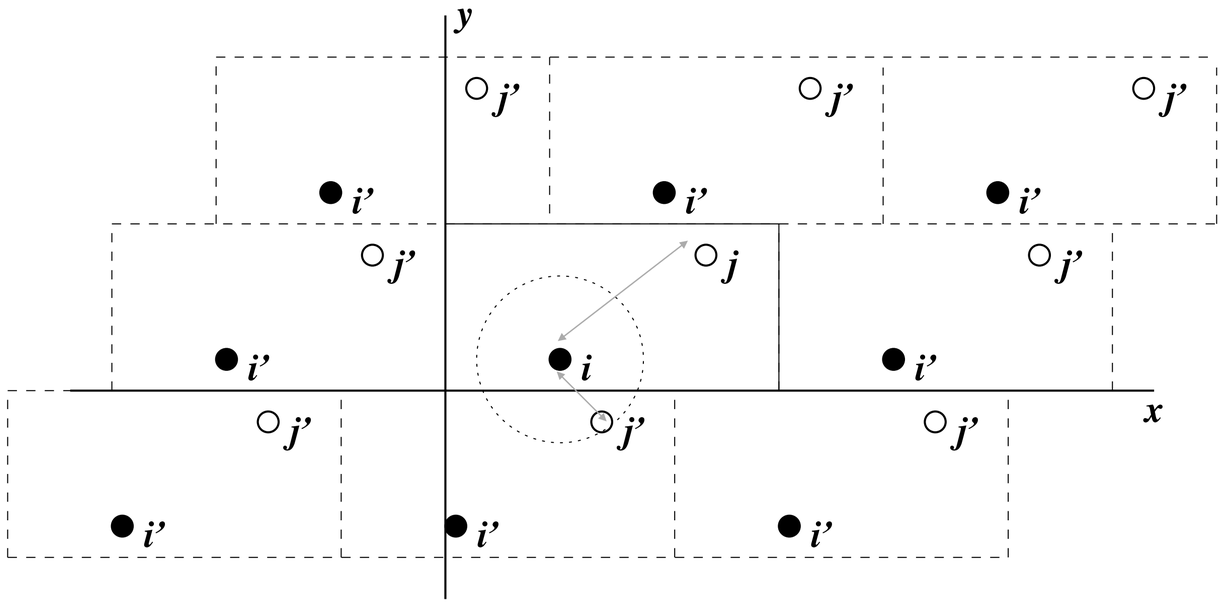
So how big should we set the box? Consider that we want to simulate molecules in their solution environment so the number of atoms is not only that of the molecule of interest but also that of all the solvent atoms.

Furthermore the size of the box, if the box is like a cuvette can generate artefacts due to the protein-box sides interactions.

Does the protein have the space to experience conformational changes?

Periodic Boundary Conditions allow to simulate boxes without borders

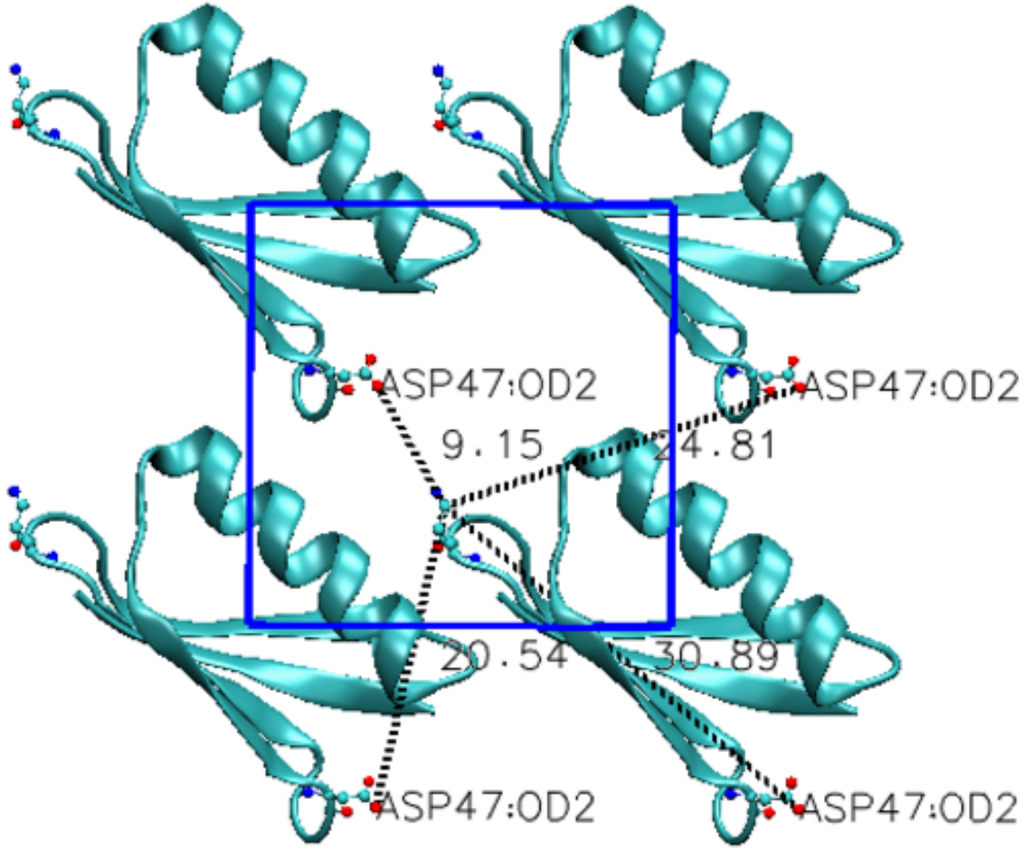




By defining PBC at least in principle we can simulate a molecule like it is ideally diluted, but if the box is periodic the distance between two atoms can assume an infinite number of values, what is the “correct” distance?

Minimum Image Convention: the distance between to atoms is always the minimum with respect to all the neighbour images interactions can only be calculated up to a distance equal to half of the shortest side of the box

Minimum Image Convention requires a minimum box size



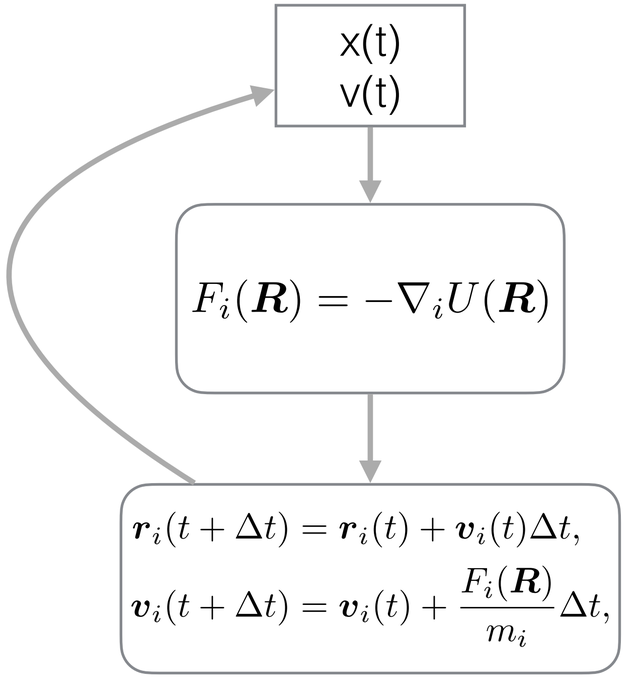
For example one can try to make the box large enough so that the intramolecular distance is generally shorter than the corresponding intermolecular distance. This is not the case of what is shown on the left. Nonetheless the above approach may be too computationally demanding.

A simple MD algorithm: The Euler algorithm

$$\boldsymbol{v}_i(t+\Delta t) = \boldsymbol{v}_i(t) + \frac{F_i(\boldsymbol{R})}{m_i}\Delta t,$$

$$\boldsymbol{r}_i(t+\Delta t) = \boldsymbol{r}_i(t) + \boldsymbol{v}_i(t)\Delta t$$

For each atom *i*, velocities are updated using forces and are used to update positions, and each iteration of the algorithm is a step forward in time of size Δt



1. starting position and velocities
2. calculate forces
3. calculate new positions and velocities

The microscopic world:

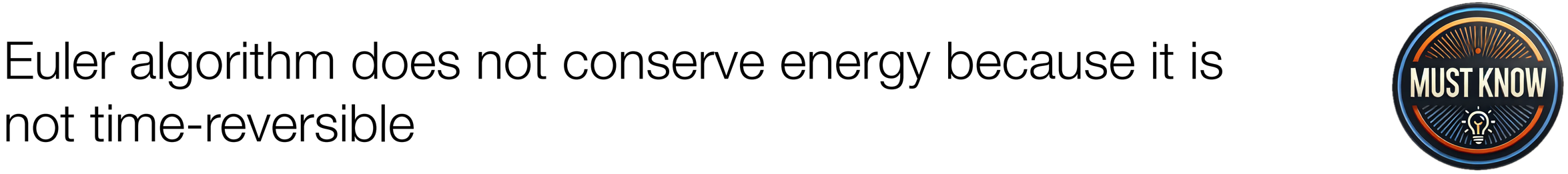
Distances in nm

Times in ps

Energies in kJ/mol

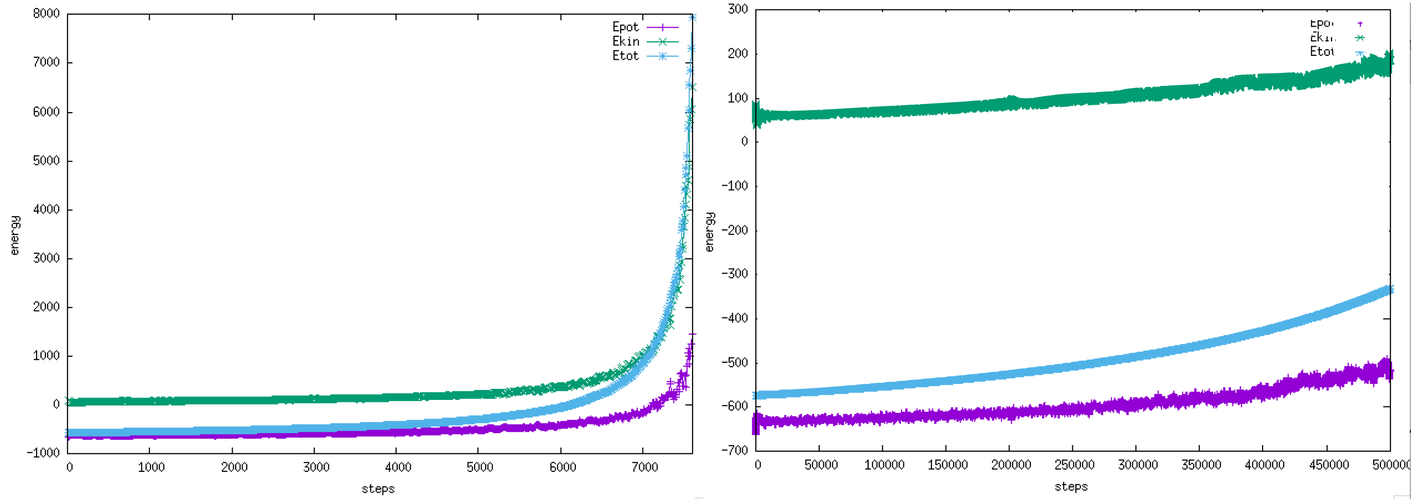
To initiate the algorithm you need some initial position and velocities and a force-field.

Euler algorithm does not conserve energy because it is not time-reversible



1. If a force does not depend explicitly on time (is conservative) then **the Total Energy of the system is conserved.**

2. If a force does not depend explicitly on time (is conservative) then there is a symmetry for time inversion, i.e. going back in time is equivalent to invert the velocities.



The Euler algorithm cannot conserve the energy, there is not a time step for which the algorithm can work correctly.

$$
\begin{aligned}
r(t+\Delta t-\Delta t) &= r(t+\Delta t) - v(t+\Delta t)\Delta t = \\
&= r(t) + v(t)\Delta t - v(t+\Delta t)\Delta t = \\
&= r(t) + v(t)\Delta t - v(t)\Delta t - a(t)\Delta t^2 = \\
&= r(t) - a(t)\Delta t^2 \neq r(t)
\end{aligned}
$$

A better Molecular Dynamics algorithm: Velocity Verlet

$$\vec{v}\left(t+\tfrac{1}{2}\,\Delta t\right)=\vec{v}(t)+\tfrac{1}{2}\,\vec{a}(t)\,\Delta t.$$

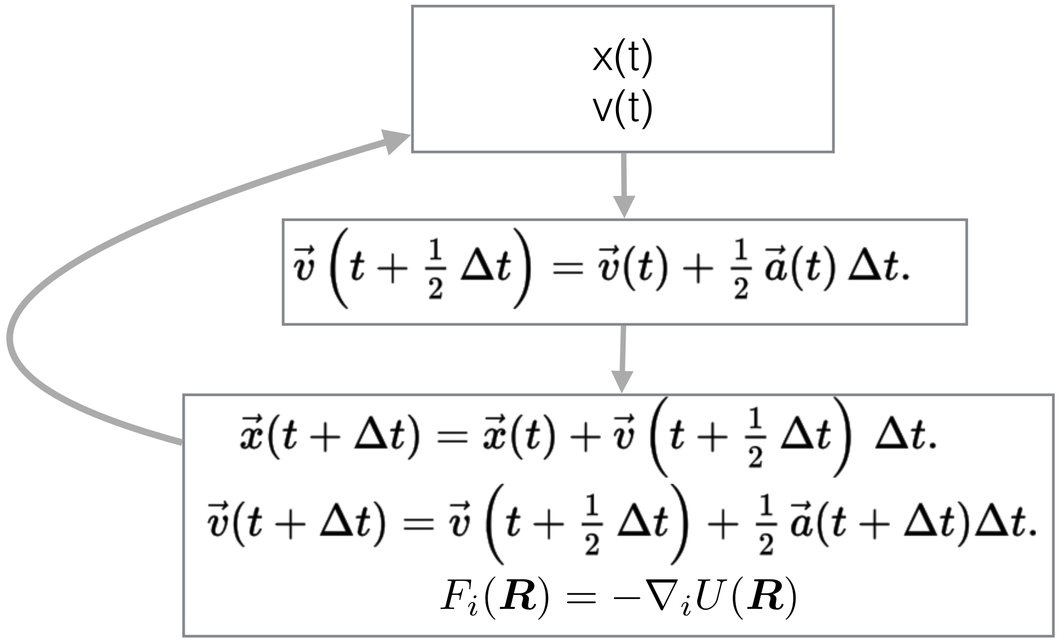
$$\vec{x}(t+\Delta t)=\vec{x}(t)+\vec{v}\left(t+\tfrac{1}{2}\,\Delta t\right)\,\Delta t.$$

The idea is to calculate more accurate velocities
to be more precise in calculating positions

The algorithm

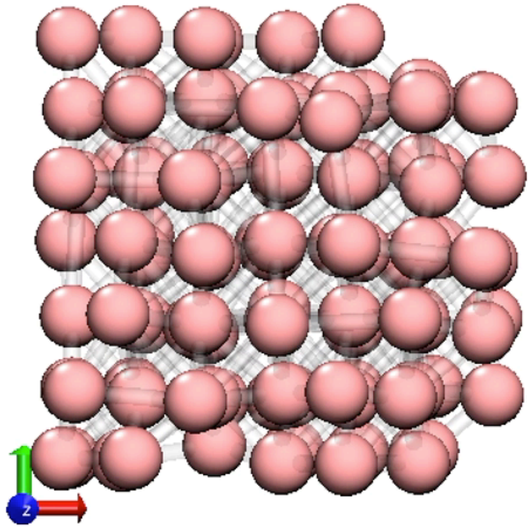
Start: x(0), v(0), f(0)

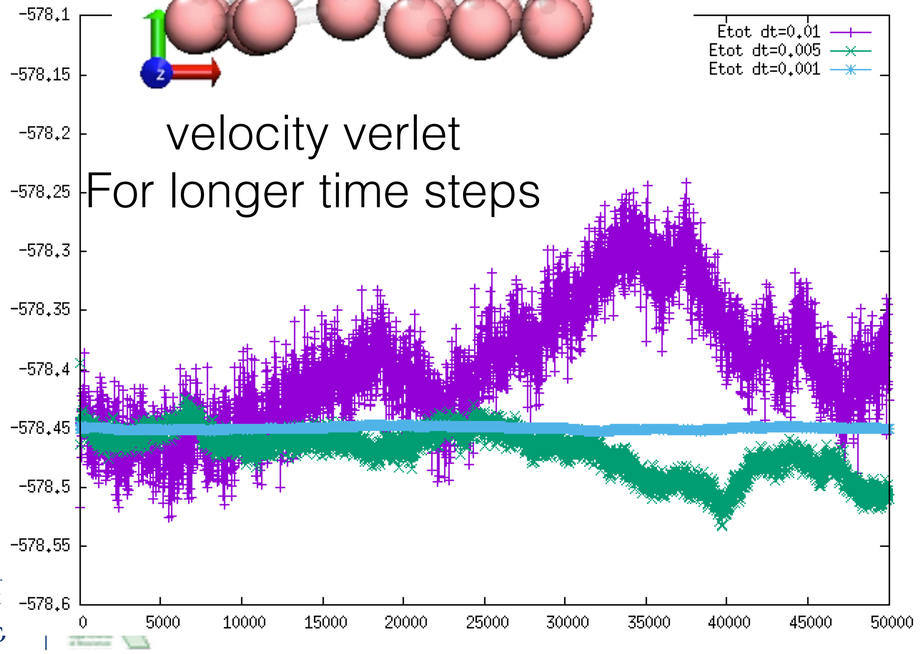
1. calculate velocities at time t+0.5dt: v(t+0.5dt)
2. calculate positions at time t+dt: x(t+dt)
3. calculate forces at time t+dt: f(t+dt)
4. calculate velocities at time t+dt: v(t+dt)



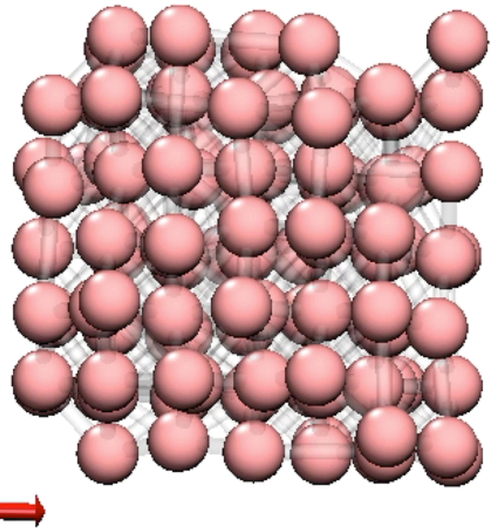
$$\begin{aligned}
r(t+\Delta t-\Delta t) &= r(t+\Delta t)-v(t+\Delta t)\Delta t+\frac{1}{2}a(t+\Delta t)\Delta t^2= \\
&= r(t)+v(t)\Delta t+\frac{1}{2}a(t)\Delta t^2-v(t)\Delta t+ \\
&-\frac{1}{2}a(t)\Delta t^2-\frac{1}{2}a(t+\Delta t)\Delta t^2+\frac{1}{2}a(t+\Delta t)\Delta t^2= \\
&= r(t)
\end{aligned}$$

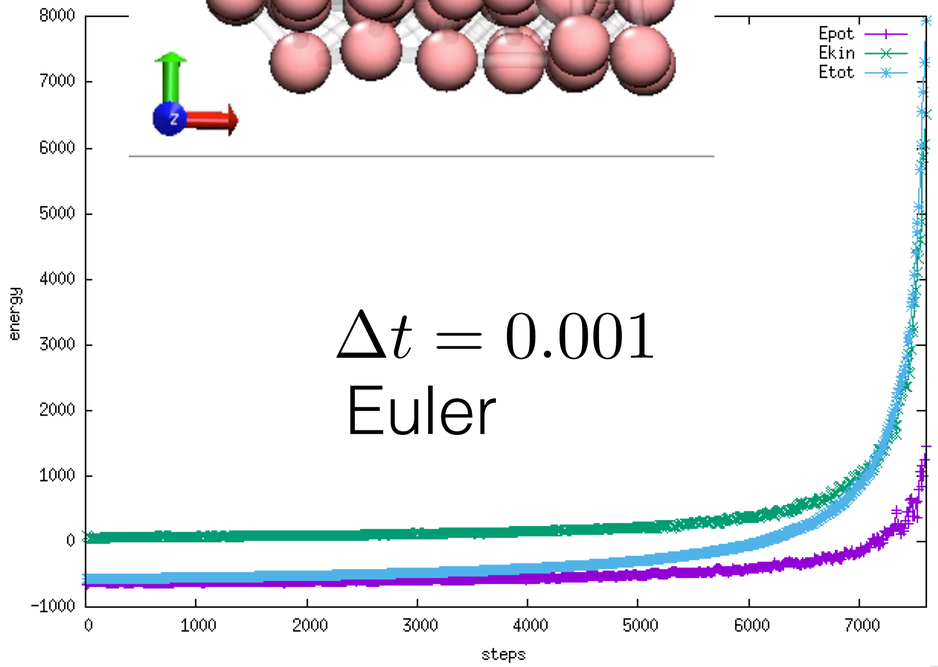
MD by Velocity Verlet can conserve the total energy





velocity verlet
For longer time steps





$\Delta t = 0.001$
Euler

MD: setting the temperature

If the goal of MD is to calculate an average quantity it does not matter if we calculate it at constant energy, or temperature, or other thermodynamic ensembles. In the thermodynamic limit average quantities do not depend on this choice. Fluctuation do, so if we are interested in probability distributions we need to chose the experimental conditions.

Equipartition Theorem gives us a definition for the temperature: If you set the temperature of a system its average kinetic energy should be equal:

$$\langle K \rangle = \frac{3}{2} NkT$$

We can thus define the instantaneuos temperature and the average temperature as:

$$T(t) = \frac{1}{3Nk_B} \sum_i^N m_i v_i(t)^2 \qquad \langle T \rangle_{NVT} \approx \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} \frac{1}{3Nk_B} \sum_i^N m_i v_i(t)^2 dt$$

We want an algorithm to obtain the correct average kinetic energy with an accuracy that is the standard error of the mean over our number of particles.

MD: the stochastic velocity rescaling algorithm

$$K_0 = \frac{3}{2} N_f k T,$$

$$K(t+\Delta t) = K(t) + (K_0 - K(t))\frac{\Delta t}{\tau_T} + 2\sqrt{\frac{K(t)K_0}{N_f}}\frac{dW}{\sqrt{\tau_T}},$$

$$v(t+\Delta t) = v(t+\Delta t)\sqrt{\frac{K(t+\Delta t)}{K(t)}}$$

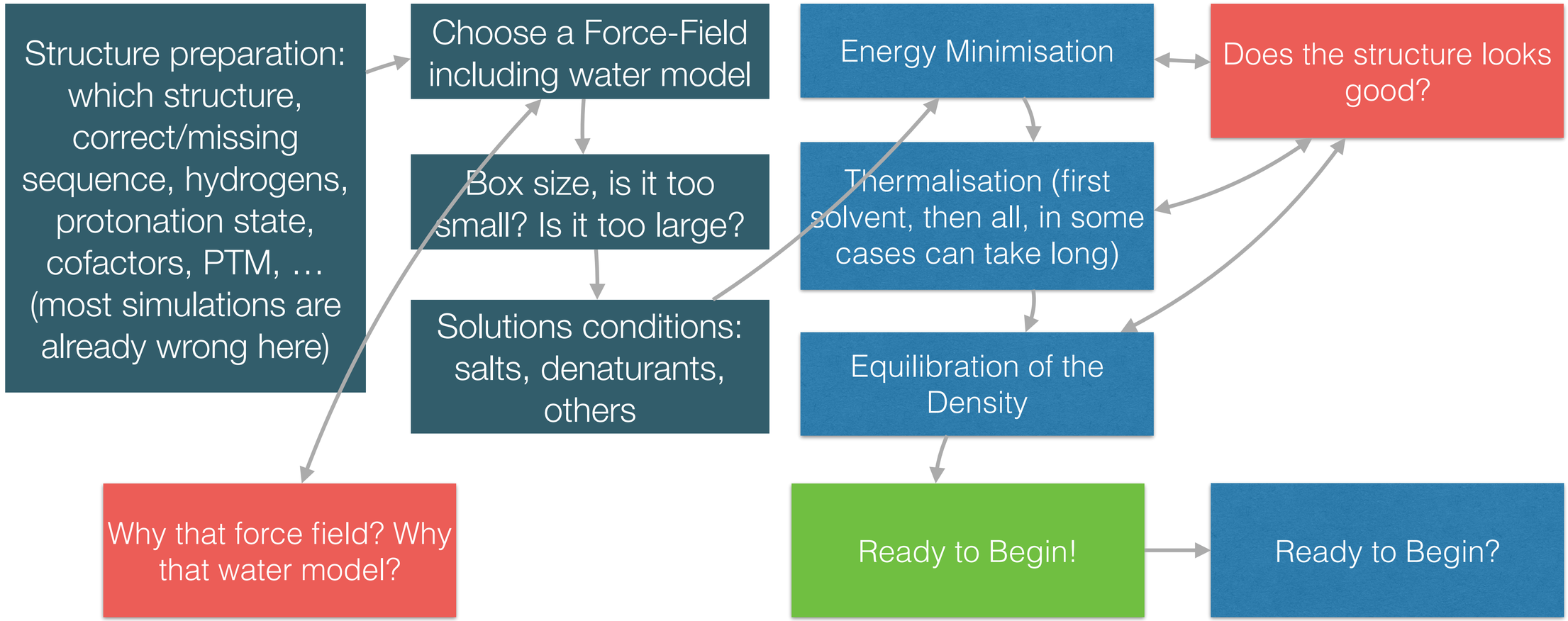
By setting the temperature we know what should be on average the value of the kinetic energy of the system. So we let the kinetic energy evolve with fluctuations inversely proportional to the number of atoms Nf. Once we set the new kinetic energy we modify the velocities to match it.

At step 0 compute forces **{x(0), v(0), f(0)}**

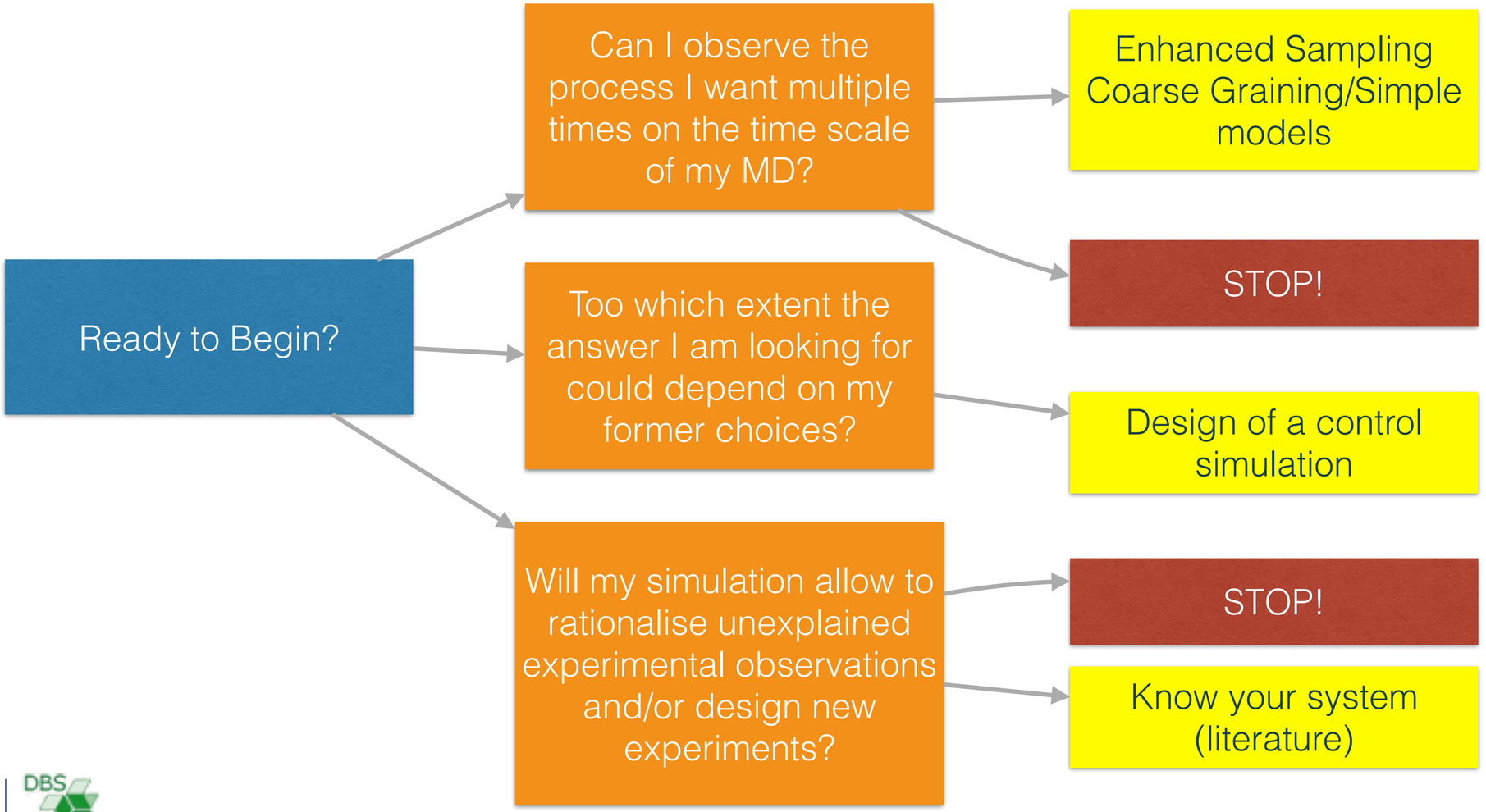
1. update kinetic energy
2. update velocities
3. update positions
4. calculate forces
5. update velocities
6. update kinetic energy

A similar approach can be used to also set the pressure

A general scheme to setup MD simulations:



A general scheme to setup MD simulations:

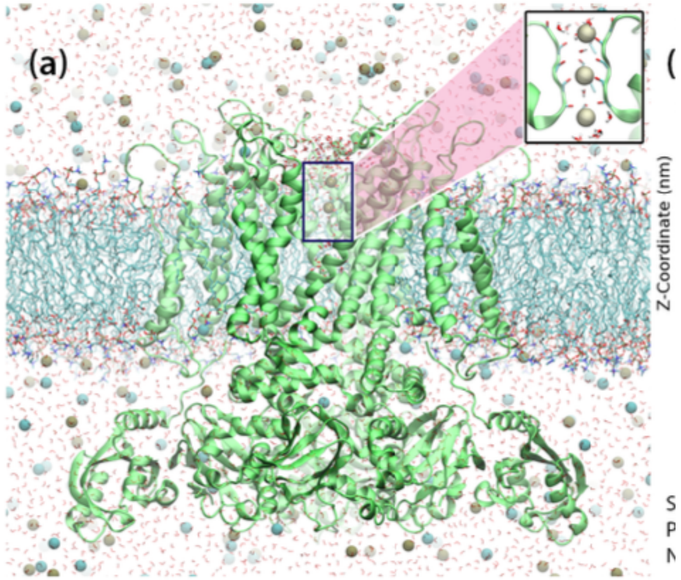


MD simulations: what does determines the accessible time scale

The typical time step of MD simulations is between 1-4 fs, most often 2 fs. So one iteration of the algorithm produces a frame dt after the previous. The number of iterations or steps determines the length of your simulation (performances are measured as ns/day).

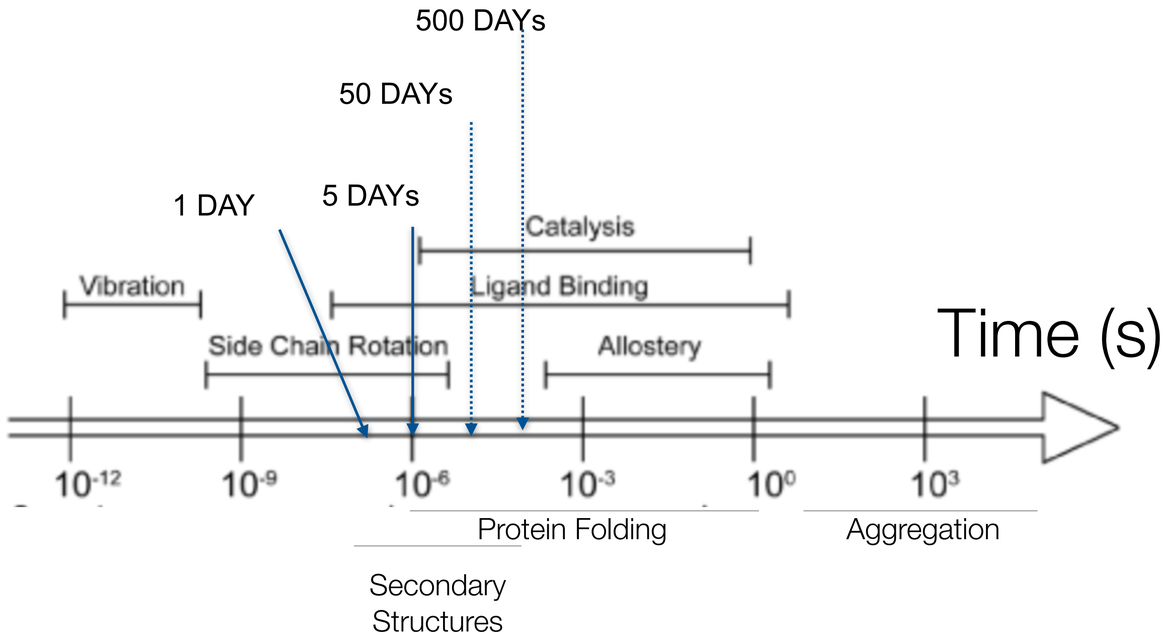
The factors that determines the performances are the box-size and number of atoms, a simulation can be ran faster by using more cores (cpu or gpu), and then more computing nodes, by parallelisation. Parallelisation can be done by dividing your box in parts (domain decomposition) and/ or subdividing sets of atoms (particle decomposition)

Nodes	CPU (ns/day)	GPU (ns/day)
1	1	7
2	2	14
4	4	27
8	8	49
16	15	88
32	30	136
64	58	183



For a system of 500,000 atoms (including solvent)

MD simulations: time step, time scales and probabilities



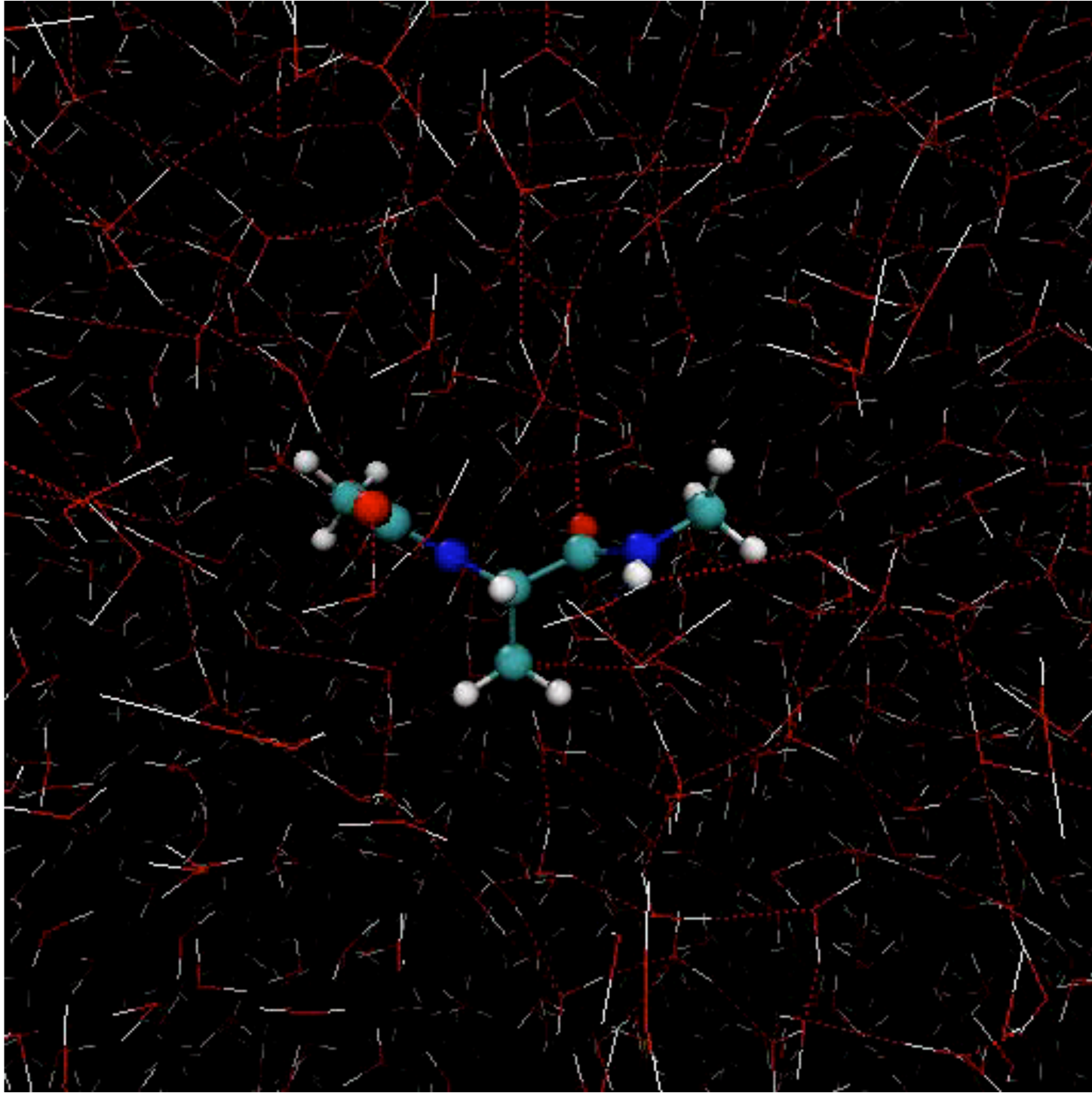
So if we can run our simulation at ~200 ns/day

the probability of observing an event with a rate of 1 ms with a simulation of 10 us is ~1% in the case of a two state kinetics

How many simulations can we run in parallel?

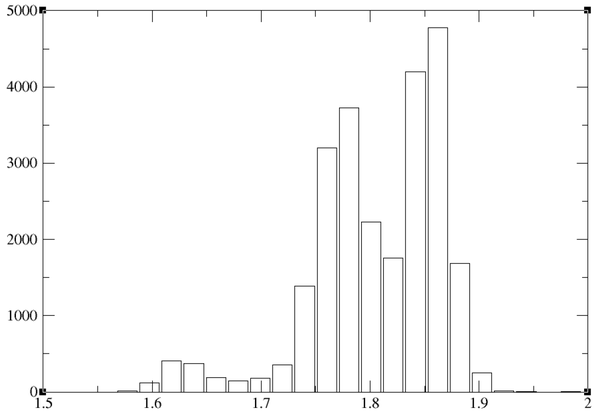
Ideally, what should a computational microscope do?

- Observe the time evolution of molecules at high spatial and time resolution ✅
- Observe them for very long time scales ⚠️ (Time scale depends on size, improving…)
- Be able to set different experimental conditions (Temperature, Pressure, solution conditions) ✅
- Be accurate ⚠️ (Force-field dependence, but improving…)
- Be interpretable, that is find suitable macrostates to compare with experiments ⚠️ (see later)
- …

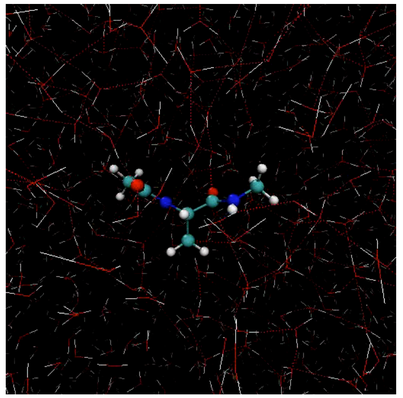


Analysis and Interpretability of MD simulations

A key difference between a simulation and an experiment is that when you do an experiment you have already selected which property of the system to observe, in a simulation instead you set up the conditions of the simulation but then you need to work on how/what to look for.



This is the histogram of the x coordinate of the Ca carbon of the alanine

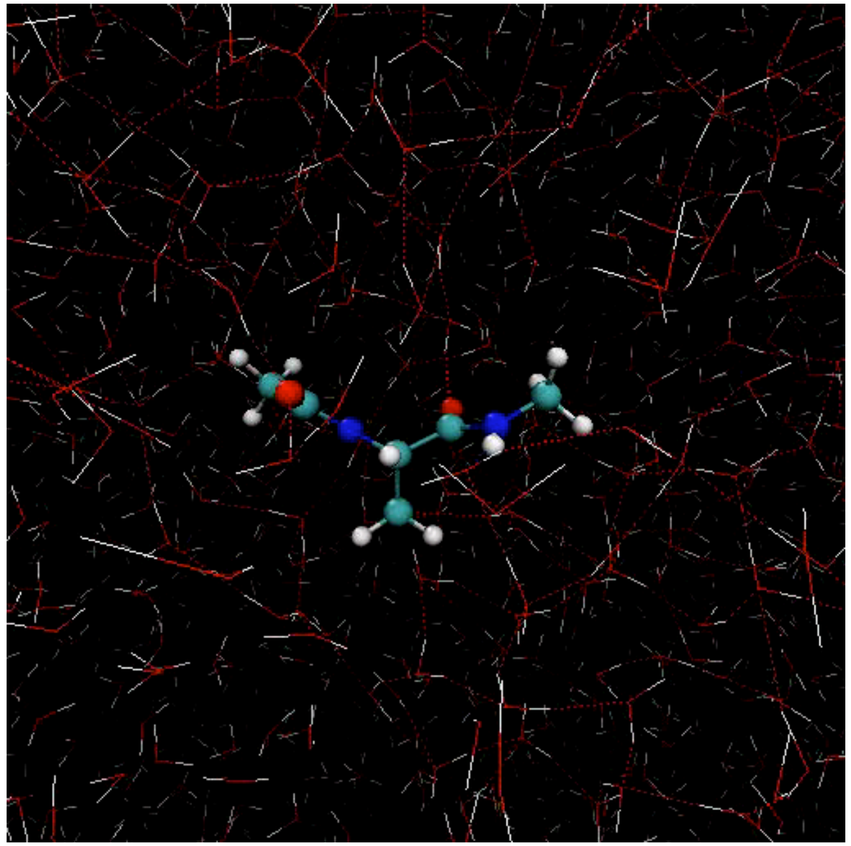


In principle we can look at any atomic quantity, but they are not necessarily interesting, for example the position of an atom in the box…

A very important feature of MD simulations once you decide what to look for is that configurations evolve following the energy gradient and this means that they already appear following Boltzmann statistics.

And a key point is that we have the TIME EVOLUTION of the quantities we look like in a single-molecule experiment.

Analysis and Interpretability of MD simulations



Then, given that we have positions, velocities and forces for all the atoms this means that we can analyse any property that can be defined using these quantities:

Geometric quantities like: distances, dihedral angles, H-bonds, Volume, Density, Secondary Structures, RMSD root mean square deviation of the positions with respect to a reference configuration, RMSF root mean square fluctuations of the positions with respect to a reference configuration, Gyration radius, the radius of the sphere including the system atoms, …

Force, Velocity related quantities like: Energy, Pressure, Diffusion, …

Experimental Observables like: NMR quantities as J-Coupling, NOE and chemical shifts can be calculated using approximate functions, SAXS/SANS, FRET, CD, …

Analysis and Interpretability of MD simulations

Choosing the quantities to analyse get into two common statistical approaches:

1) **clustering techniques**; and
2) **dimensionality reduction techniques**

Clustering techniques are techniques whose goal is to classify data in sets with a label, dimensionality reduction techniques are instead techniques that looks for optimal projections of highly dimensional data into a limited number of dimensions. The two approaches can also be combined together.

If we take as an example the alanine dipeptide simulation in water, there we have 22 atoms belonging to aladp and 1554 atoms of water (518 water molecule), for a total of 1576 atoms and 4728 coordinates. This means that the problem can be consider as a problem in 4728 spatial dimensions.

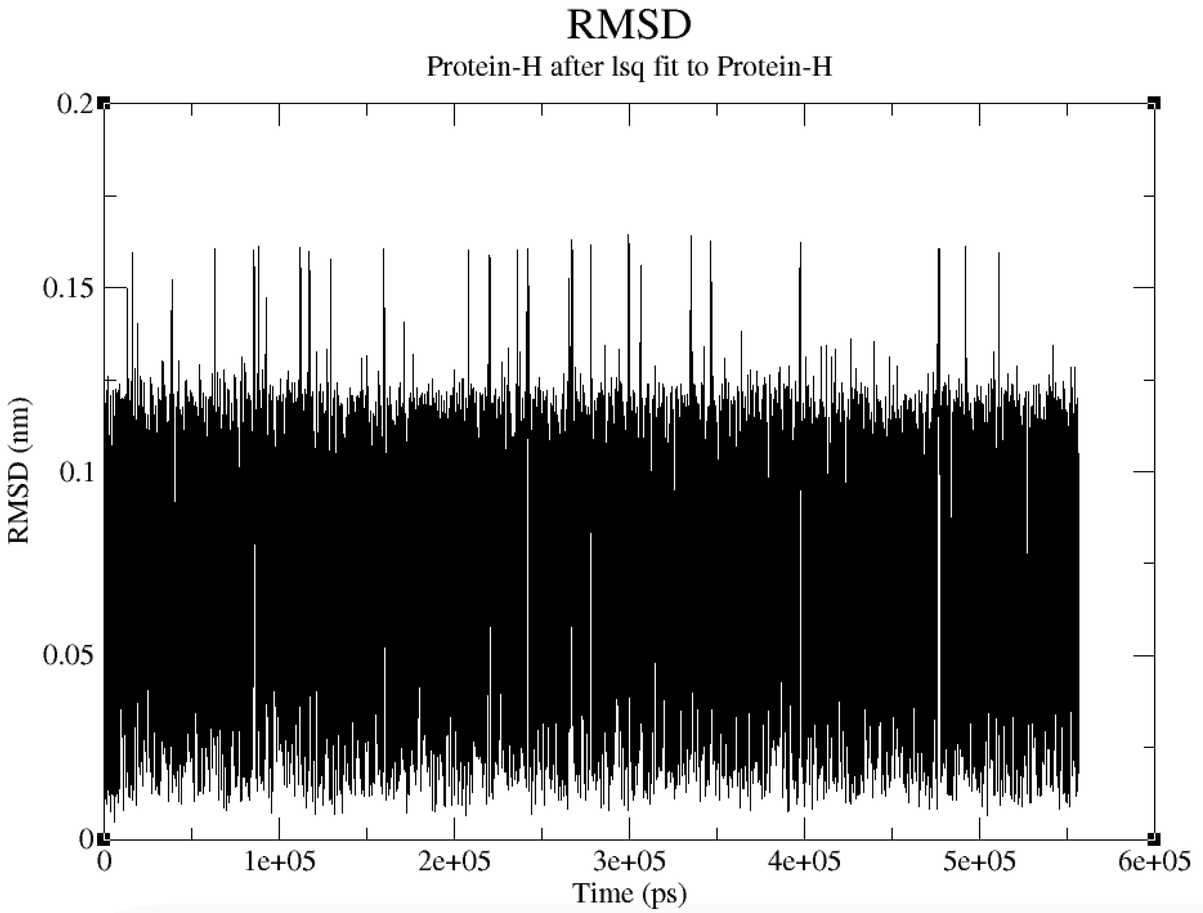
To Learn MORE: Glielmo, A. *et al*. Unsupervised Learning Methods for Molecular Simulation Data. *Chem Rev* (2021) doi:10.1021/acs.chemrev.0c01195.

A simple case: Alanine Dipeptide

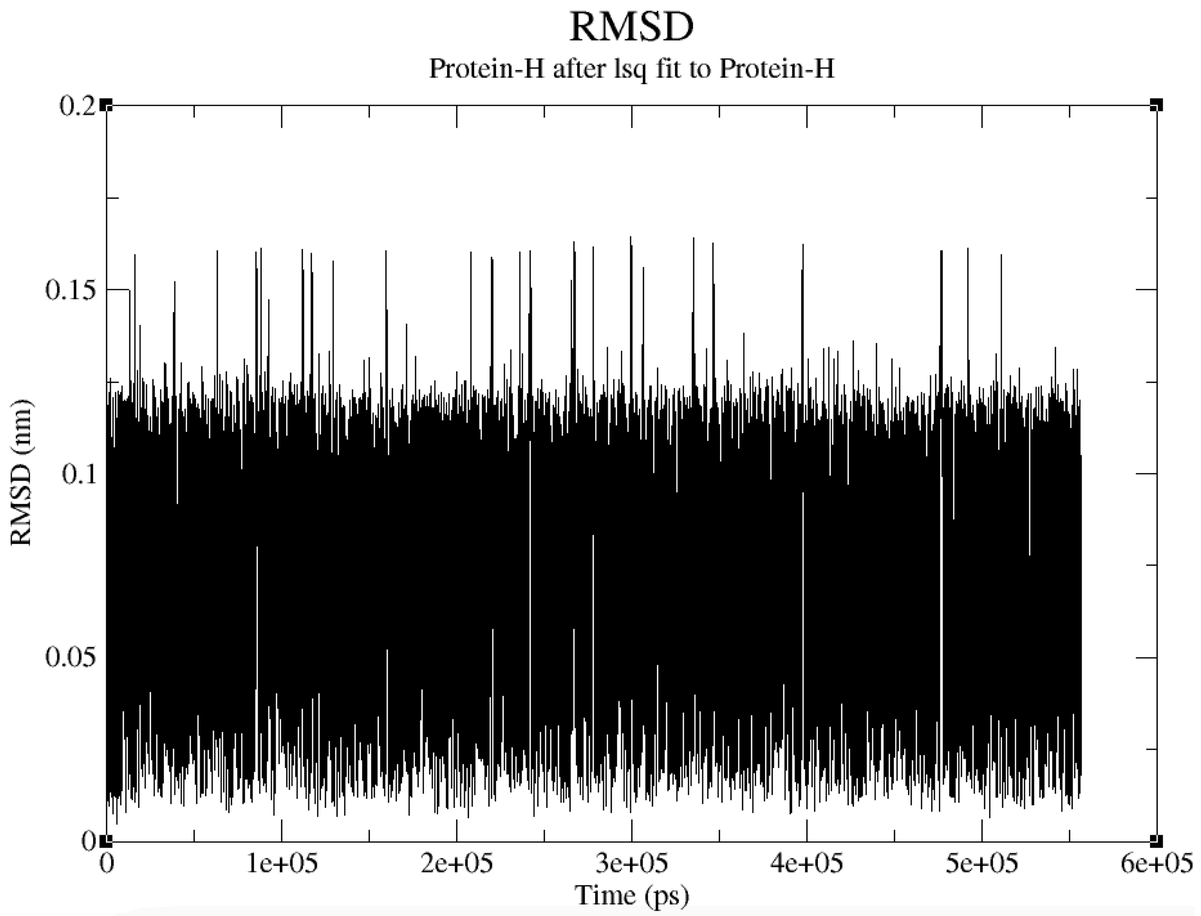
To do an analysis we should in general first have done a simulation to answer some questions, this should already guide us in the analysis to perform. Let's say that we want to learn about the conformational freedom of alanine dipeptide.

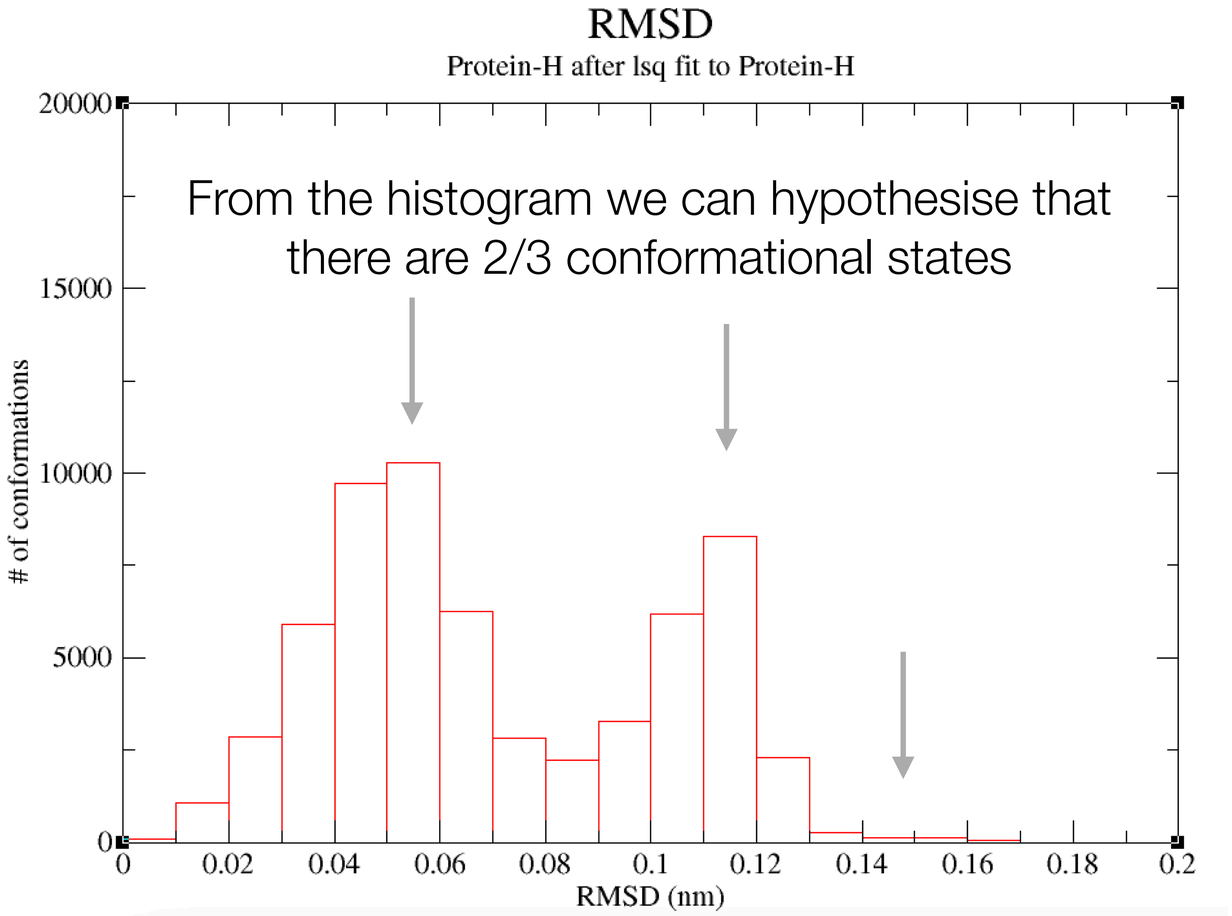
We may start looking at the **RMSD** of the molecule in time. The RMSD measures how much each conformation in time differ from a reference configuration (for example the first frame of the trajectory) by comparing the difference in the atomic position after optimally superimposing the two configurations.

https://en.wikipedia.org/wiki/Root-mean-square_deviation_of_atomic_positions



A simple case: Alanine Dipeptide



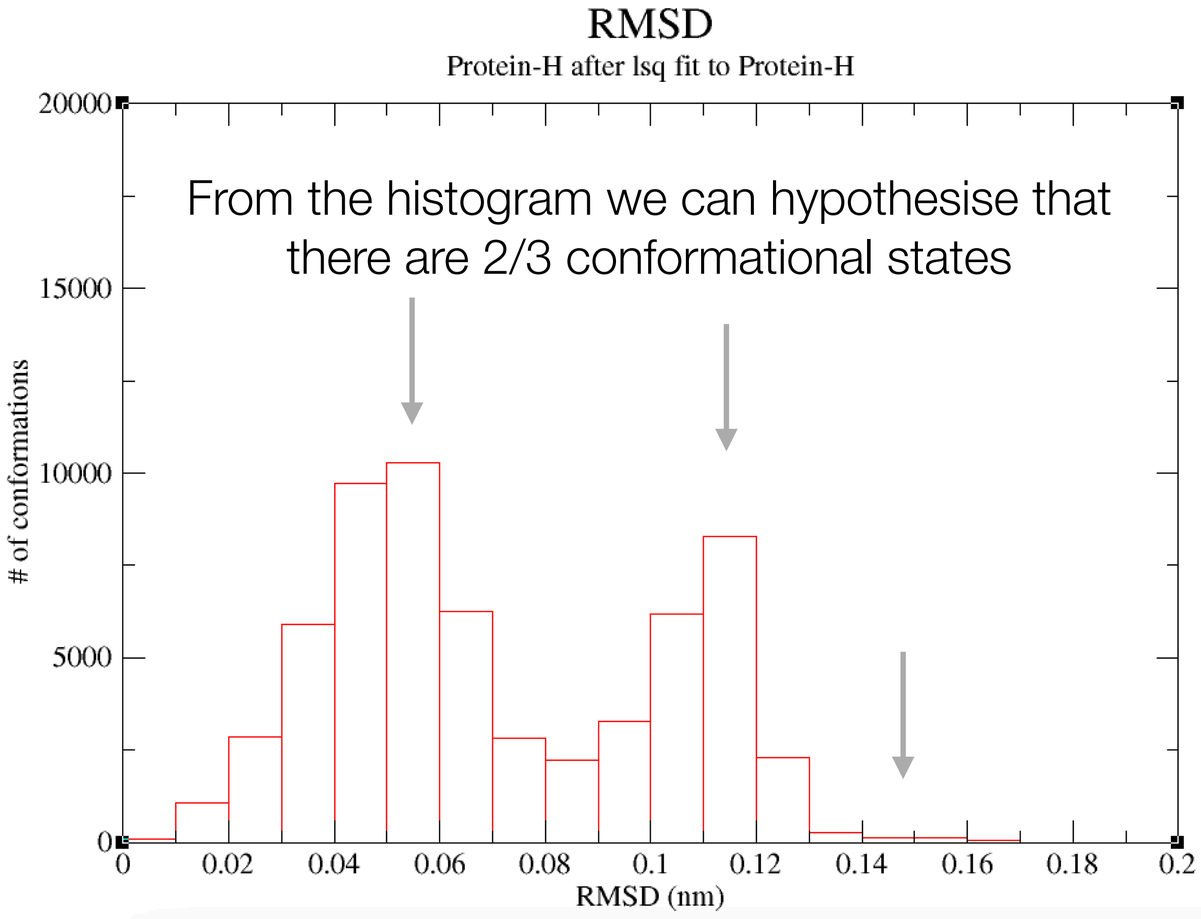


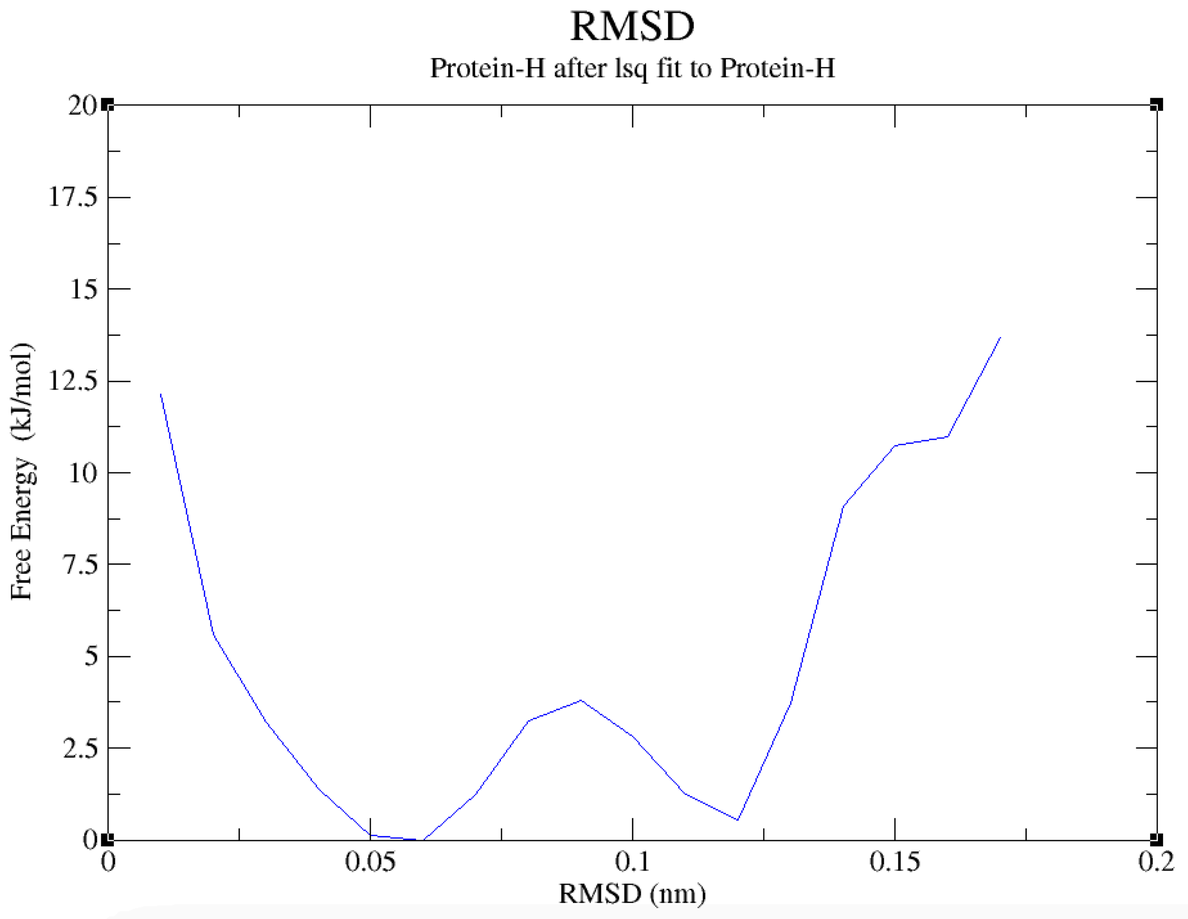
UNIVERSITÀ DEGLI STUDI DI MILANO

DBS

From our time-resolved data we can make an histogram counting each conformation as 1 because in MD conformations appear following Boltzmann probability.

A simple case: Alanine Dipeptide

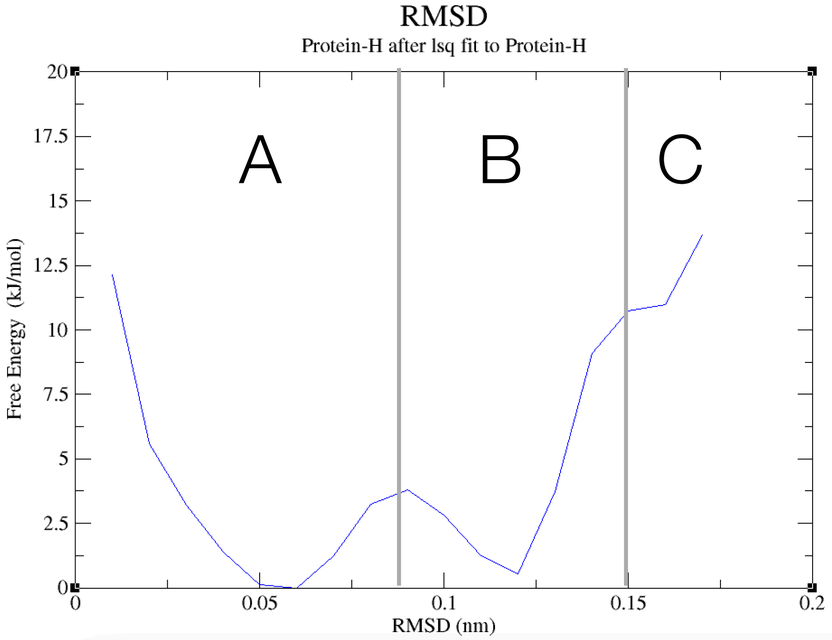




From the histogram we can estimate the Free Energy profile along the RMSD variable by taking F(rmsd) = -kbT ln(count(rmsd))

AlanineDP: From dimensionality reduction to clustering

What we have done is a dimensionality reduction of the conformational space of our simulation from the many dimension of the coordinate to a single dimension, that is also collective variable. This may or may not be important to describe the conformational dynamics of ALADP, but allows to identify 2/3 states. In principle we can assign to all conformations belonging to the three state a label and so clusterize our trajectory:



Usually we say that the resulting clusterisation is better if the structures belonging to the cluster are overall homogenuos with respect to some property.

Furthermore, the number of configurations per state, divided by the total number of configuration will give us a state population:

pA=#A/(#A+#B+#C) F_A=-k_BT ln(pA)
pB=#B/(#A+#B+#C) F_B=-k_BT ln(pB)
pC=#C/(#A+#B+#C) F_C=-k_BT ln(pC)

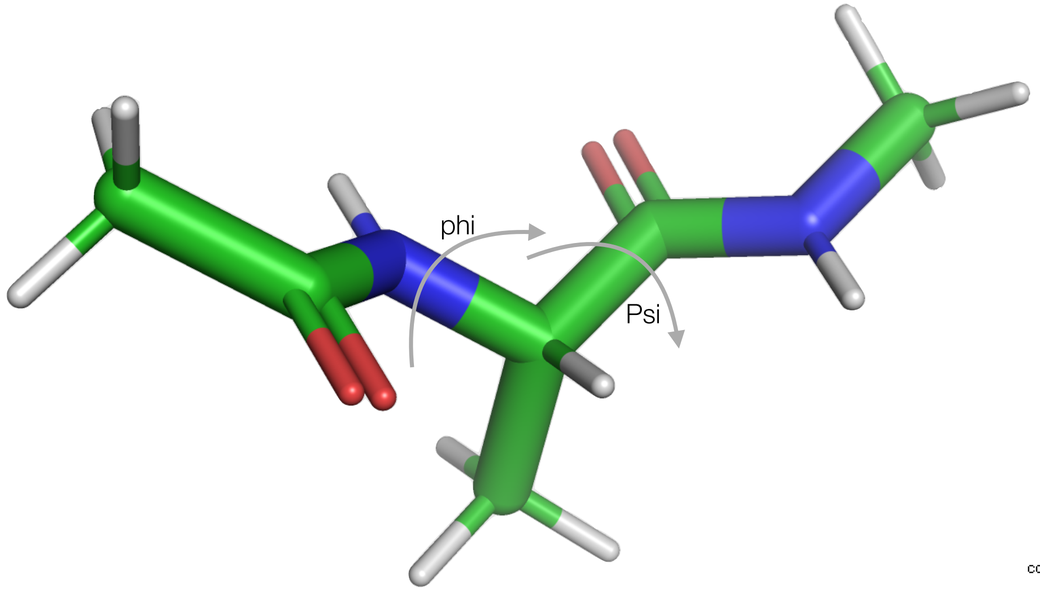
$$\Delta F_{AB} = F_A - F_B = -k_B T \ln\left(\frac{p_A}{p_B}\right)$$

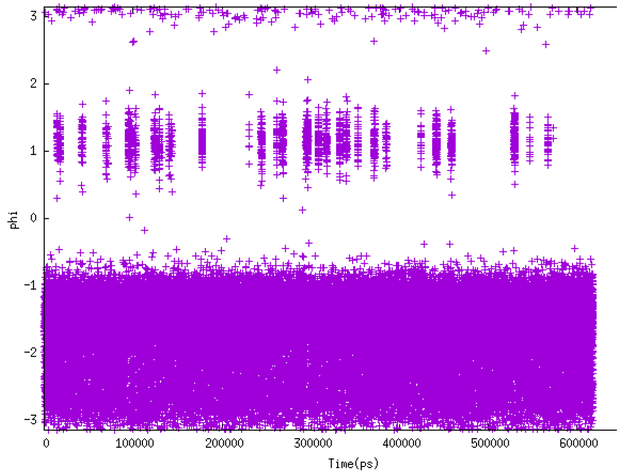
...

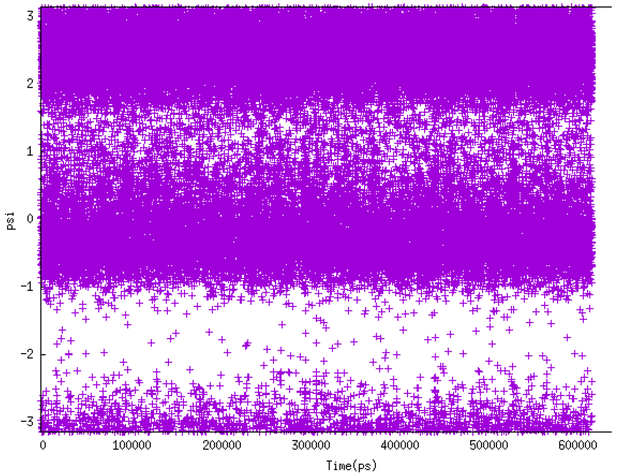
UNIVERSITÀ DEGLI STUDI DI MILANO

Another point of view:

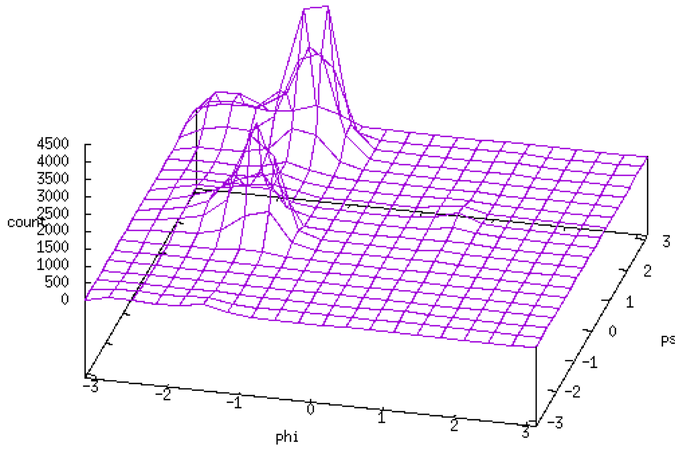
Going on looking and learning about your system you may decide to analyse it in terms of backbone dihedral angles



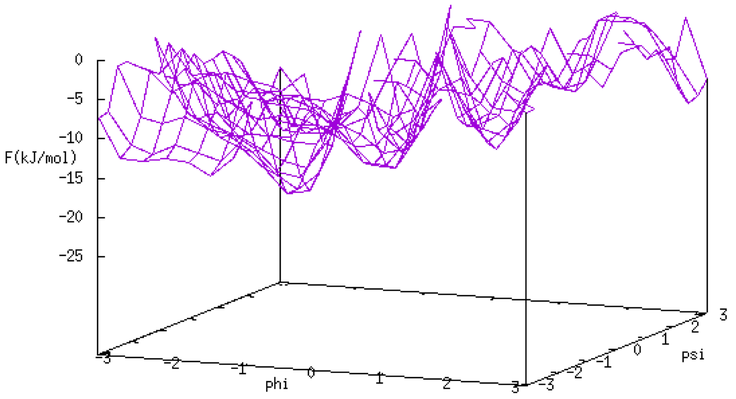




Histogram in 2D

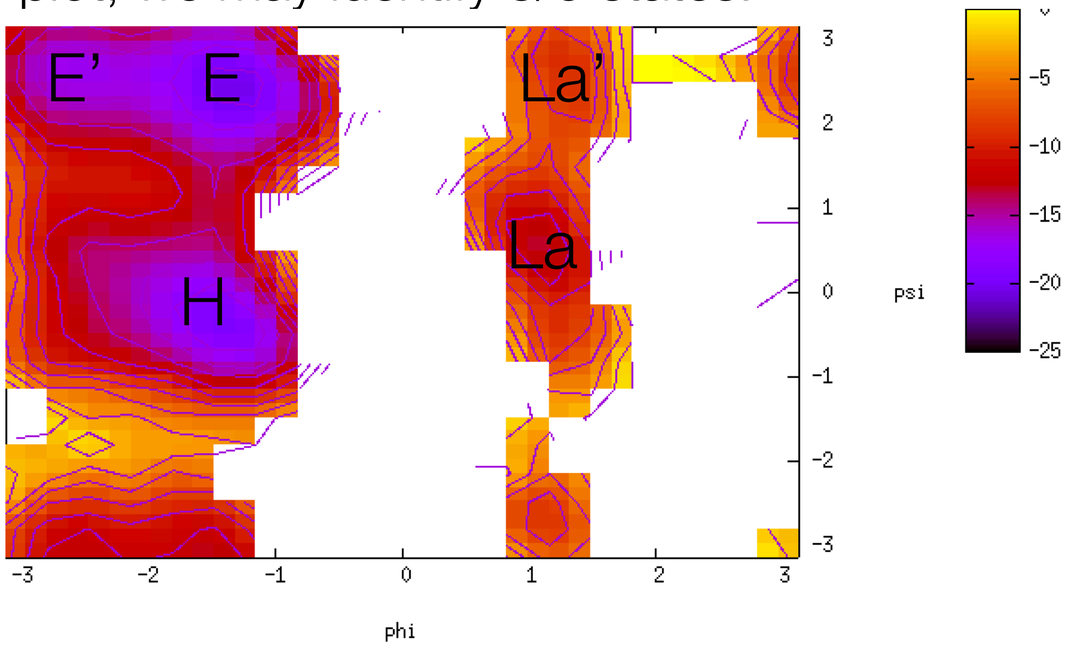


Free Energy Surface



Another point of view:

This is again the free energy as a contour plot, we may identify 3/5 states.



Dimensionality reduction -> Clustering,
I am using different labels because they may or may not correspond to the formers.

A further analysis of the conformations assigned to the different states may allow to rationalise the conformational space of alanine dipeptide in the way that we call Ramachandran Plot and its connection with the different secondary structures

You may already guess that the more complex the system you work with the more difficult/arbitrary become this analysis.

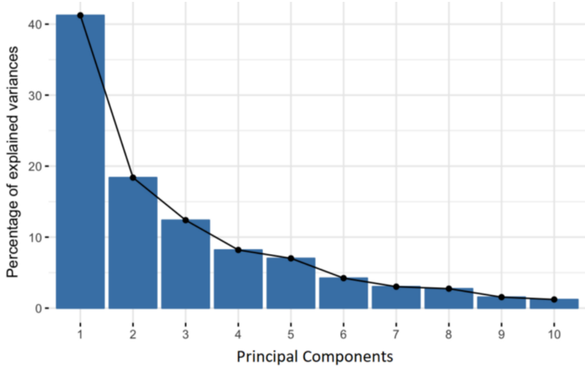
There are also machine learning techniques that try to find 'optimal' dimensionality reductions, the most used is Principal Component Analysis.

Dimensionality reduction techniques

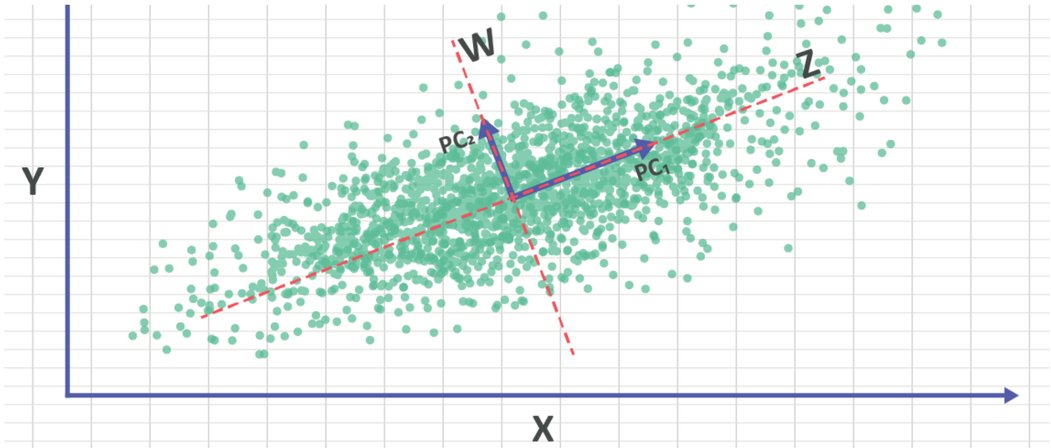
In addition to knowledge base dimensionality reduction, there are techinques that performs it by targeting specific properties. The most common is Principal Component Analysis (PCA), others include:

- Time-independent component analysis (TICA)
- Linear Discriminant Analysis (LDA)
- T-distributed stochastic neighbour embedding (t-SNE)
- Isomap
- Uniform manifold approximation and projection (UMAP)
- Autoencoders

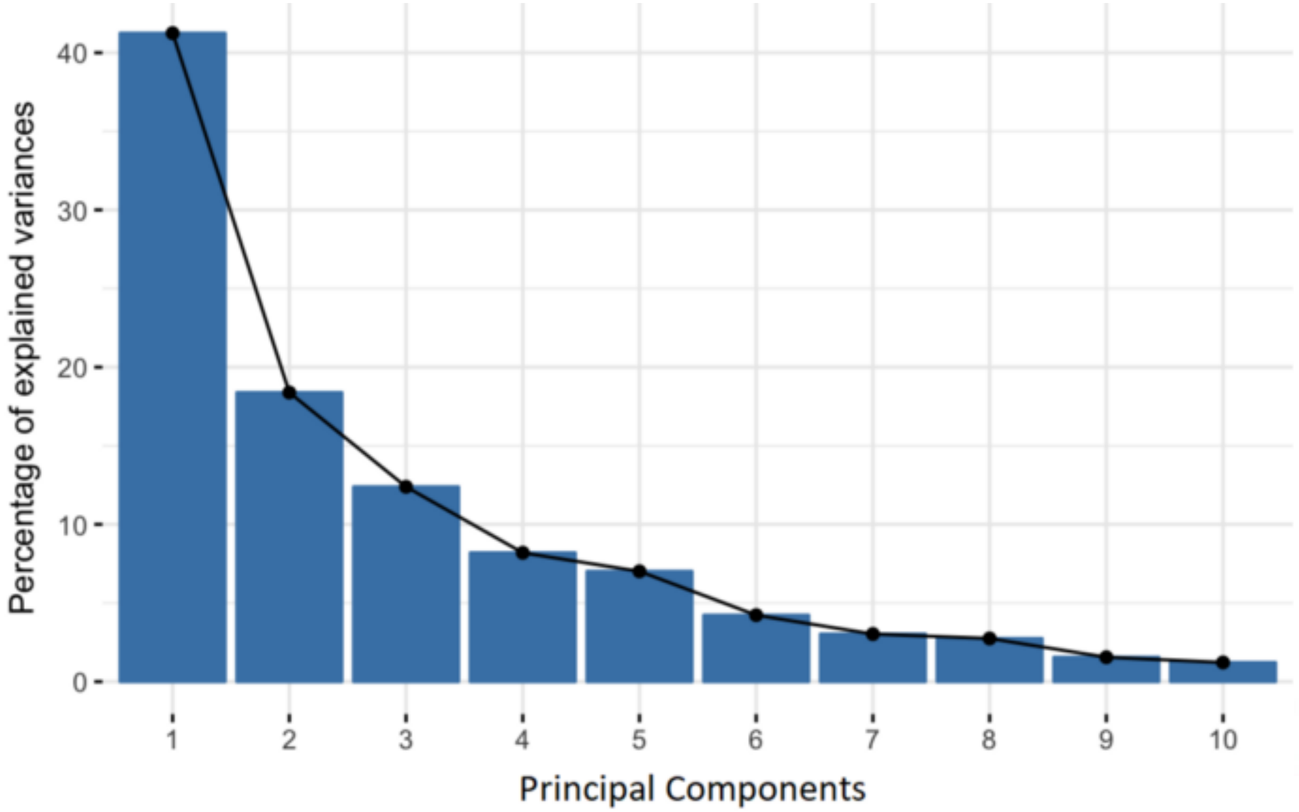
PCA search for a decomposition that captures the variance of the data, in the case of MD of the variance of atomic positions after removing translation and rotations.

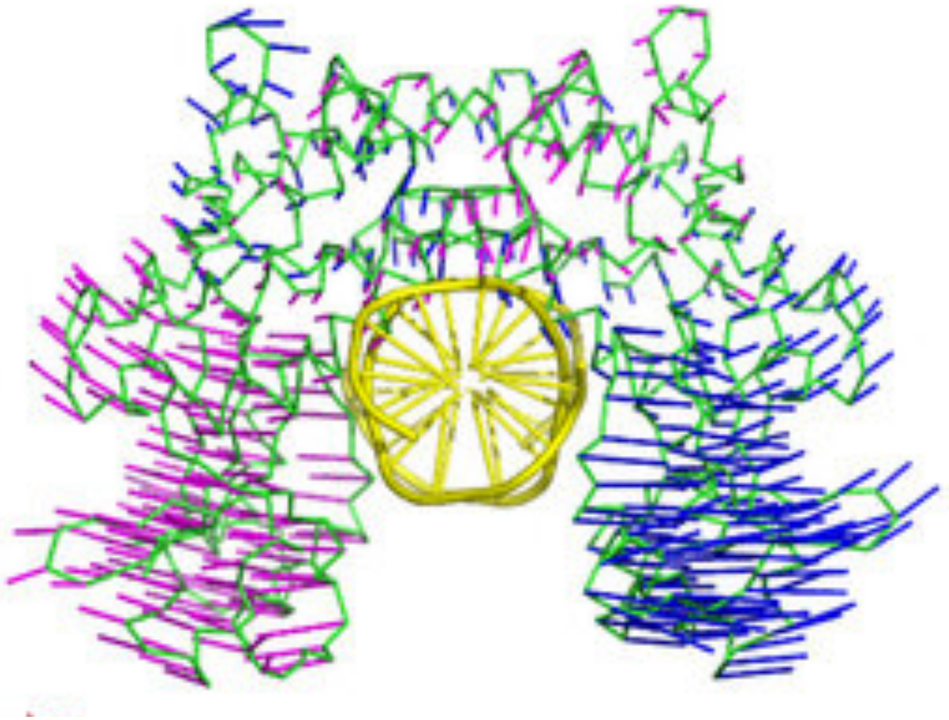


After selecting a given number of compoents the trajectory can be reprojected



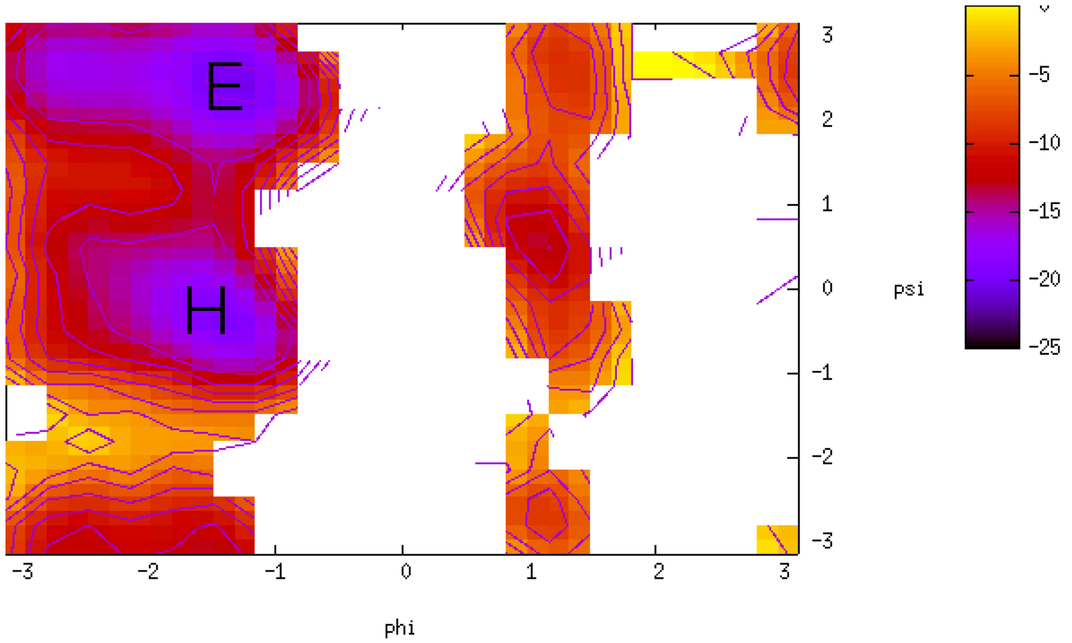
Principal component analysis



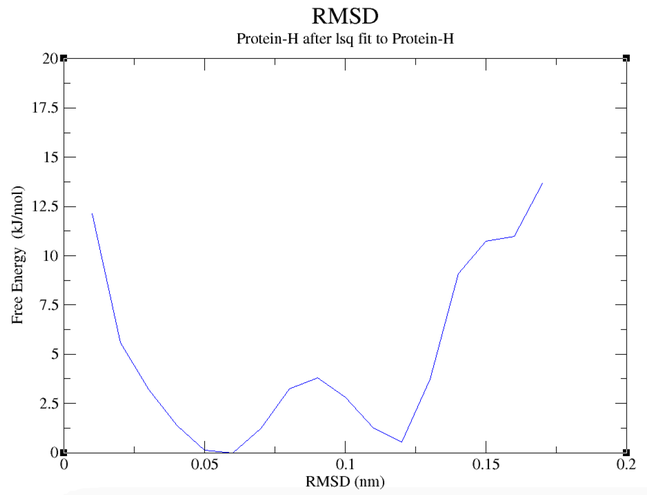


Equilibrium != Kinetics

By looking at histograms of simulation observables we are analysing equilibrium properties of the simulation, indeed we are removing the time-resolved information. In principle we have seen that kinetic rates are associated to free energy barriers, but while it is possible to go from rates -> barriers the opposite is not trivial:



On the left the barrier between the two main states H and E is ~8 kJ/mol (between 3-4 isolines), in the RMSD Free Energy the barrier between A and B is ~4 kJ/mol



UNIVERSITÀ DEGLI STUDI DI MILANO

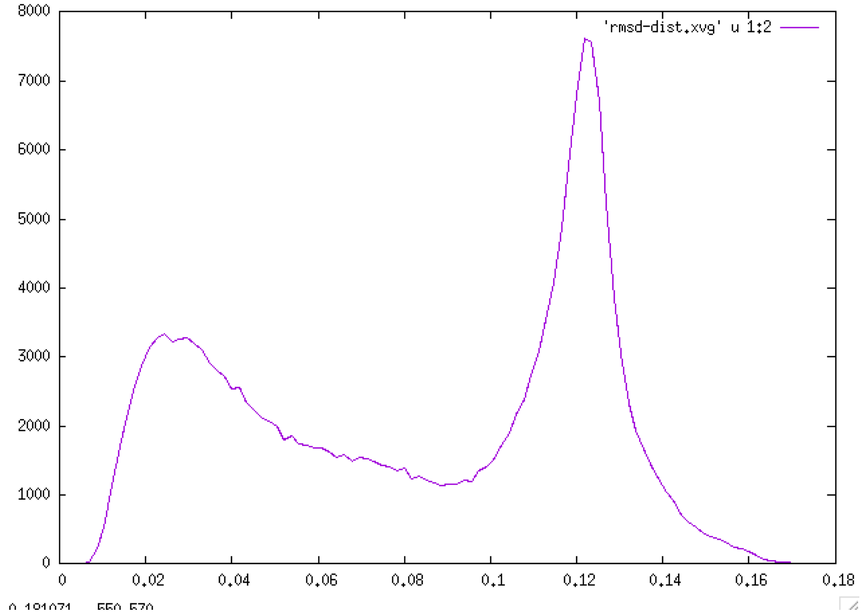
DBS

A direct clustering approach

There are multiple clustering algorithm in literature, clustering is a subfield of the statistical (or machine) learning field.

https://en.wikipedia.org/wiki/Cluster_analysis

For example a common approach in MD is that of calculating the RMSD of all pairs of conformations in the simulation, for example if we have 4000 conformation this would result in a 4000x4000 distance matrix. Then one can make an histogram of all RMSD value



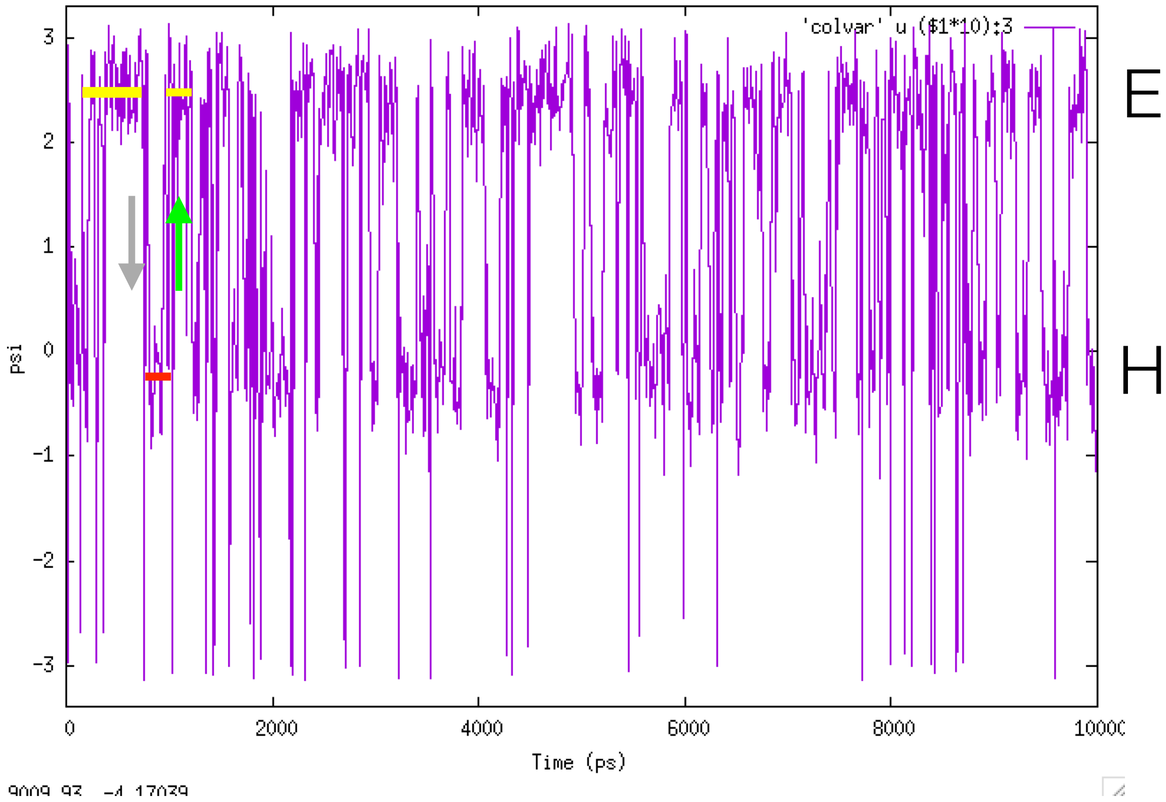
This suggest that the conformations are either at 0.02 nm from each other or at 0.12 nm. Then we can consider two conformations to belong to the same cluster if they are closer than 0.06 nm.

As a result we get 2 clusters: one including 97% of the population and one if the reminder 3%, this is again a different result from the previous clusterings. Again we should investigate whether this clustering is useful to say something on the system or not.

Kinetics > Equilibrium

To perform a kinetic analysis of the simulation we still need to perform a dimensionality reduction + clustering step, that is we need to look at the time-resolved trace of some observable connecting two or more states.

From the previous work we should work in the phi/psi space. We will learn more on this in the “Markov State models” lecture. Here to make it simple we focus on the psi collective variable.



E

H

We can zoom in the time resolved plot, from this we can learn a lot:

- **Residence times**: this will be the average time spent in a state
- **Transition rates**: the number of transitions per time unit
- **Equilibrium properties**: remember that from kinetics you can get free energies

Errors estimates: block averaging and replicates

The standard approach to calculate errors is to repeat an observation multiple time to get its average and standard error of the mean. Molecular Dynamics simulations allow to add an additional way to calculate statistical errors that is block averaging:

In practice we **estimate** averages using sums: $< f > \doteq \frac{1}{N}\sum_{i=1}^{N} f(x_i)$

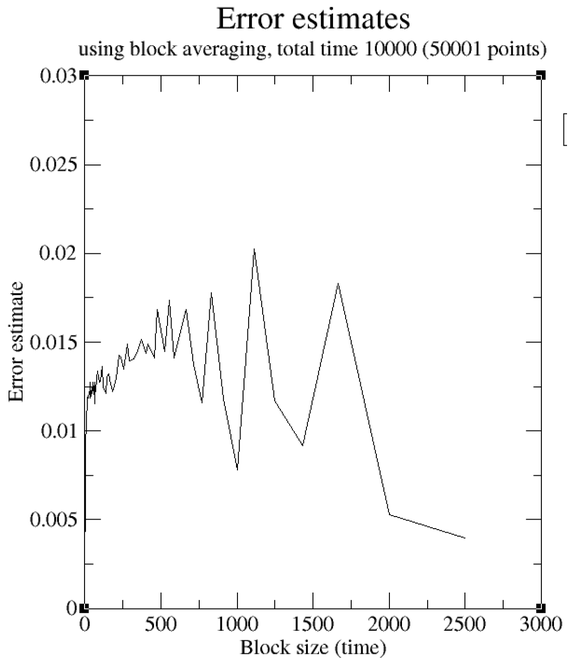
Variance and Standard deviation:

$$\sigma^2 = var(f) = < (f - < f >)^2 > = \int dx (f(x) - < f >)\rho(x) \doteq \frac{1}{N-1}\sum_{i=1}^{N}(f(x_i) - < f >)^2$$

$$\text{Std-err} = \sqrt{\frac{var(f)}{N}} = \frac{\sigma}{\sqrt{N}} = \sqrt{\frac{1}{N^2 - N}\sum_{i=1}^{N}(f(x_i) - < f >)^2}$$

In this second case you consider your simulation has multiple experiments, multiple blocks, and then you calculate averages and standard errors of the means over the block. But then you can try to divide your simulations in 3, 4, 5, … blocks

Let's say that your simulation is 10,000 ps long, then you can split it into 4 blocks by 2500 ps, 5 by 2000 ps etc, when you have only few blocks your error estimate is noise, but there is a regime range, here around 500-600 ps.

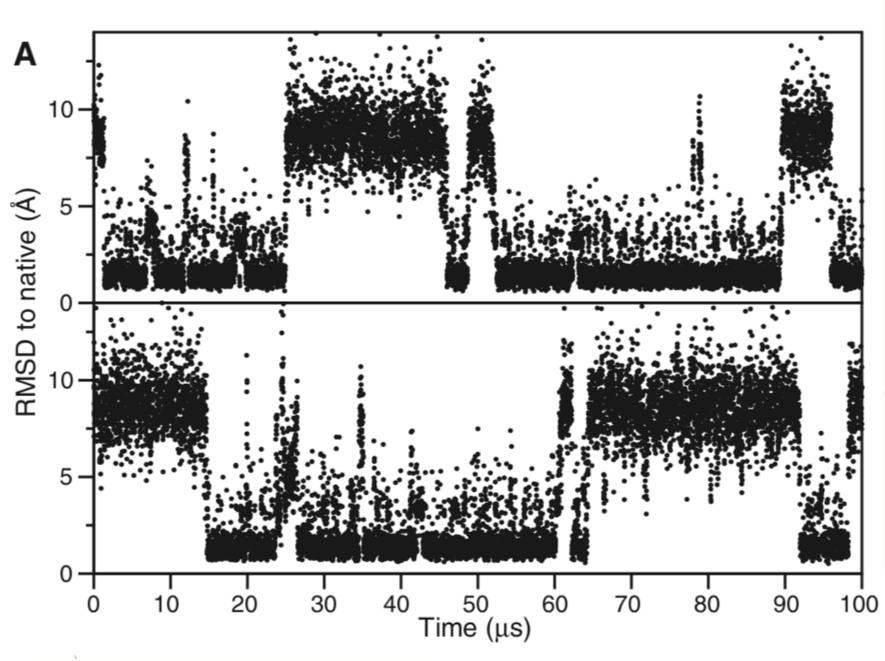


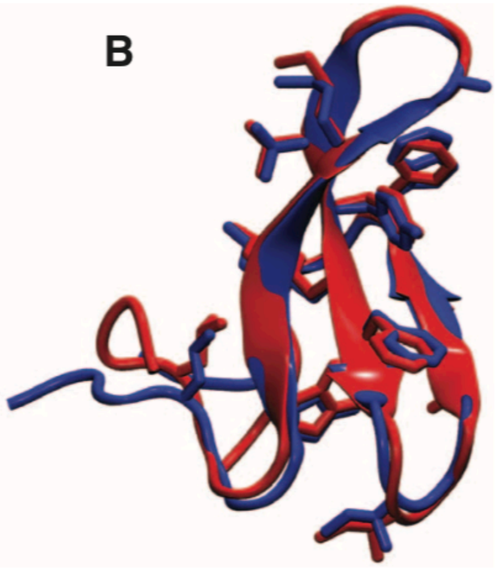
UNIVERSITÀ DEGLI STUDI DI MILANO

DBS

MD examples: Protein Folding

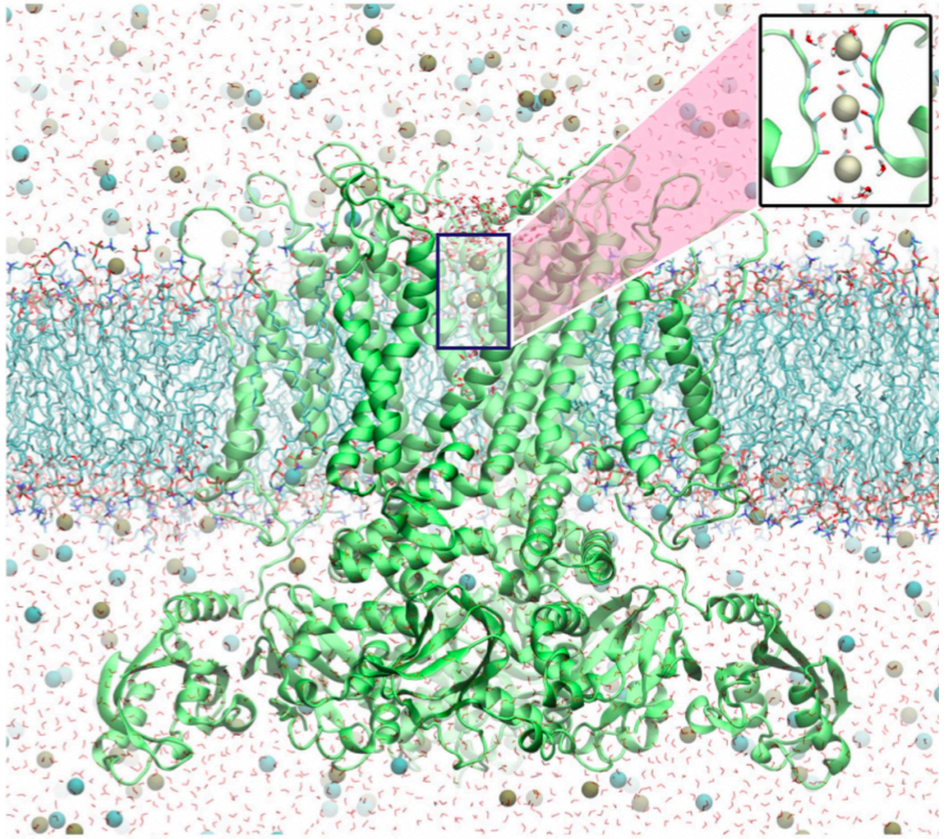
Here we want to study the reversible folding/unfolding of small protein. We take two different unfolded conformations (high RMSD from native) and we let them fold. These are very long MD and you can see multiple events. The presence of multiple events allow to calculate properties by block average and then compare them with the second simulation.

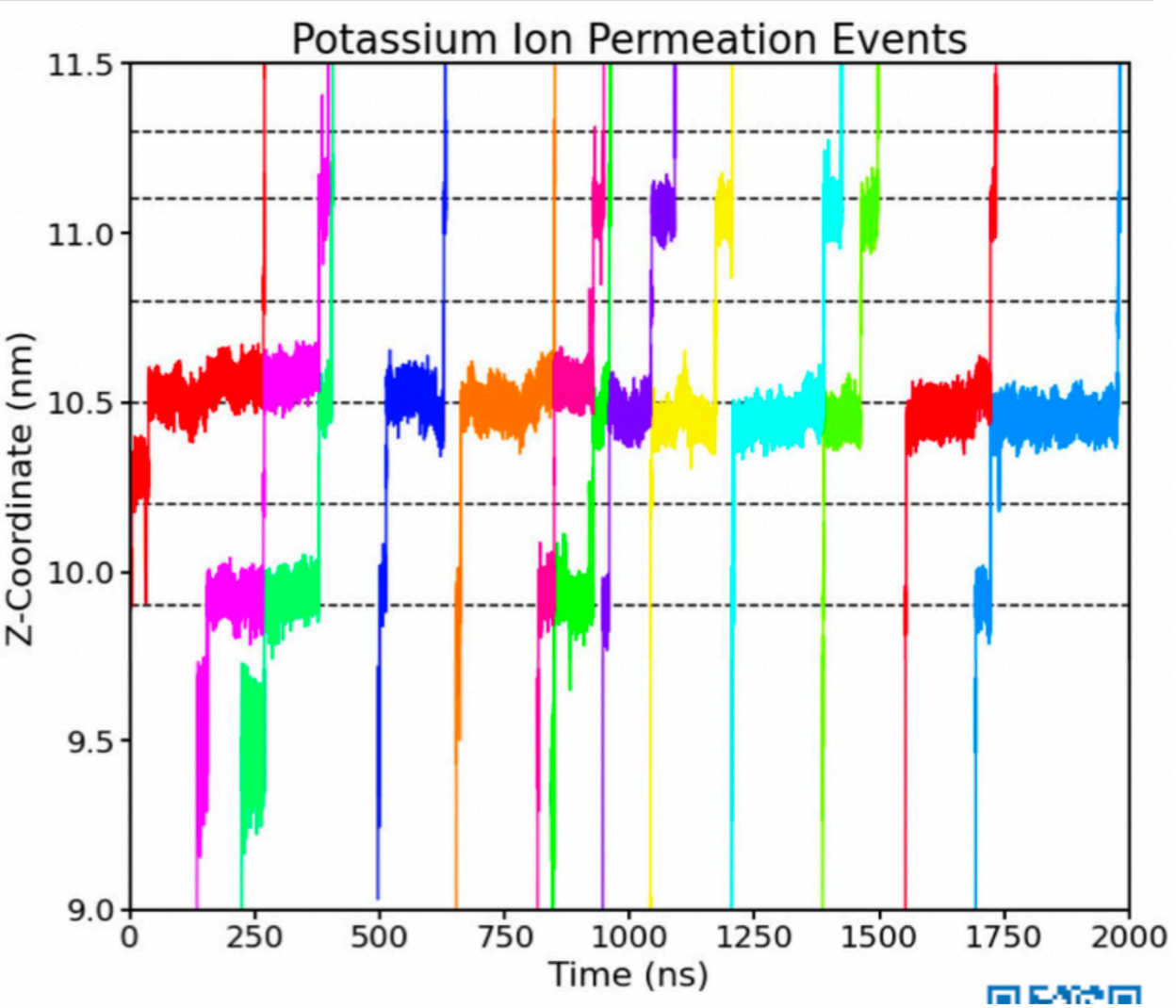


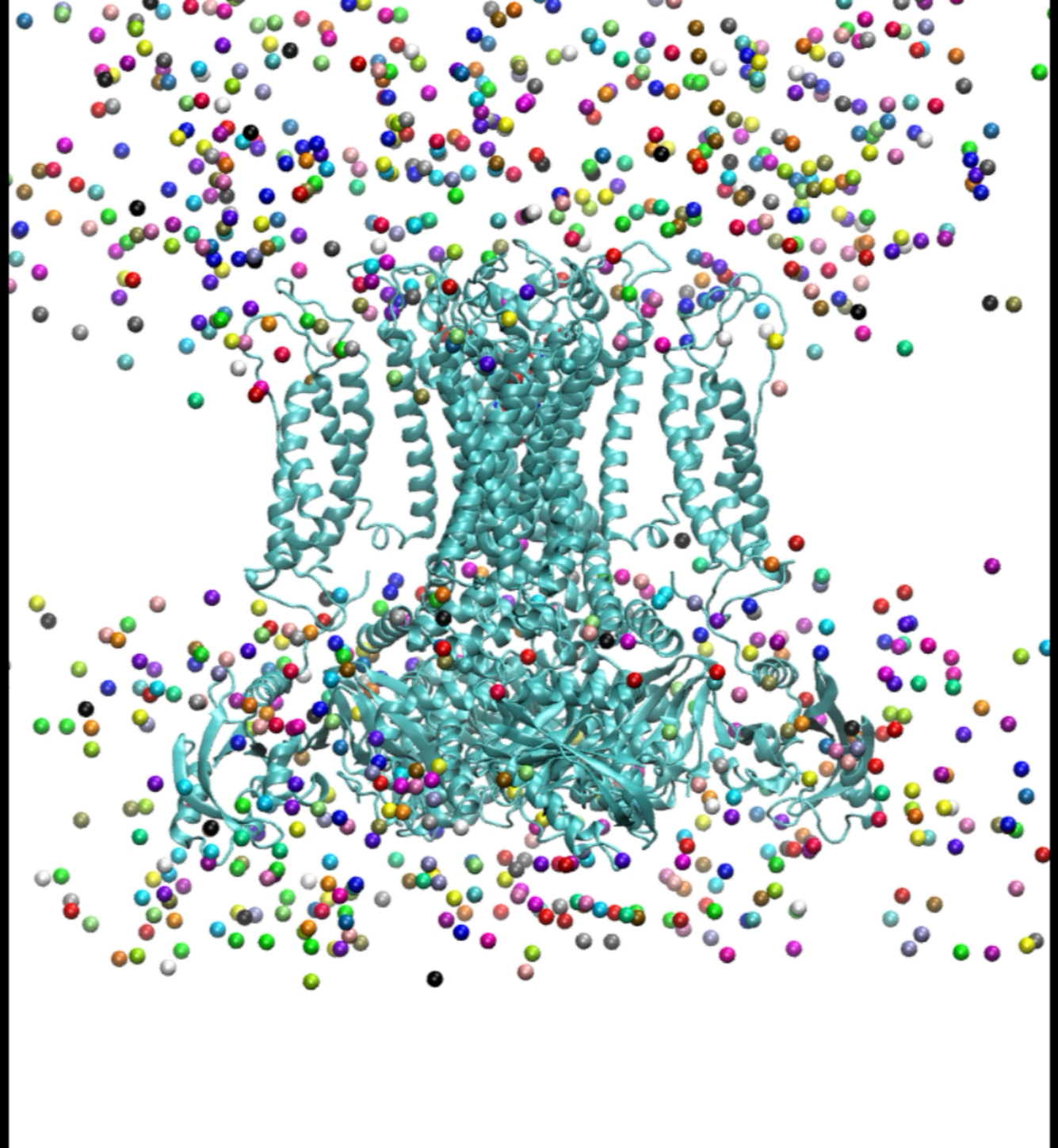


MD examples: Protein Function

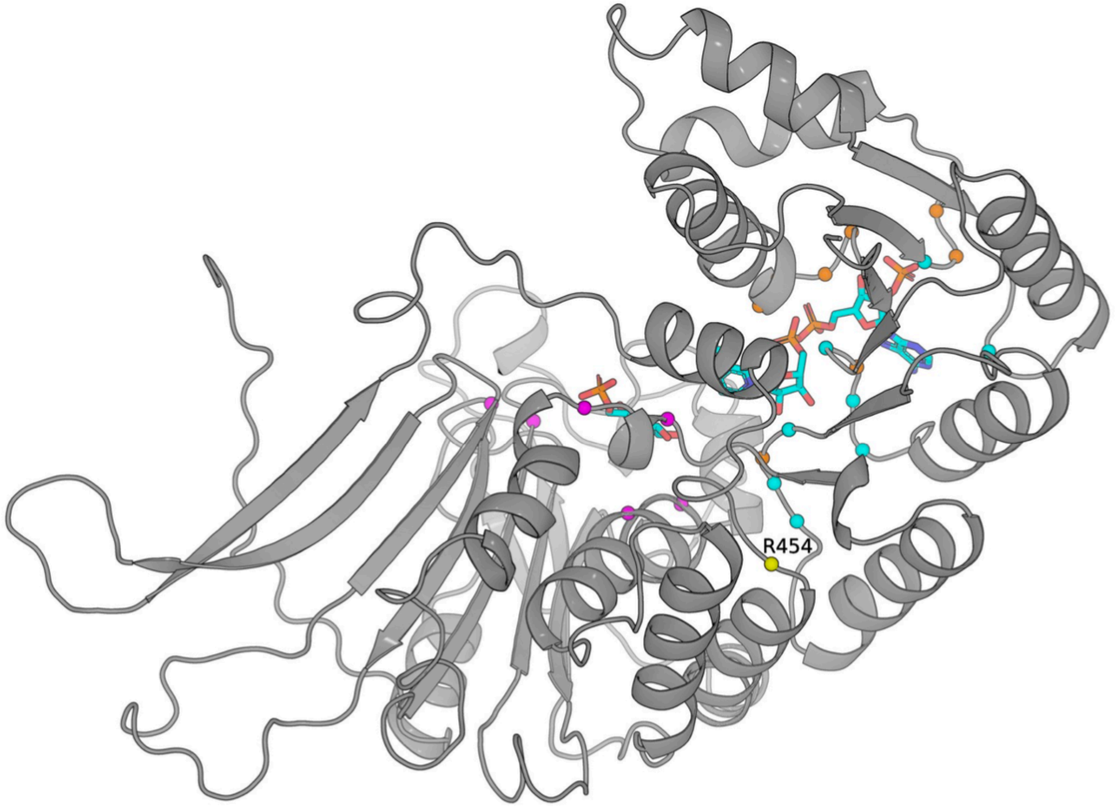
Here we observe the mechanisms by which K+ ions go through this channel

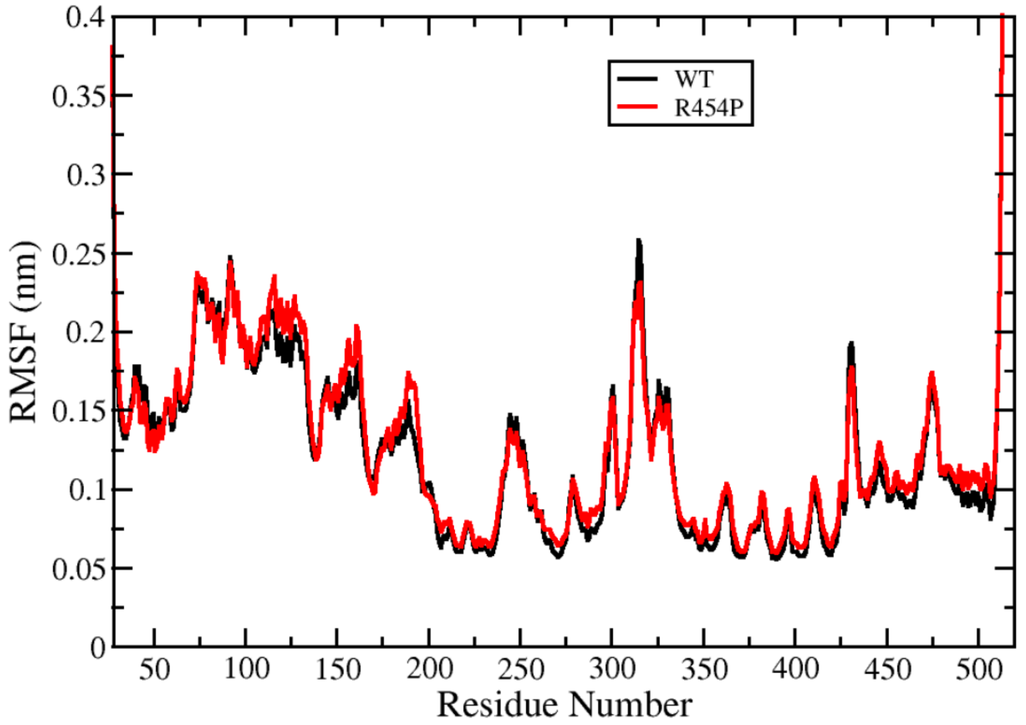






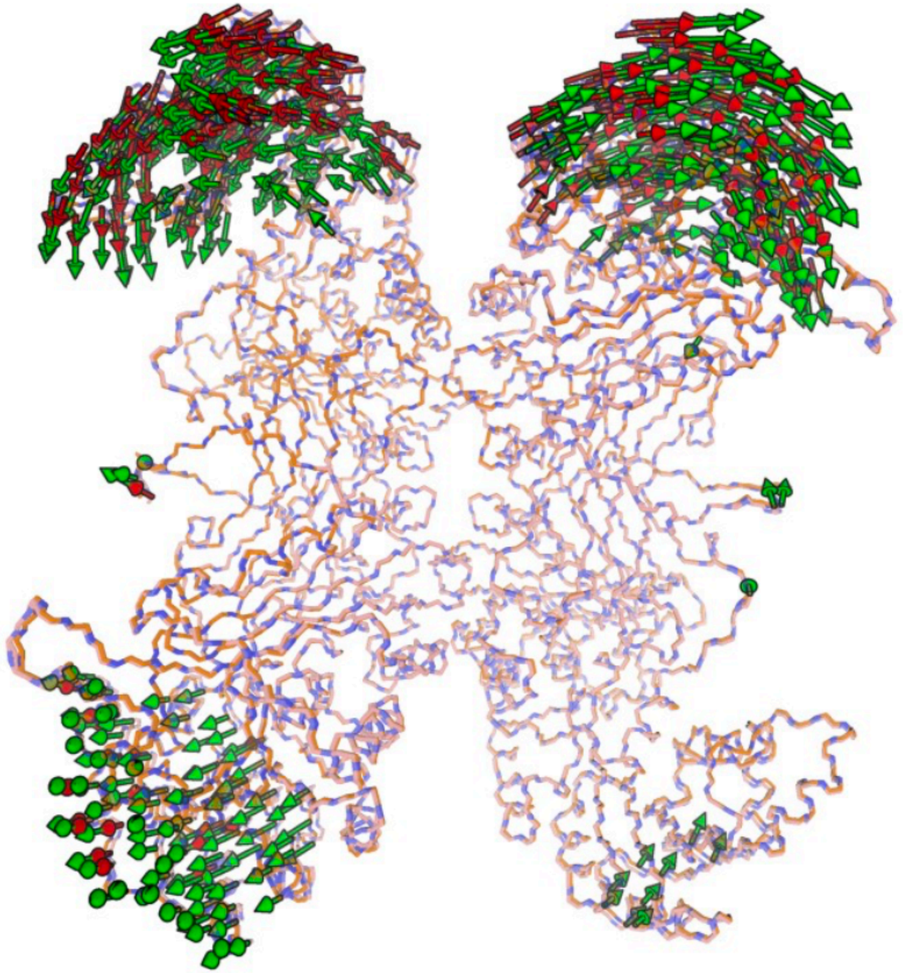
MD examples: Conformational Dynamics



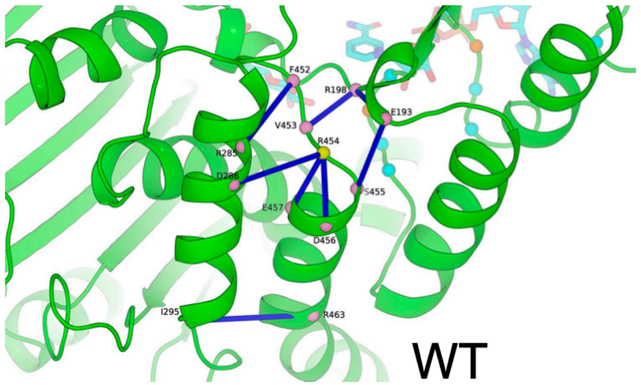


Often MD is used to investigate the different flexibility of different protein regions: here for example we show that a mutation does not seems to change the local flexibility of this protein

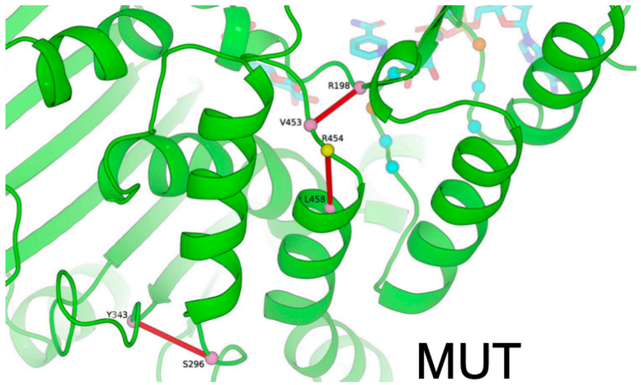
MD examples: Conformational Dynamics



Nonetheless, a more global view, for example provided by principal component analysis, show the presence of a global motion, that is stronger in the case of the mutation.



WT



MUT

And then this can be associated for example to a difference in the hydrogen bond network around the site of the mutation

Questions:

- What is a force-field?
- How are covalent interactions described?
- How are non-covalent interactions described?
- Can you describe the Euler algorithm?
- Why does Euler is not a good MD algorithm?
- What is the typical time step of a simulation?
- What are PBC in MD?
- How do we measure distances in MD?
- What does determine the speed of a MD simulation?
- What is the relationshiip between kinetic energy and temperature?
- How do you estimate a free energy profile from MD?
- What are the main steps to setup an MD simulation?