

Free energy perturbation methods

Christian Blau - BioExcel

blau@kth.se

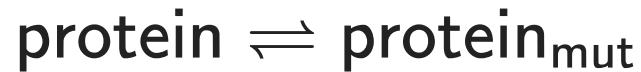
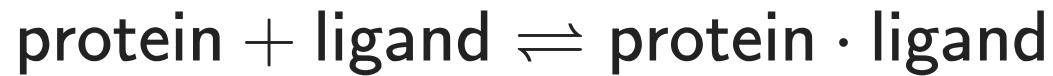


Content

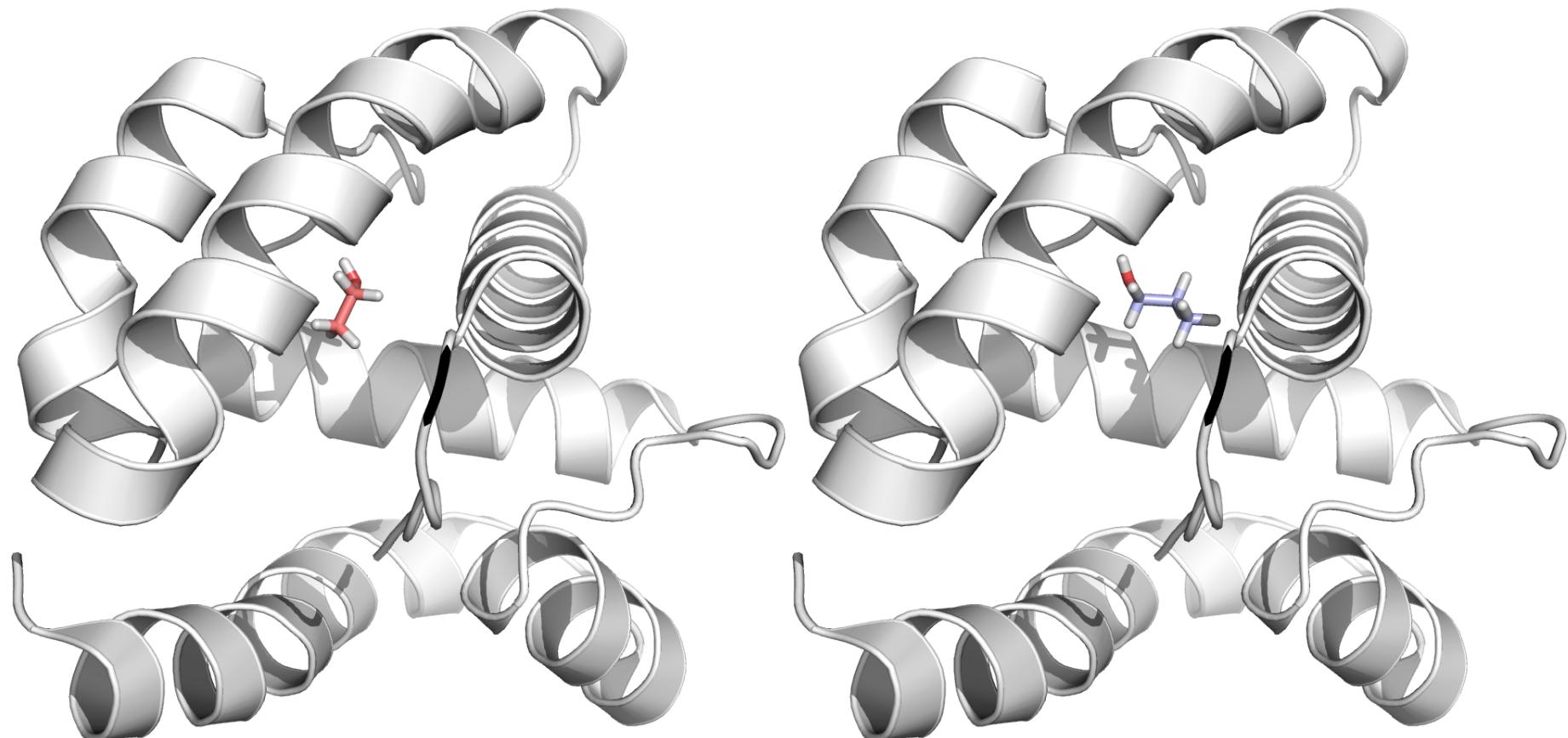
- Setting the scope
- Crash-course / refresh statistical physics
- 10 free energy perturbation methods and flavours
- Simulation setup considerations

The challenge

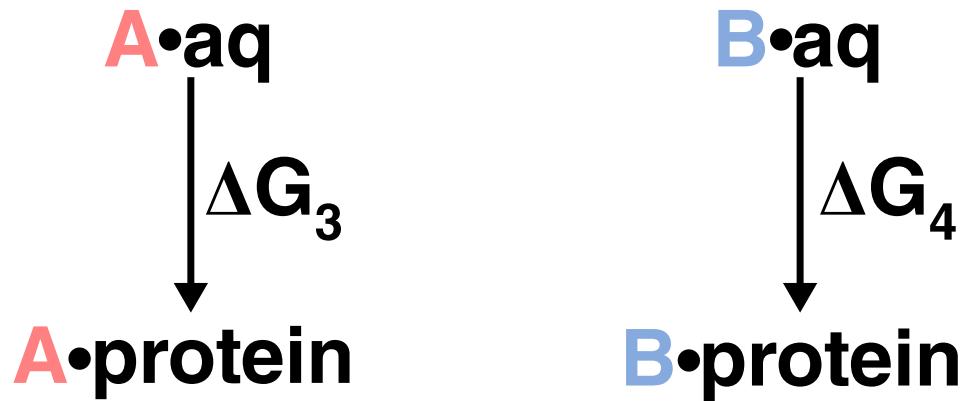
Protein mutations / ligands



Concrete system

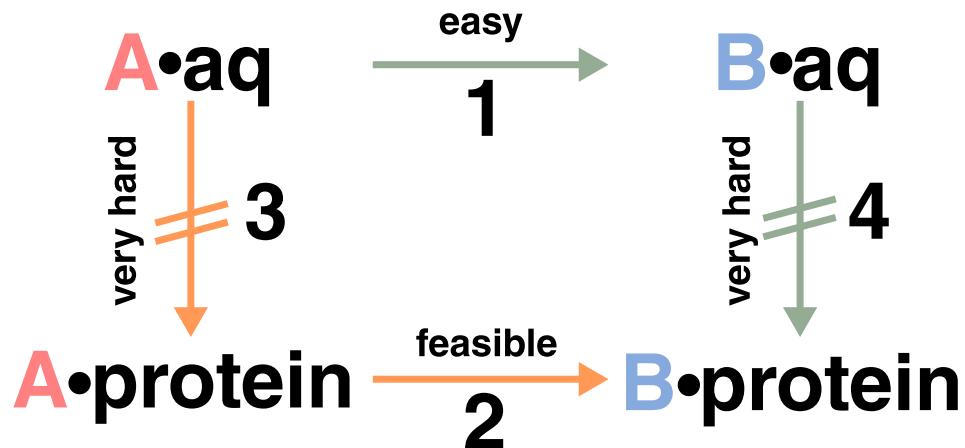


Relative binding free energies



- ABFE hard, but possible

Relative binding free energies



Basic framework

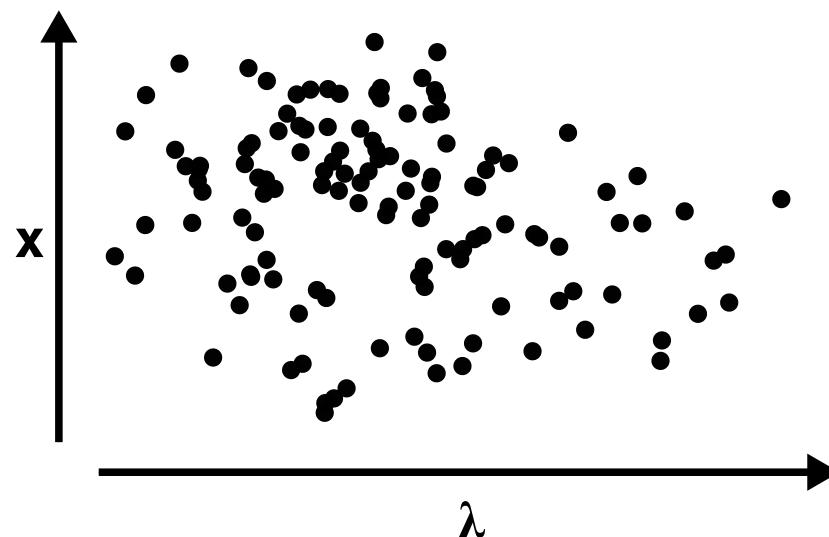
Boltzmann free energies

$$G = -k_B T \log P$$

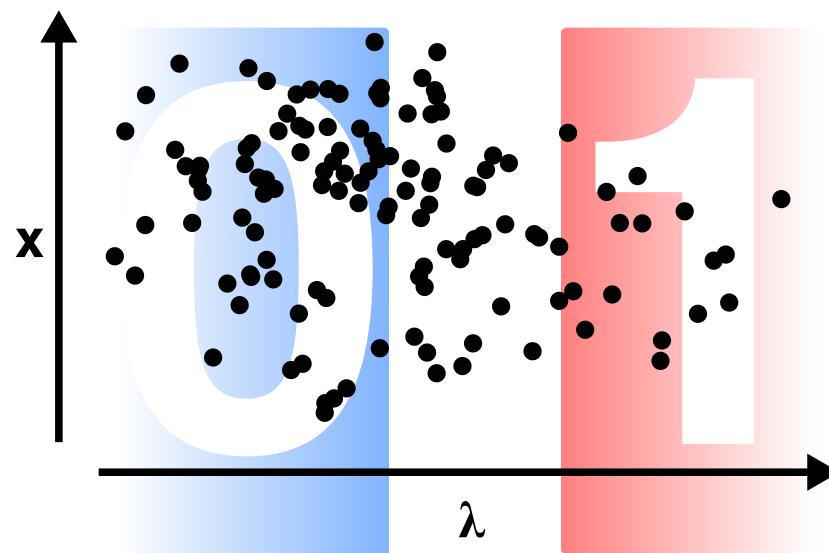
Calculating free energies is counting

$$\Delta G_{12} = G_1 - G_2 = -k_B T \log \frac{P_1}{P_2}$$

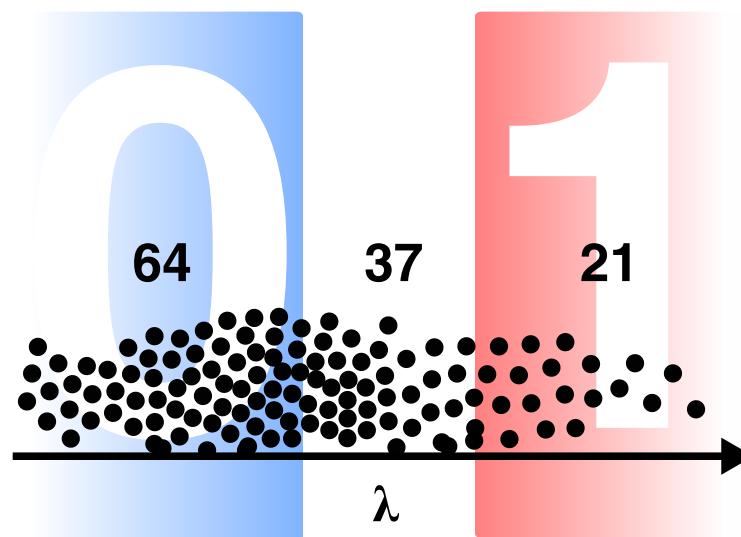
Calculating free energies is counting



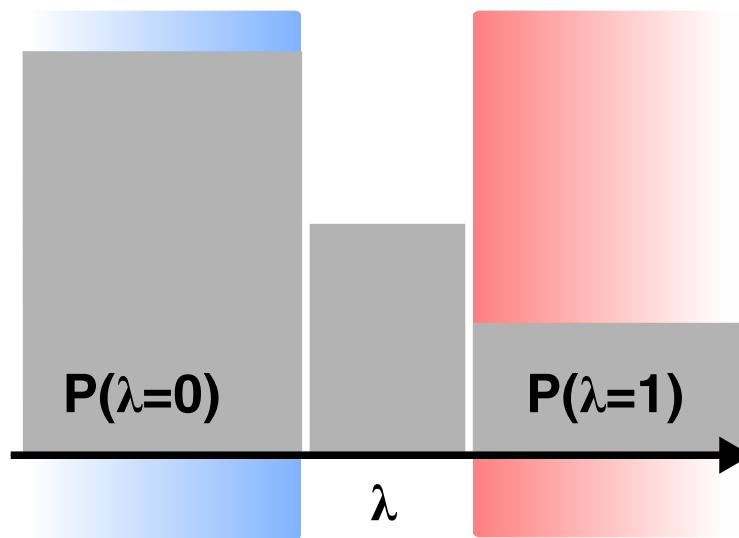
Calculating free energies is counting



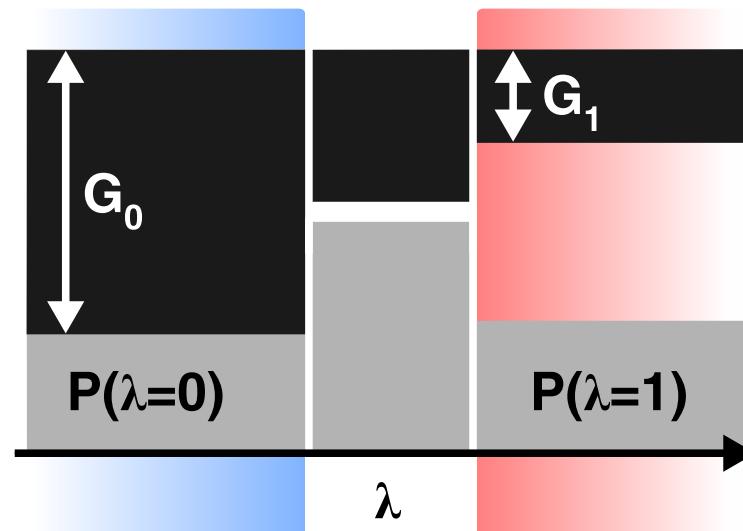
Calculating free energies is counting



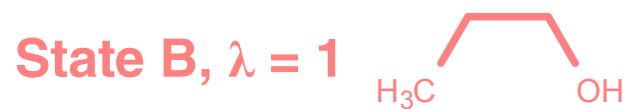
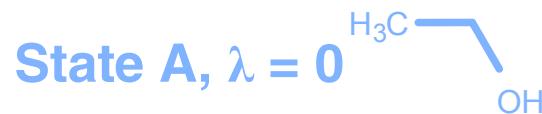
Calculating free energies is counting



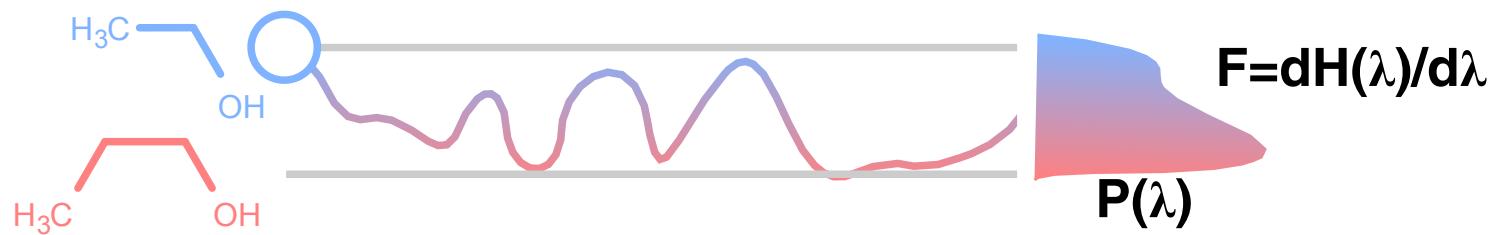
Calculating free energies is counting



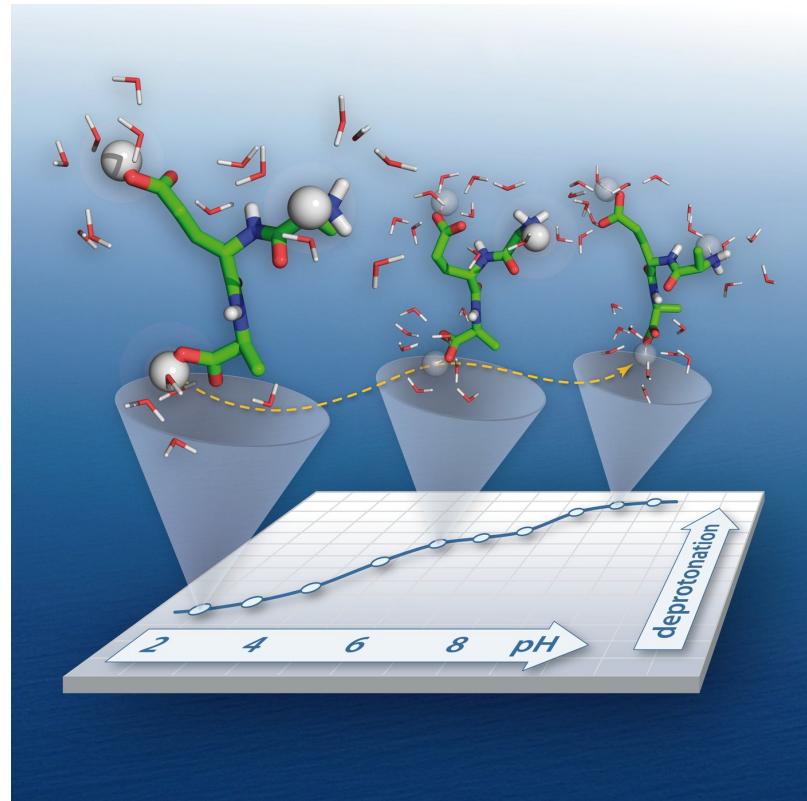
How to “count” chemistry



Lambda dynamics

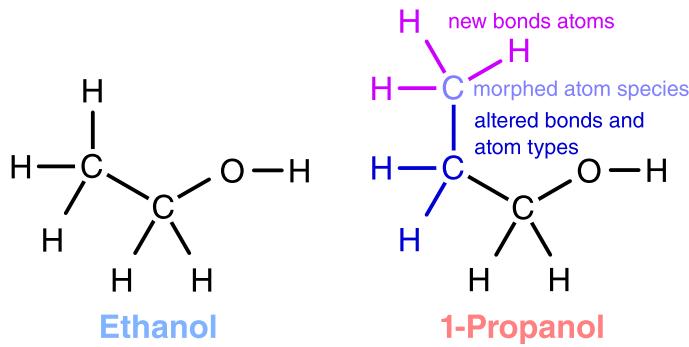


Dynamic protonation / constant pH simulation

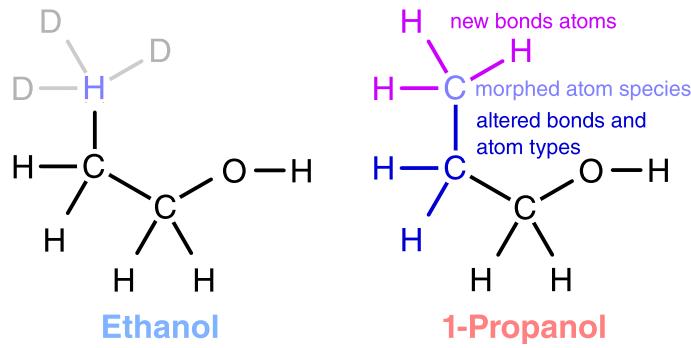


© Max Planck Institute for Biophysical Chemistry / Plamen Dobrev / Carsten Kutzner

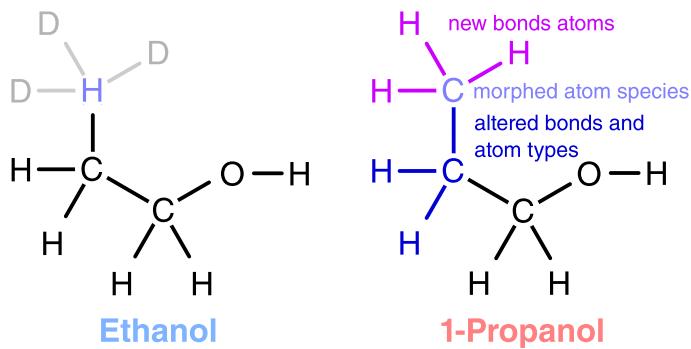
Moving between lambas - morphing states



Moving between lambas - morphing states

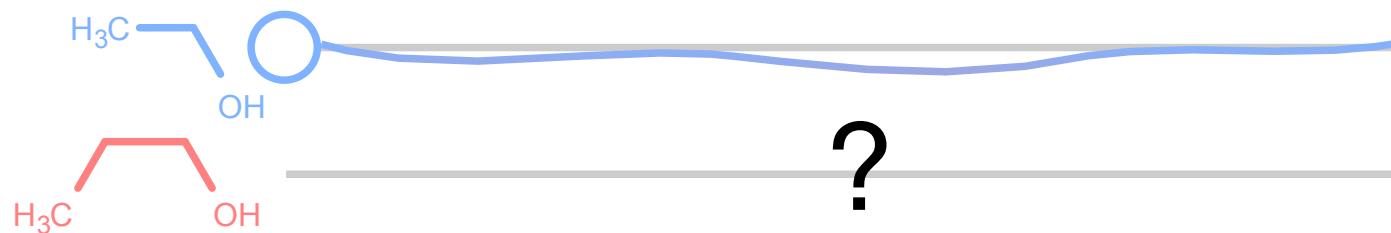


Moving between lamdas - morphing states



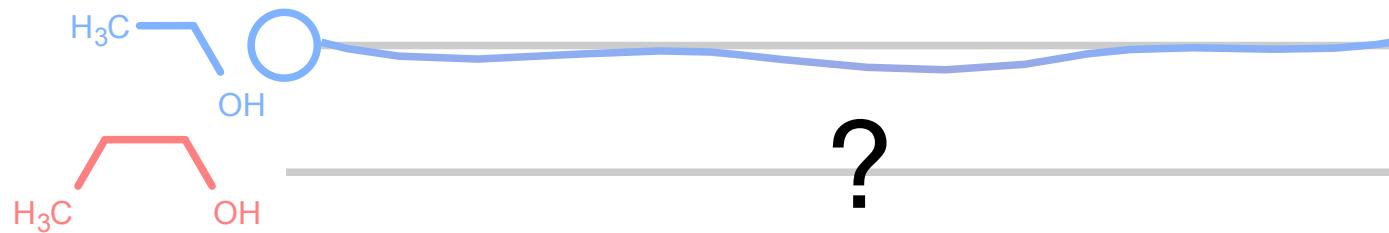
- a tedious job, use pmx in the GROMACS universe

The issue with lambda dynamics



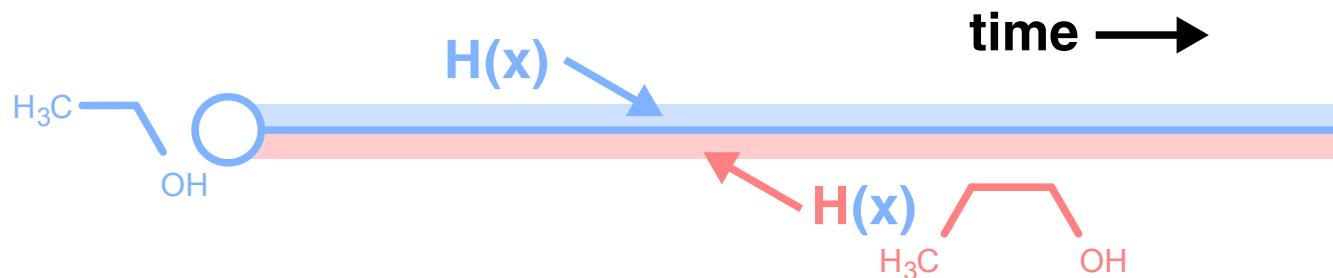
Energy/Hamiltonian based methods

Main concept



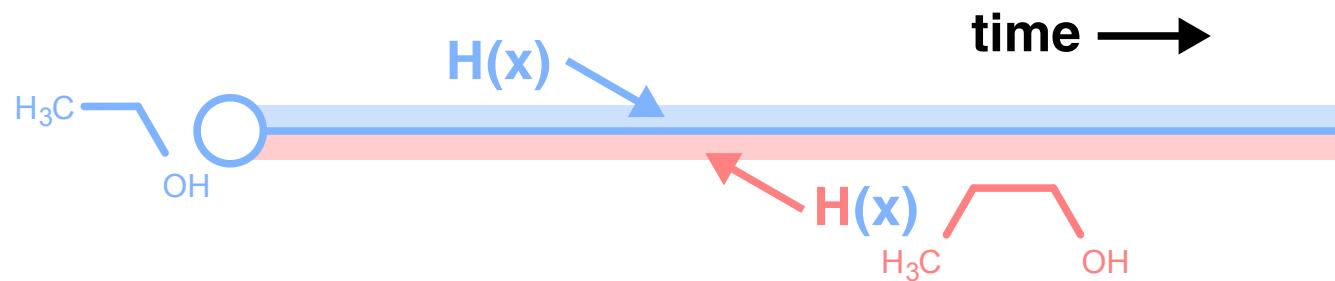
- free energy calculation methods = enhancing sampling

Simple FEP



Zwanzig, R. W. J. Chem. Phys. 1954, 22, 1420-1426

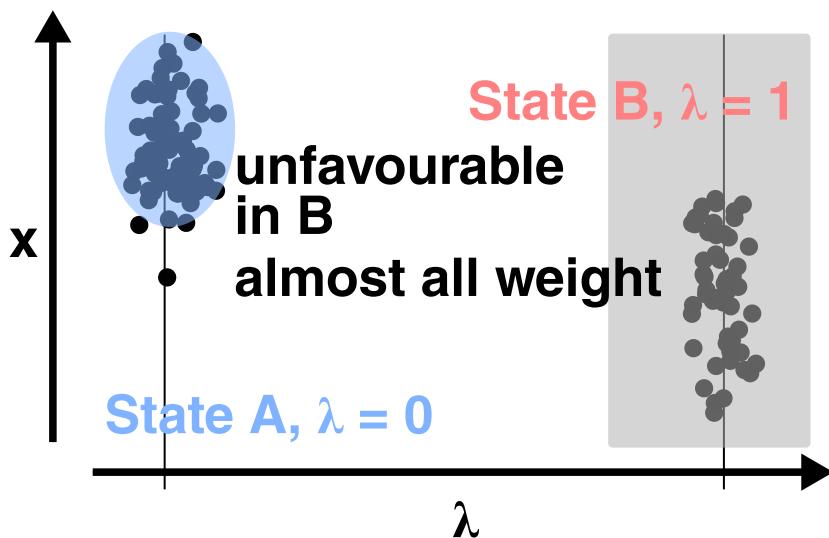
Simple FEP



$$\Delta G_{AB} = -\frac{1}{\beta} \log \langle \exp(-\beta(H_B(\vec{x}) - H_A(\vec{x}))) \rangle_A$$

Zwanzig, R. W. J. Chem. Phys. 1954, 22, 1420-1426

Simple FEP - main issue



$$\Delta G_{AB} = -\frac{1}{\beta} \log \langle \exp(-\beta(H_B(\vec{x}) - H_A(\vec{x}))) \rangle_A$$

Simple FEP

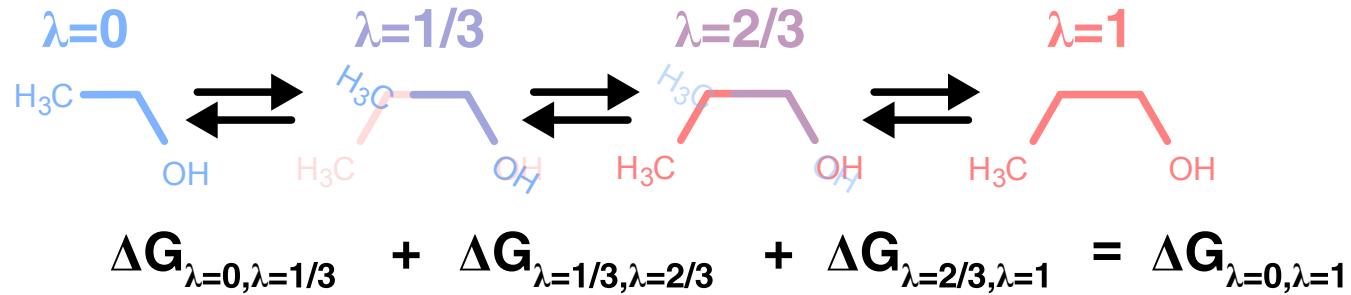
Pros

- technically the simplest form
- need to run just one simulation
- only needs energy calculation
- no need to generate a starting structure for the B state

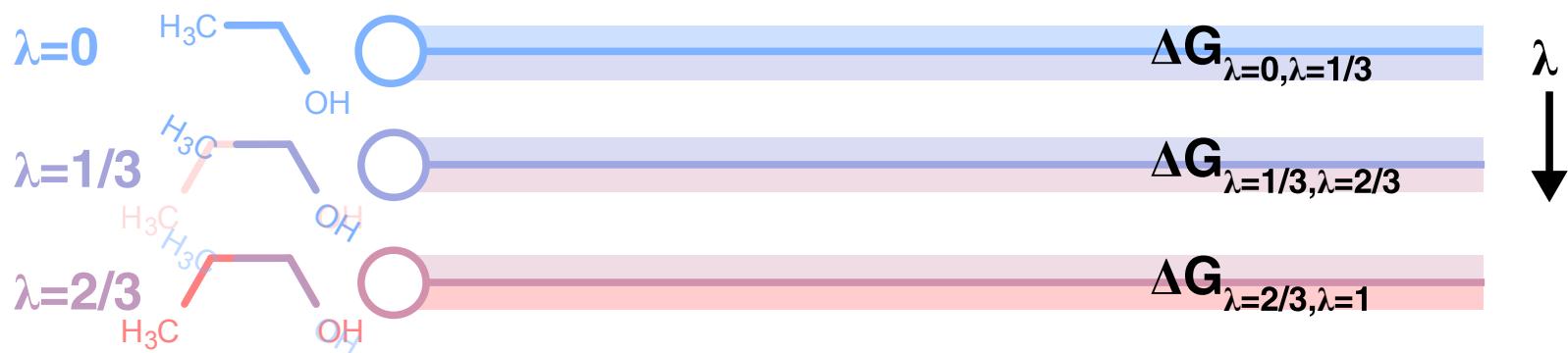
Cons

- very bad performance
- no sampling in B state will mean that reasonable configurations will be missed

Stratified FEP



Stratified FEP



Stratified FEP

- **Pros**
 - technically still easy to set up
 - only needs energy calculation
- **Cons**
 - still very bad performance
 - need to generate a starting structure for the intermediate states

Simple BAR



- Bennet's Acceptance Ratio

Simple BAR



$$\Delta G_{AB} = \frac{1}{\beta} \log \frac{\sum_{x \in A} f [H_A(\vec{x}) - H_B(\vec{x}) + C] / n_A}{\sum_{x \in B} f [H_B(\vec{x}) - H_A(\vec{x}) - C] / n_B} + C$$

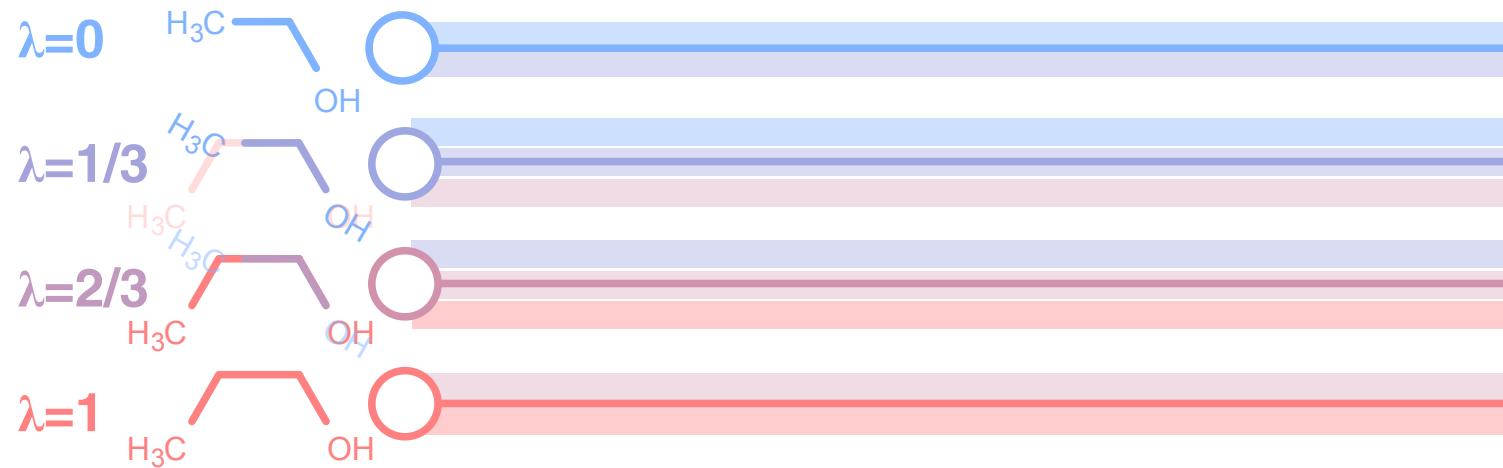
with

$$C = \Delta G_{AB} - \frac{1}{\beta} \log \frac{n_A}{n_B}; f[x] = \frac{1}{1 + \exp(\beta x)}$$

Simple BAR

- use gmx bar or similar tools
- **Pros**
 - better performance than simple FEP
- **Cons**
 - Still largely dependent on ensemble overlap

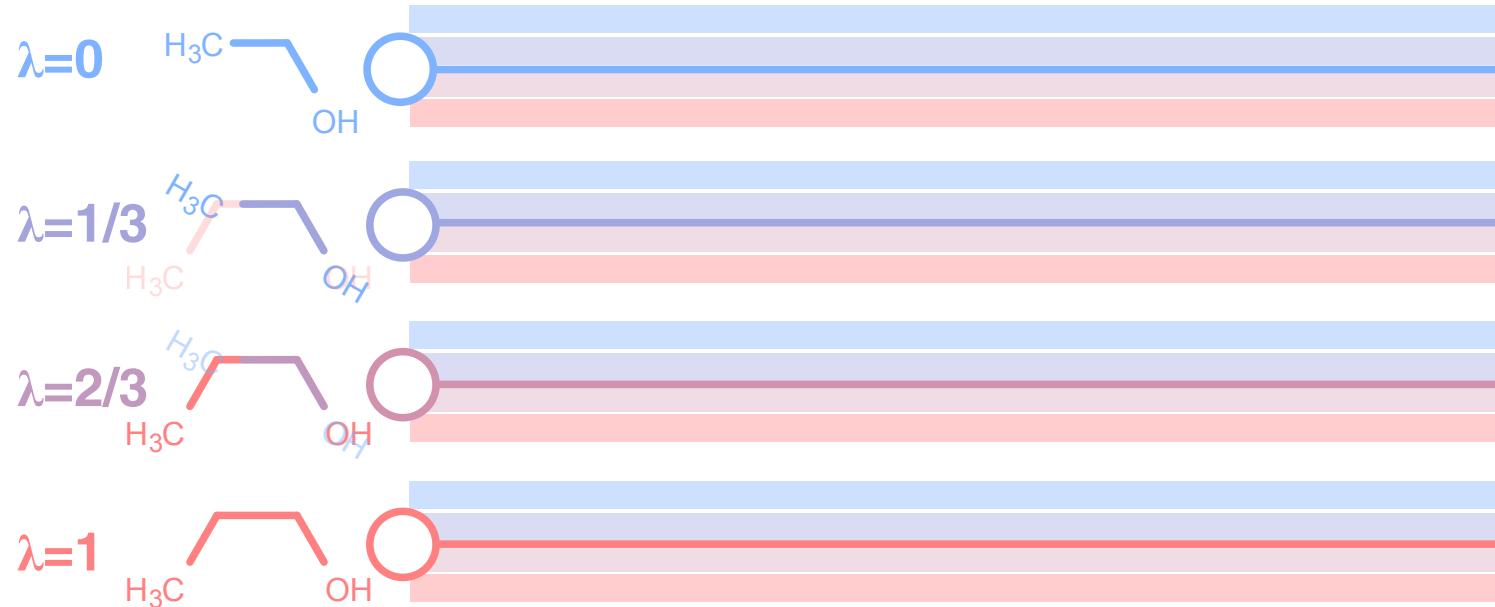
Stratified BAR



Stratified BAR

- Pros
 - combines the benifits of BAR and stratification
- Cons
 - can do even better with same simulation effort (see mBAR)

Multistate BAR

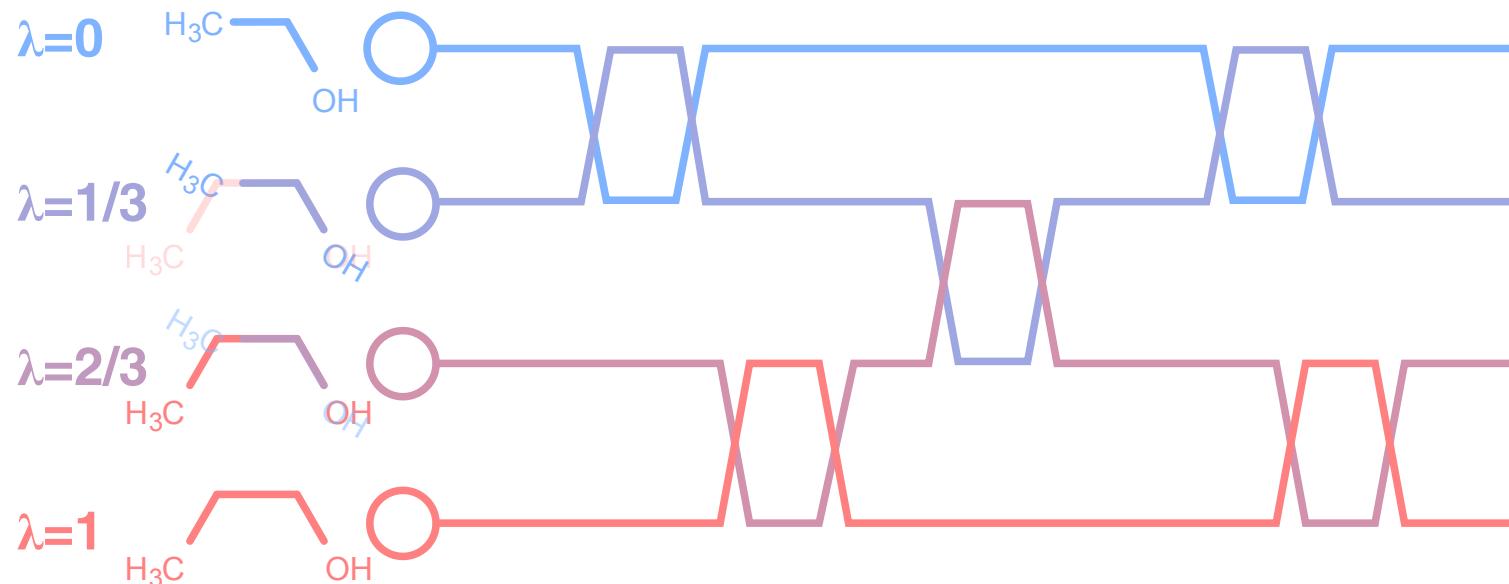


Multistate BAR

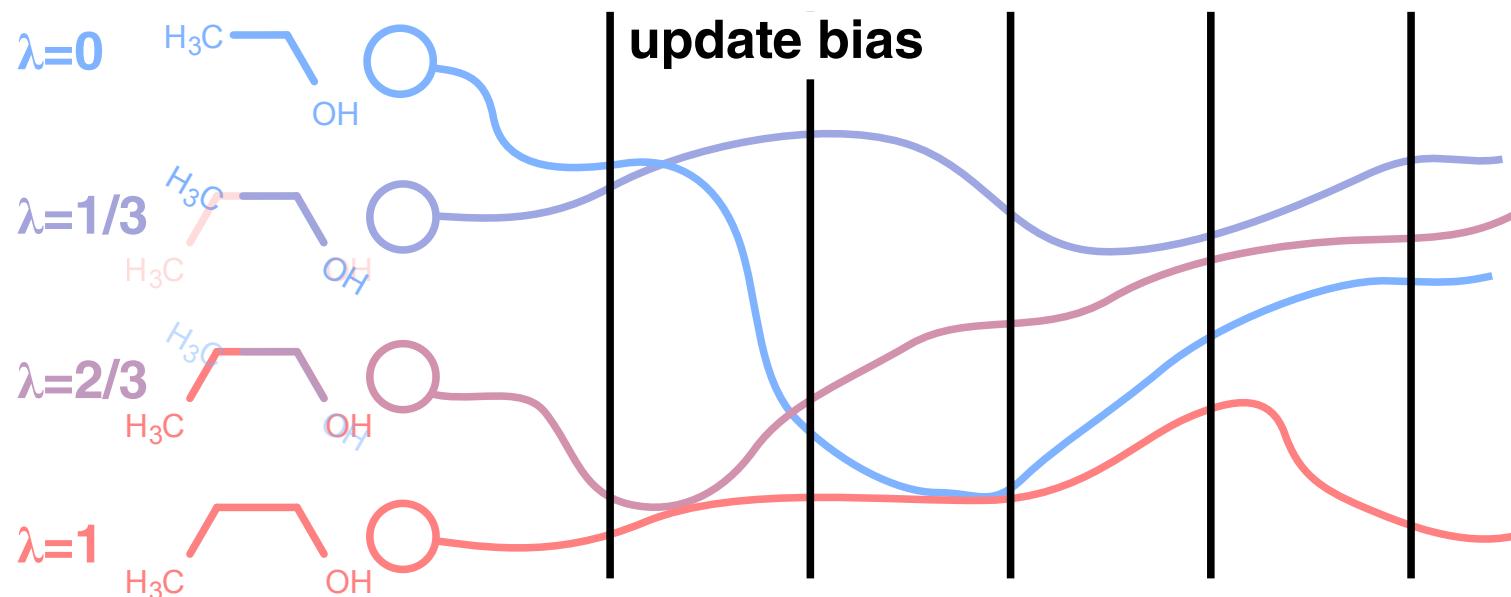
$$G_i = -\beta \log \sum_{j \in \text{simulations}} \sum_{n \in \text{configurations}_j} \frac{\exp(-\beta H_k(x_n^j))}{\sum_{k \in \text{simulations}} n_k \exp(G_k - \beta H_k(x_n^j))}$$

- use [alchemlyb](#) to solve that type of equation

Hamiltonian replica exchange

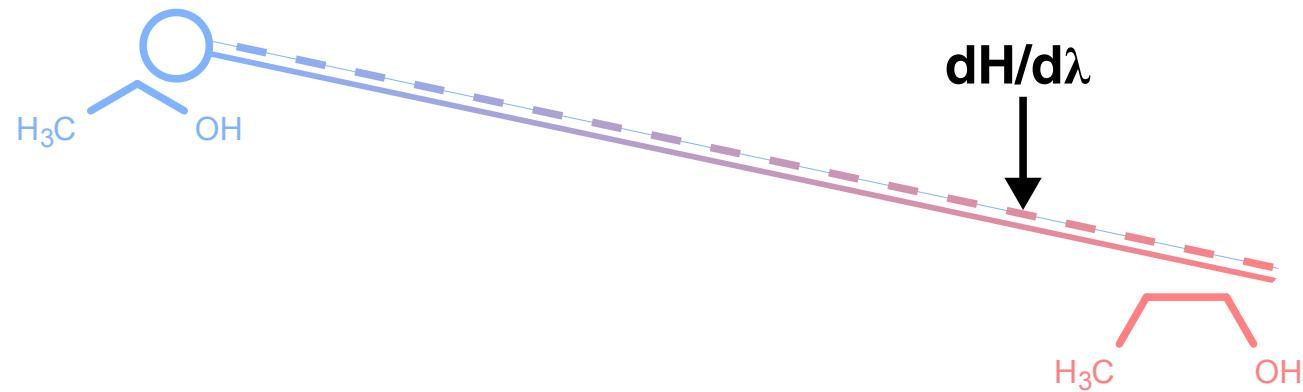


AWH

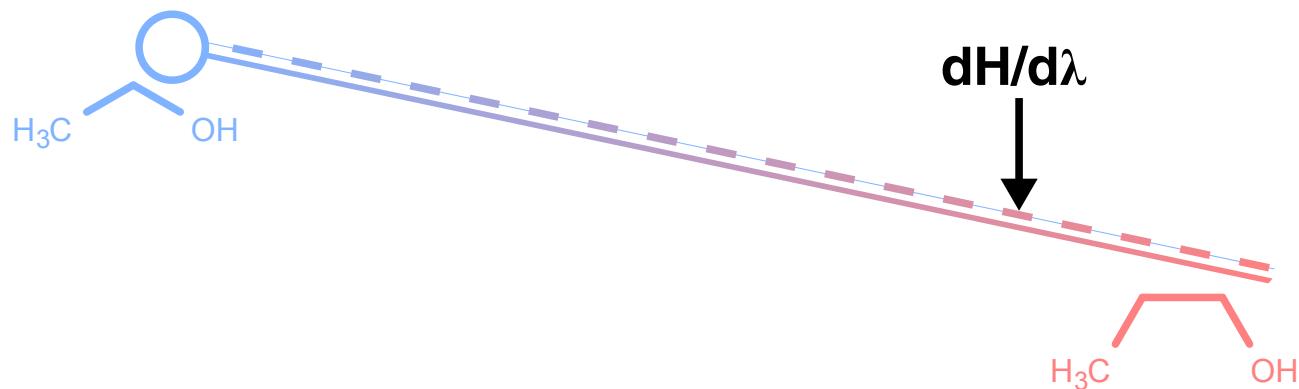


Energy/Hamiltonian- derivative based methods

Thermodynamic integration



Thermodynamic integration

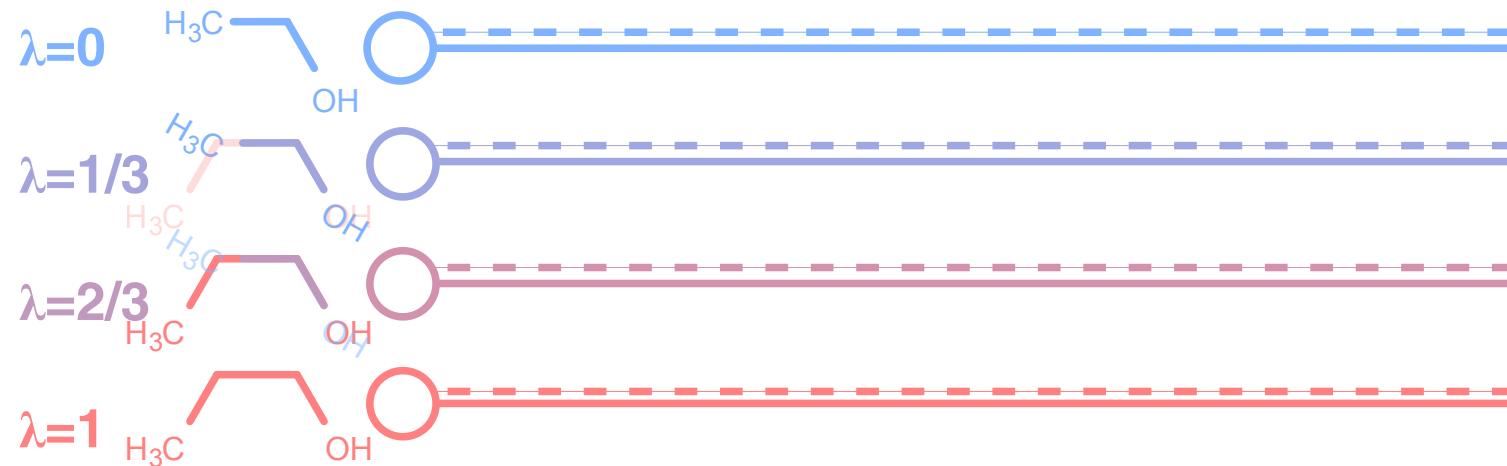


$$\Delta G = \int_{\lambda=0}^1 \left\langle \frac{dH}{d\lambda}(x) \right\rangle_\lambda d\lambda$$

Theromdynamic integration

- Cons
 - computationally expensive
 - has a systematic error for non-infinite sampling times due to the steadily moving λ value

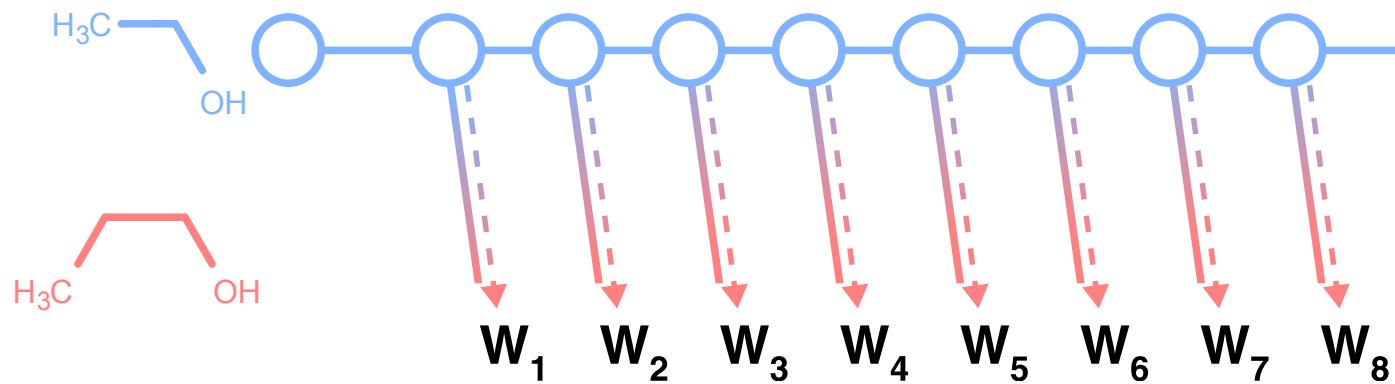
Discrete thermodynamic integration



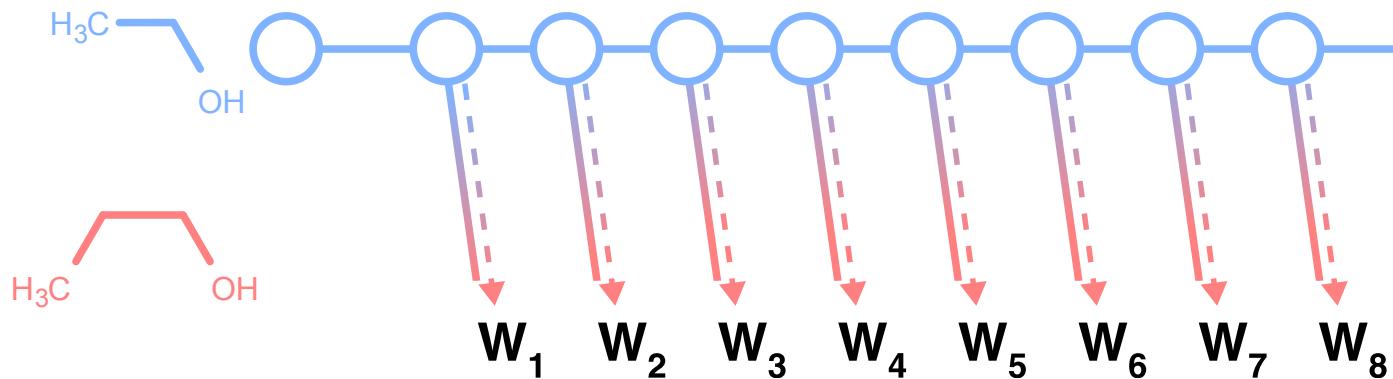
Discrete thermodynamic integration

- **Pros**
 - better approximation than slow-growth TI
- **Cons**
 - need to prepare system at different starting points
 - need some overlap between states between λ (tight λ spacing) to reasonably approximate the integral that yields the free energy.
 - still bad performance compared to other methods

Jarzynski equation



Jarzynski equation



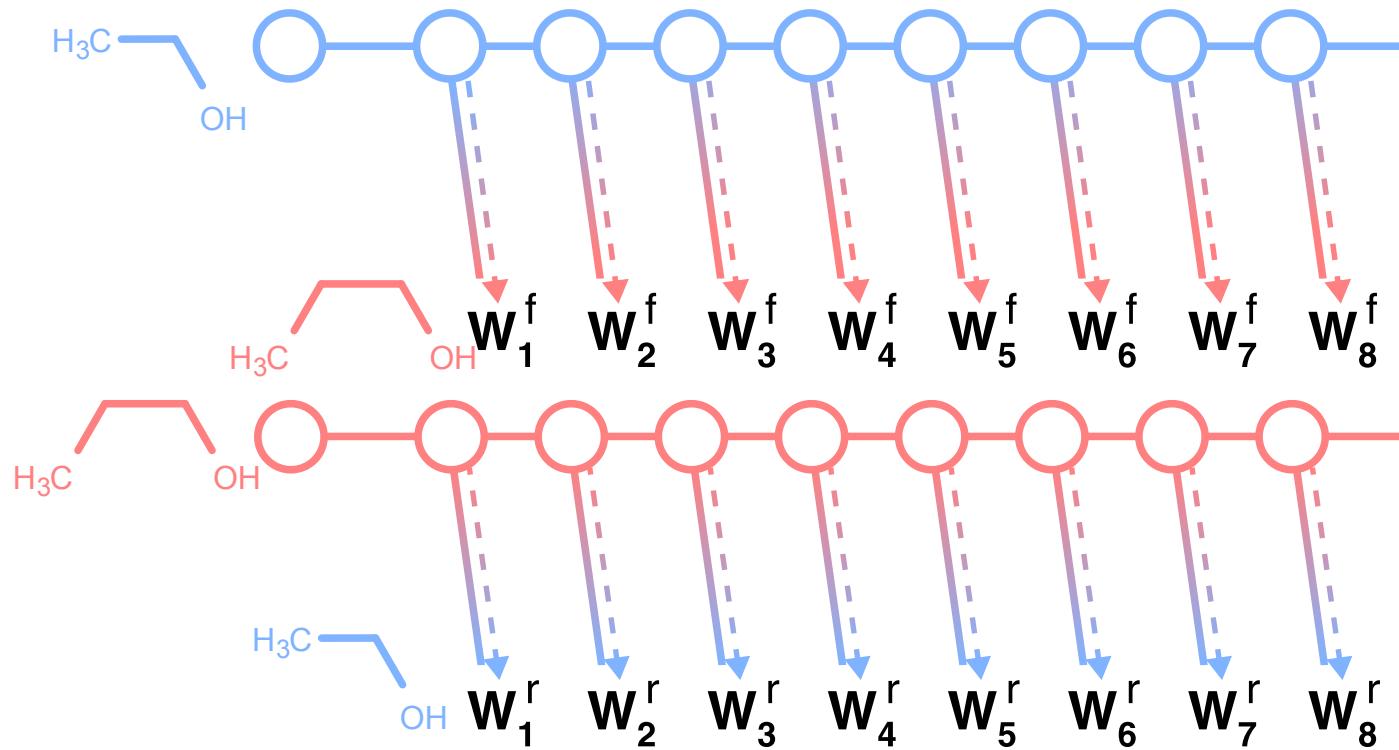
$$W = \int_{\lambda=0}^1 \frac{dH}{d\lambda}(x) d\lambda$$

$$\Delta G = -\frac{1}{\beta} \log \left(\frac{1}{N} \sum_i \exp(-\beta W_i) \right)$$

Jarzynski

- **Pros**
 - good when you don't know what the B-state looks like
 - some systematic bias for finite number of samples (long-tail distribution)
- **Cons**
 - Crooks is better in most other cases

Crooks fluctuation theorem

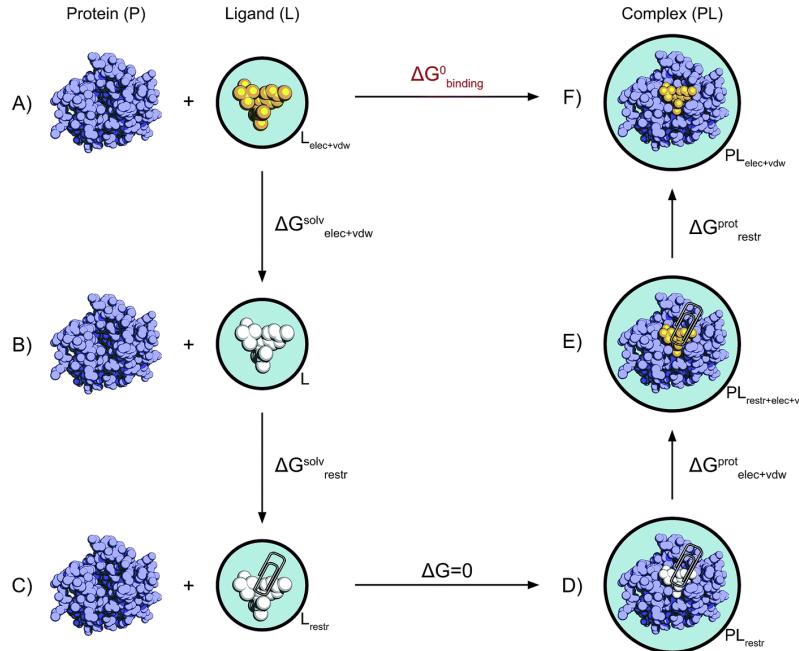


Crooks fluctuation theorem

$$\log \frac{P_f(W)}{P_r(W)} = \beta(W - \Delta G)$$

Absolute binding free energies

Absolute binding free energies



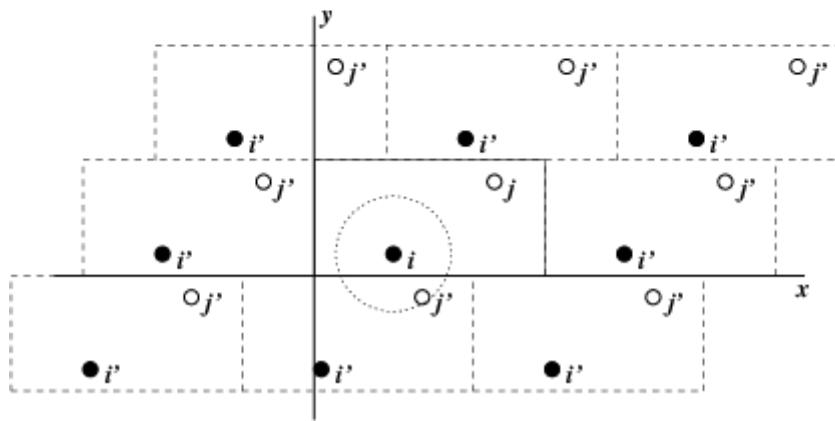
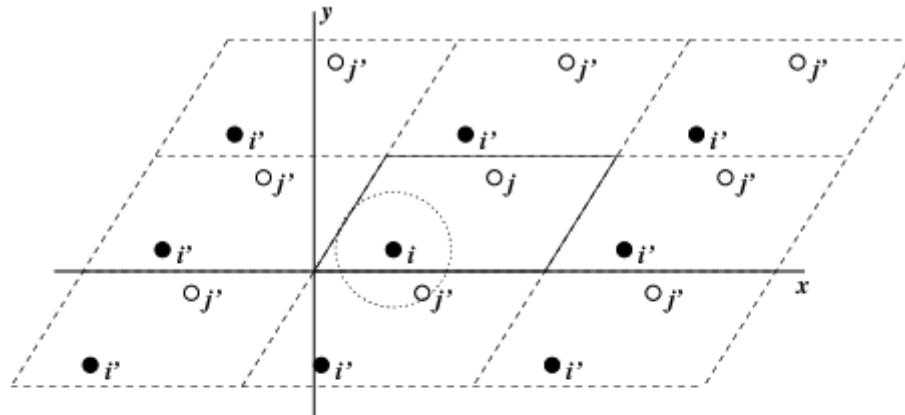
© Matteo Aldeghi “Accurate calculation of the absolute free energy of binding for drug molecules”

Simulation system setup

X-ray to GROMACS

- gmx pdb2gmx
- Watch out for
 - terminal residues
 - titration sites

Periodic boundary conditions



Adding simulation box

- 1.0 nm distance to periodic image is usually save

Adding simulation box

- rhombic dodecahedron for most efficiency (approx. 15% speed gain)

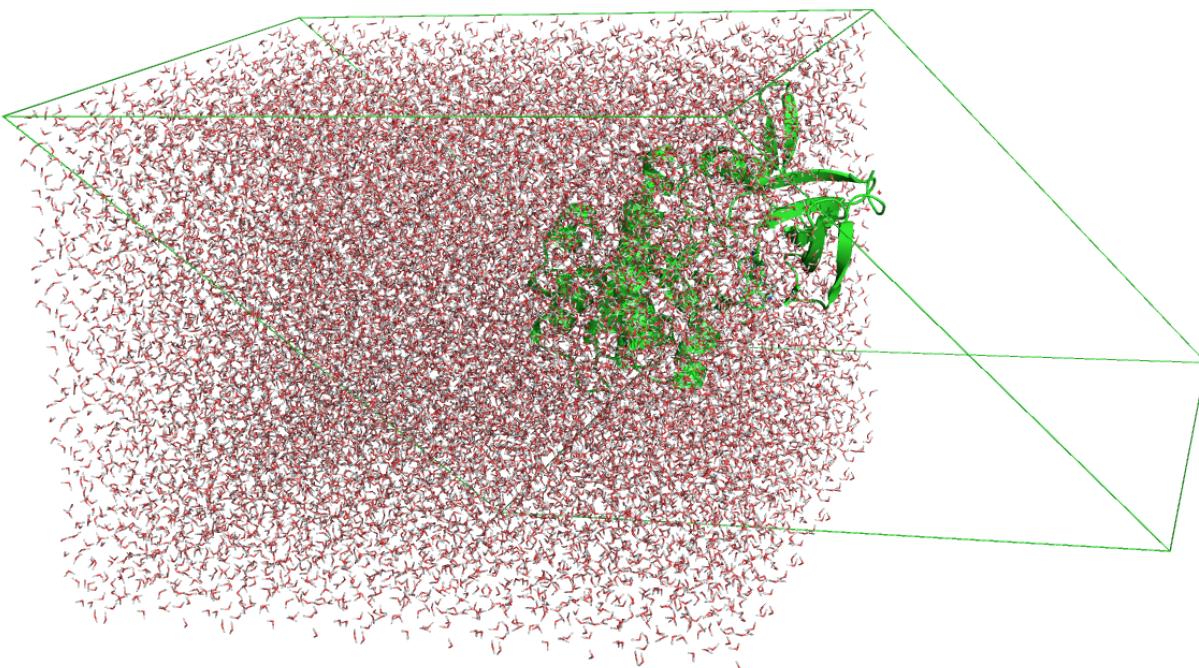
Adding simulation box



Adding water

- copy-paste block of water

Adding water



Adding ions

- 150mM NaCl
- randomly swaps water with ions
- ensure ions far enough from ligand / protein
- use GROMACS2020 or higher
- make sure to not use outdated force-fields

Setting force-field

- most popular choices amber, charmm
- for FEP chose all the force-fields if you can

Setting force-field and simulation parameters

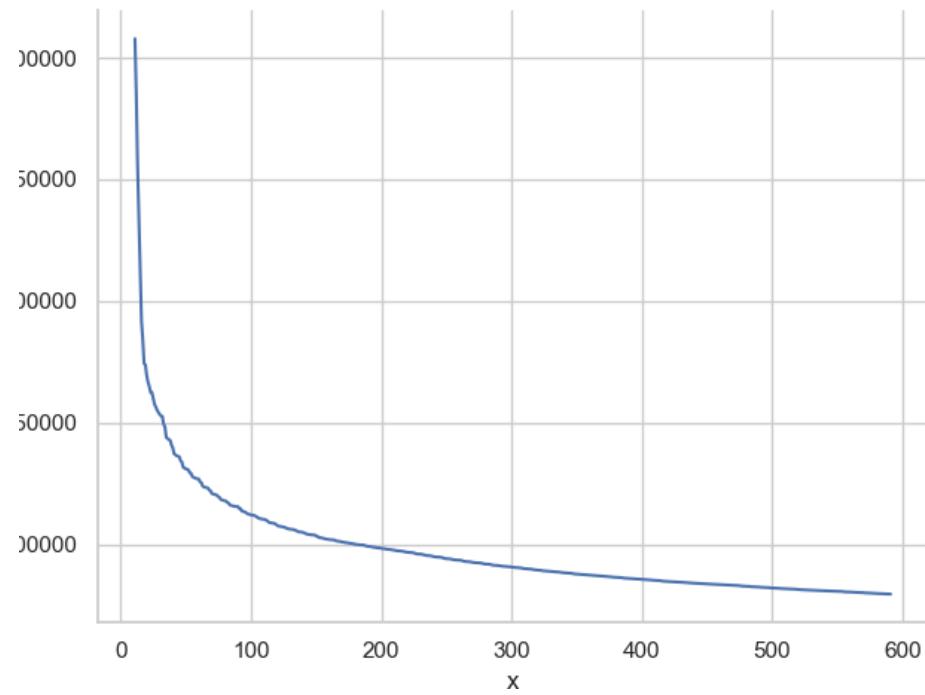
- match force-field and parameters for
 - short-range electrostatics/vdW cutoff
 - vdw interaction switch
 - dispersion correction
 - coulomb interaction type (PME, etc)
 - PME approximation: fourier-spacing * rcoulomb

Energy minimisation

- steepest descend, usually around 10,000 steps

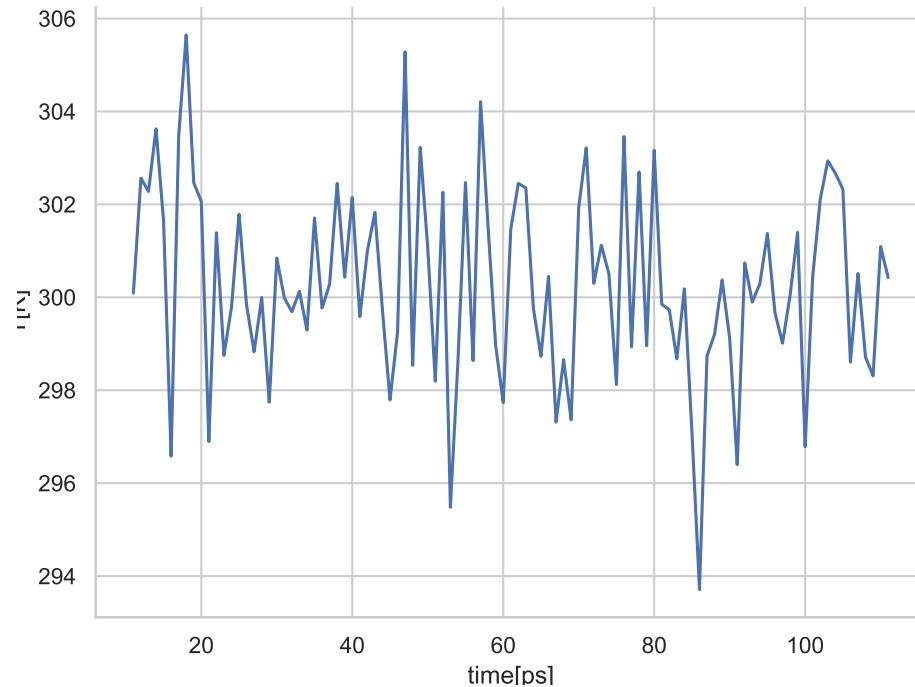
Energy minimisation

- typical output



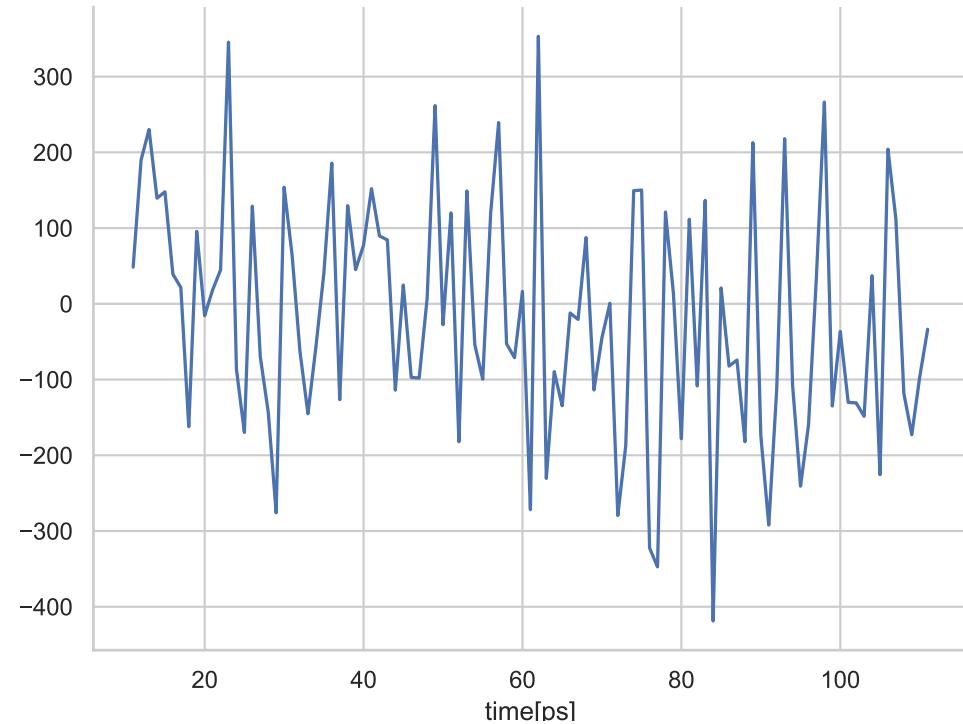
Temperature coupling

- keep heavy atoms fixed springs
- use v - rescale overall



Pressure coupling

- use Berendsen first, then Parinello-Rahman
- from GROMACS2021 on use stochastic cell rescaling



Setting up efficient equilibrium simulation

- GPUs drastically enhance performance
- Focus on sampling, rather than individual simulation length
- -multidir option
- GROMACS performance bottlenecks:
 - clock-rate
 - communication (MD step in ms)
 - **not** memory bound

Questions

