Argonne
NATIONAL
LABORATORY

... for a brighter future

A Linear Programming – Linear Assignment Approach for the Protein Morphing Problem

Sanghyun Park and Mihai Anitescu

Mathematics and Computer Science Division

Argonne National Laboratory

U.S. Department of Energy

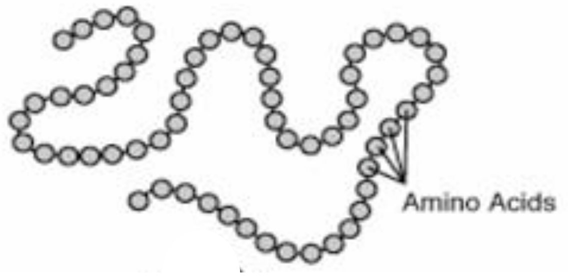
UChicago ▸ Argonne LLC

A U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC

Paradigm of Molecular Biology

Protein folding

Sequence → Structure → Function



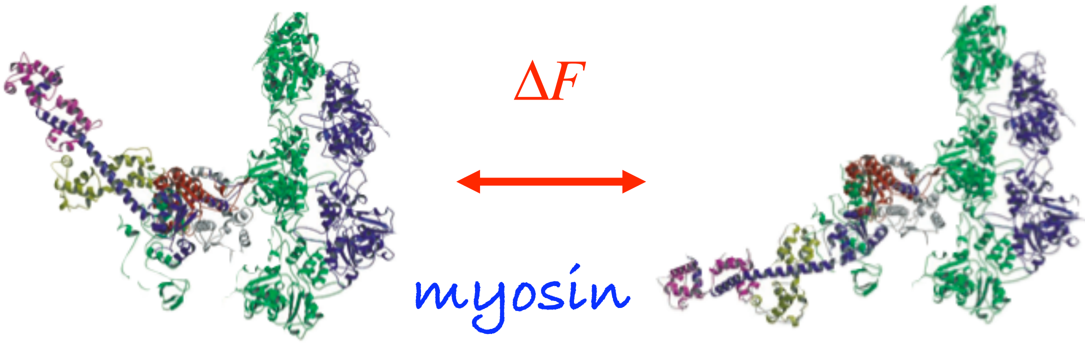
- Function is emergent.
- The structure of the protein is key to its function.
- Different structures = different functions

myoglobin

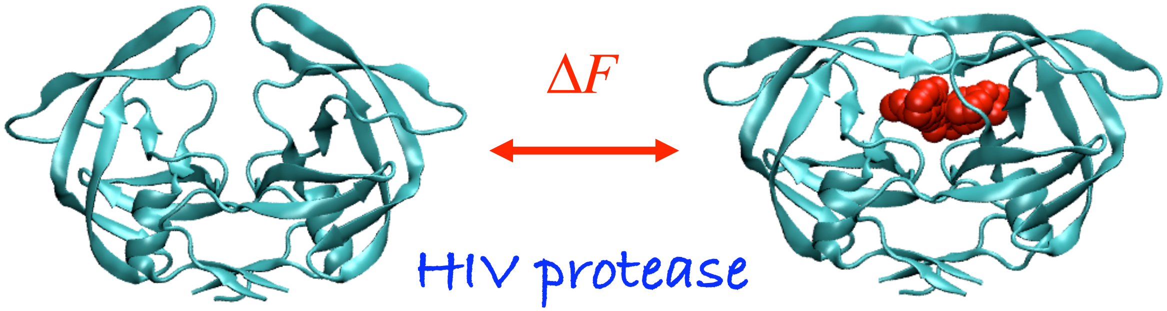
High concentrations of myoglobin in muscle cells allow organisms to hold their breaths longer. — Wikipedia

Conformational Change is Important …

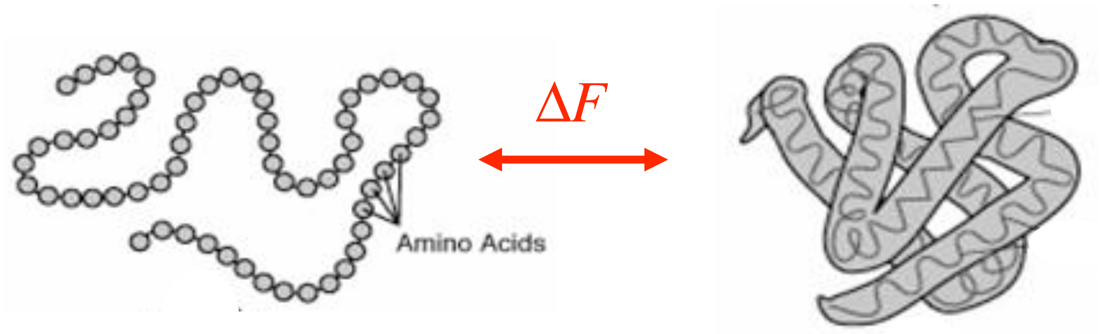
Domain motion essential to expose to binding.



Ligand binding (Drug discovery)



Protein folding



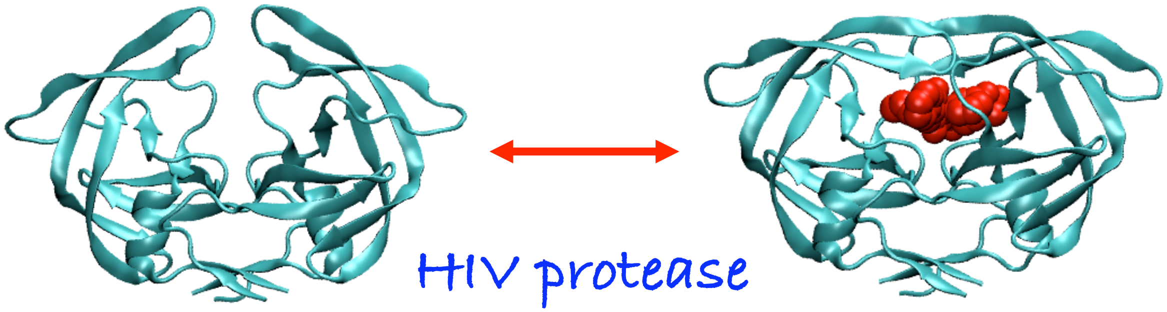
Why are free energy variations important.

□ They tell whether state A is favored over state B.

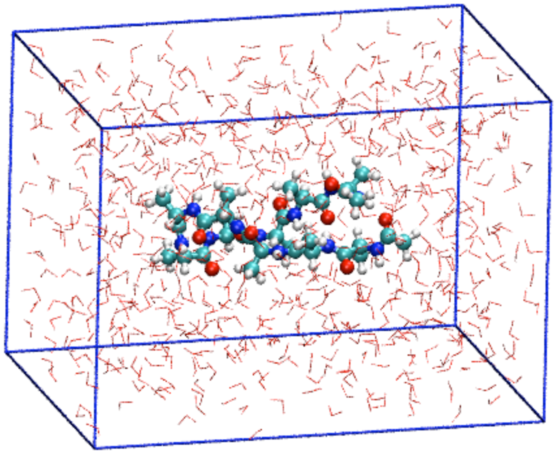
□ For instance, is the configuration that has the drug molecule in it favored over the one that doesn’t (i.e. will this drug bind in the prescribed position at a given temperature)?

□ The key quantity that quantifies the relative odds is $\exp\left(\frac{-\Delta E}{k_B T}\right)$

Ligand binding
(Drug discovery)



Conformational Free Energy

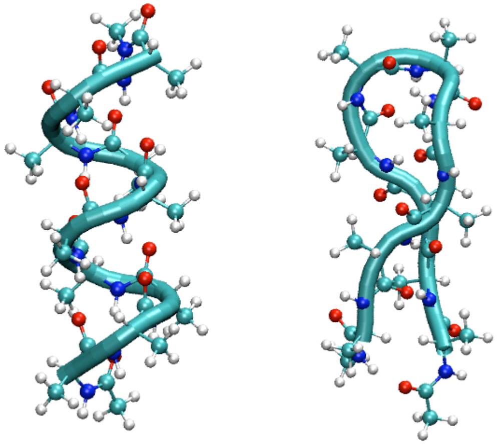


$$e^{-\beta F(\mathbf{X})} = \int d\mathbf{Y}\, e^{-\beta U(\mathbf{X},\mathbf{Y})}$$

X: protein coordinates
Y: water coordinates
U(X,Y): potential energy function
F(X): conformational free energy

$$\beta = \frac{1}{k_B T}$$

deca-alanine



ΔF = F(B) - F(A)

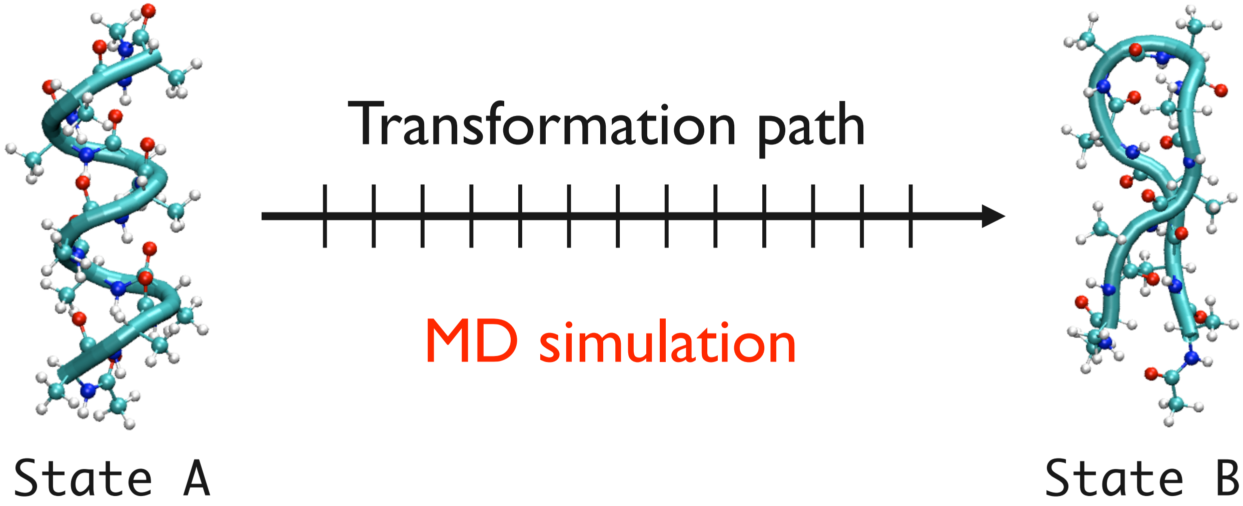
Computations of F(X) don’t work.

➡ Compute ΔF directly.

Why direct computations of the free energy do not work

- Calculations of the free energy are enormously expensive.
- We have to average out the water coordinates.
- In a box, there are a few thousand water molecules, that is a few thousand water coordinates – and that is for a small protein !!!
- One must estimate nasty quadrature over a 1000 dimensional space (for fixed X, the energy function has lots of minima).
- This difficulty is known as the curse of dimensionality: The fact that in excessively large dimensions the sample density decreases exponentially.
- This is thus, a very hard problem

Free Energy Perturbation --- computing the free energy difference



Transformation path

MD simulation

State A

State B

$$e^{-\beta\Delta F} = \left\langle e^{-\beta[U(\mathbf{A},\mathbf{Y})-U(\mathbf{B},\mathbf{Y})]}\right\rangle_{\mathbf{A}}$$

Zwanzig (1954)

Bennett (1976): bi-directional

Jarzynski (1997): non-equilibrium

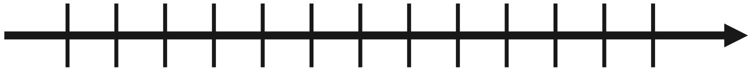
Computational cost = days x hundreds

Ideal for massive parallelization

Computing free energy variations

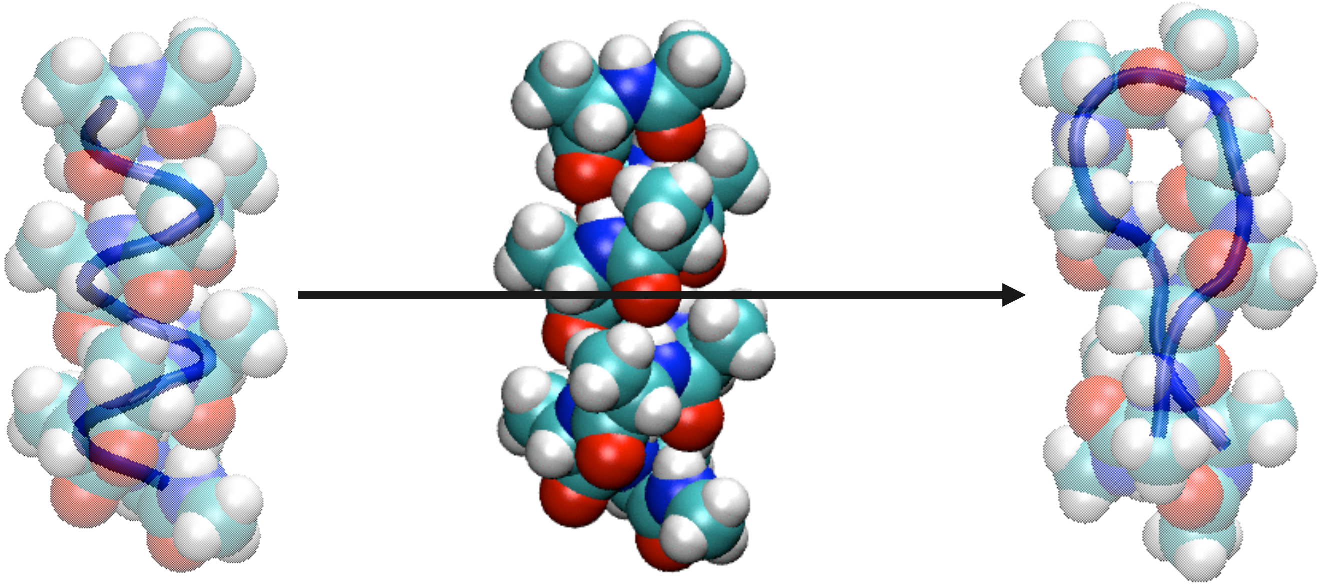
- I have just said that computing free energies is hard, so how is this possible?
- The trick is to find a "good" path in the phase space.

- Then divide the path in small segments



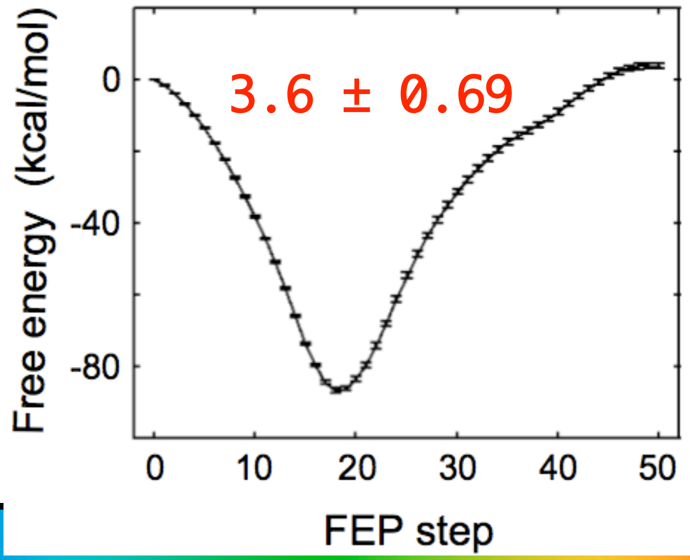
- If the path is good, the free energy difference including its variance can be computed *relatively* easily by using some version of importance sampling
- So we have transformed the problem into the one of finding a good path in phase space.

One such path: Direct Morphing



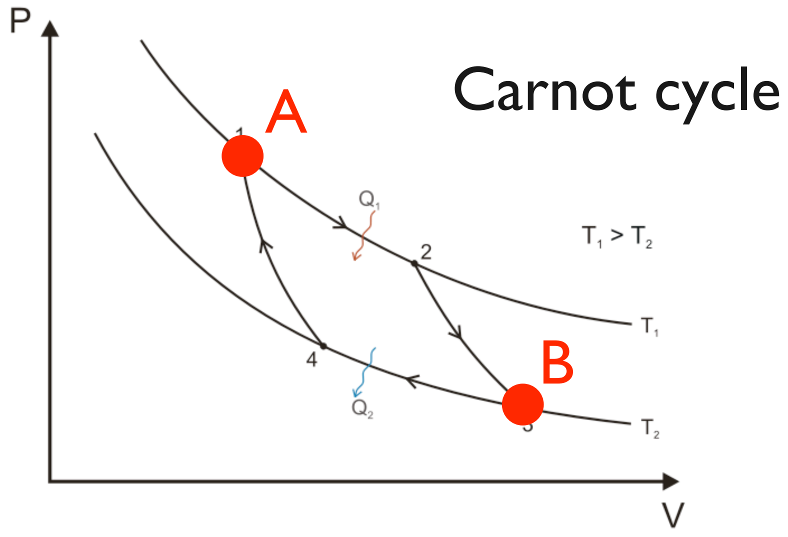
$$\mathbf{x}_n = (1-\lambda)\,\mathbf{a}_n + \lambda\,\mathbf{b}_n$$

$$0 \le \lambda \le 1$$



3.6 ± 0.69

Thermodynamic Cycle



Carnot cycle

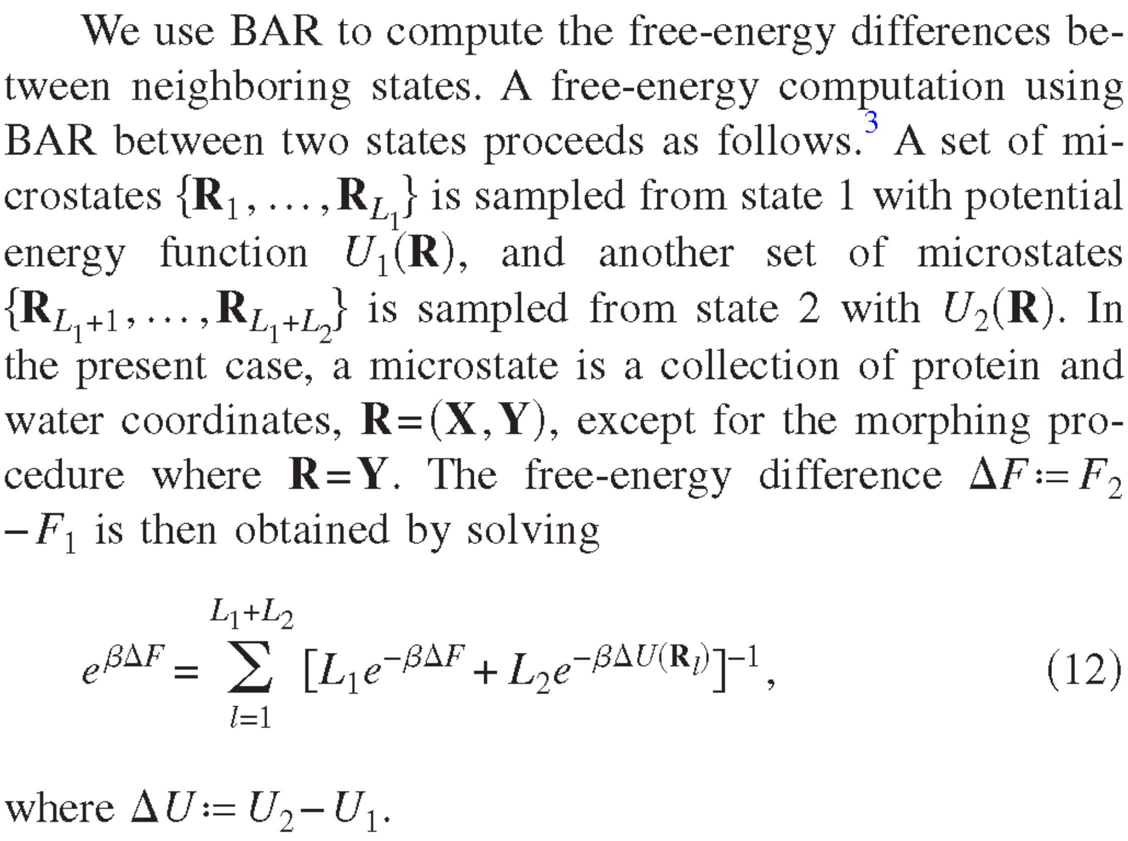
ΔF depends only on the end states, not on the path.

Computer simulations are not bound by reality.

Computing Free Energy Between 2 States,

- We sample the distribution of states attached to each potential function – "conformation" – by using molecular dynamics.
- E.g., we start a molecular dynamics calculation with the potential function for fixed protein atoms, but moving water atoms, until the simulation "relaxes" and the system can be assumed ergodic.
- With the samples we create a free energy estimate using Bennett's acceptance ration method -- BAR

Bennet's Acceptance Ratio Method



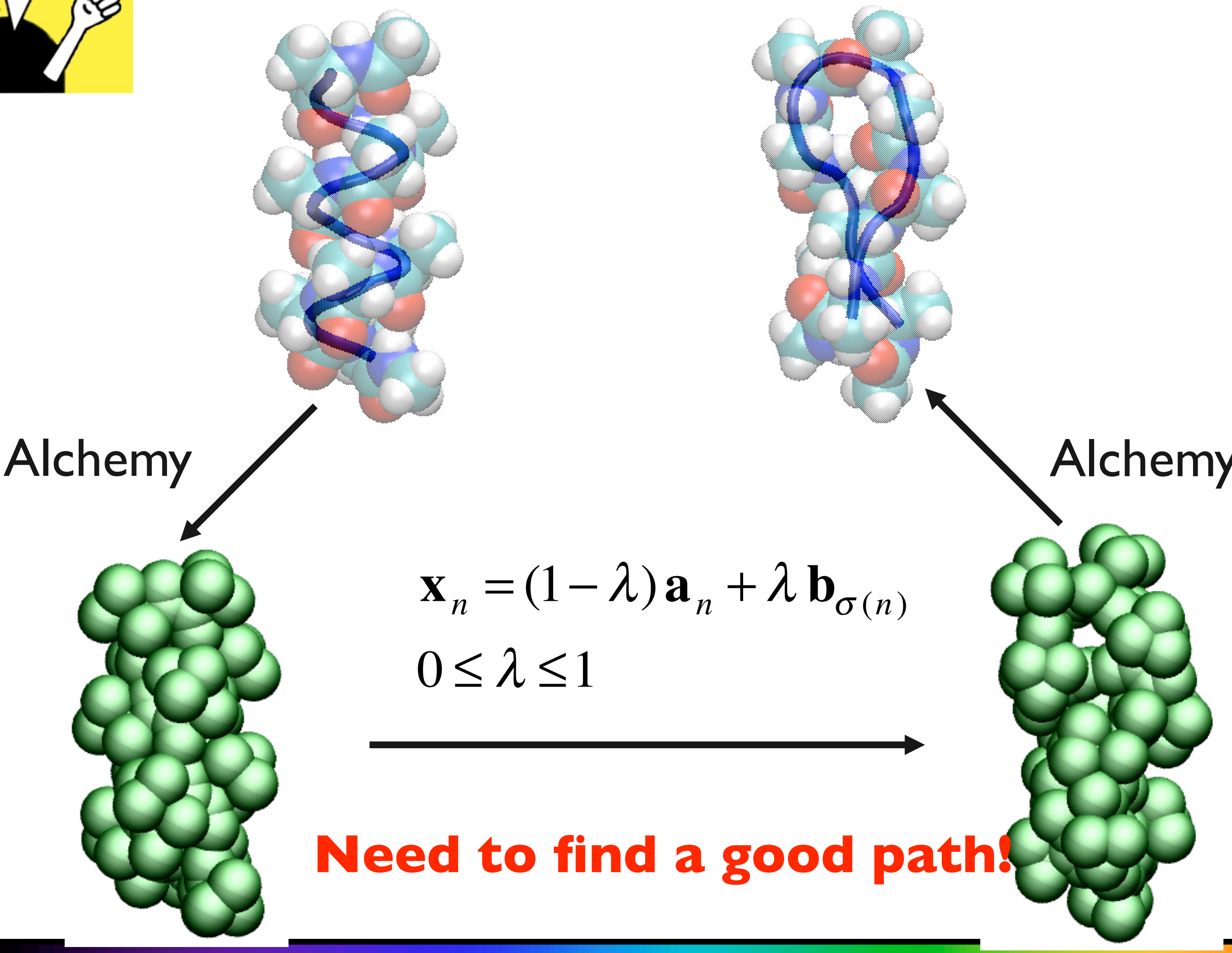
We use BAR to compute the free-energy differences between neighboring states. A free-energy computation using BAR between two states proceeds as follows.[3] A set of microstates $\{\mathbf{R}_1, \ldots, \mathbf{R}_{L_1}\}$ is sampled from state 1 with potential energy function $U_1(\mathbf{R})$, and another set of microstates $\{\mathbf{R}_{L_1+1}, \ldots, \mathbf{R}_{L_1+L_2}\}$ is sampled from state 2 with $U_2(\mathbf{R})$. In the present case, a microstate is a collection of protein and water coordinates, $\mathbf{R}=(\mathbf{X},\mathbf{Y})$, except for the morphing procedure where $\mathbf{R}=\mathbf{Y}$. The free-energy difference $\Delta F := F_2 - F_1$ is then obtained by solving

$$e^{\beta \Delta F} = \sum_{l=1}^{L_1+L_2} [L_1 e^{-\beta \Delta F} + L_2 e^{-\beta \Delta U(\mathbf{R}_l)}]^{-1}, \tag{12}$$

where $\Delta U := U_2 - U_1$.

- For the equation to be nonsingular, we need the second term in the sum to be significant – good overlap between ensembles.
- Energy difference is small: good transformation path AND broken down in pieces.

Morphing for Dummies



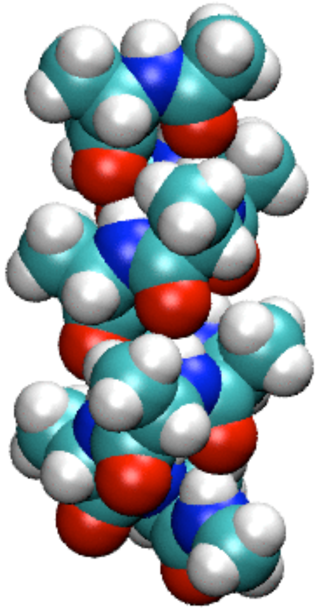
$$\mathbf{x}_n = (1-\lambda)\,\mathbf{a}_n + \lambda\,\mathbf{b}_{\sigma(n)}$$

$$0 \le \lambda \le 1$$

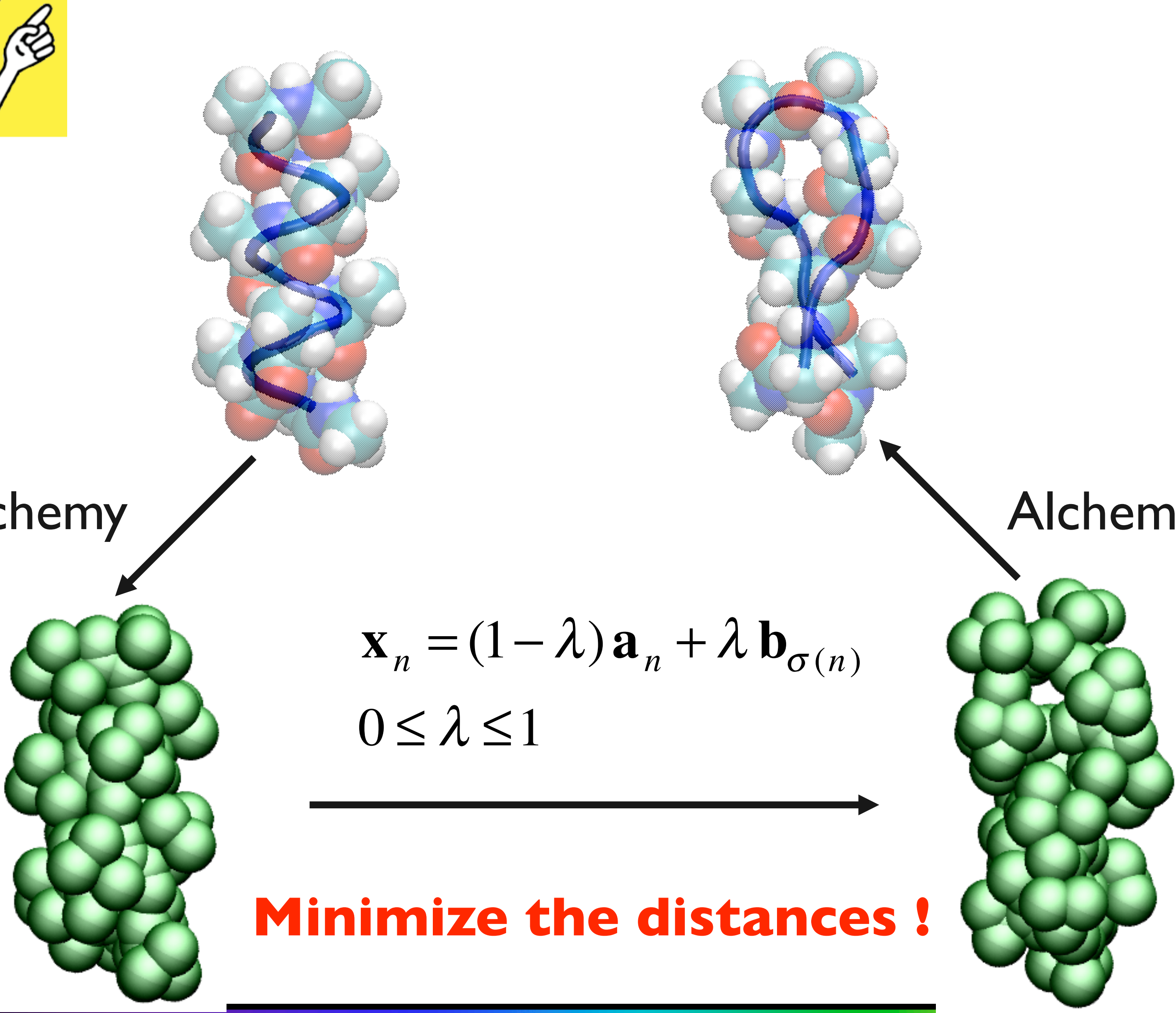
Need to find a good path!

How to find a good path in phase space

- But note that mapping the particles to the same ones from the linear structure may result sometimes in enormous traveled paths for some of them.
- And we are not bound by having each element of the trajectory feasible in the sense of it corresponding to a real compound. Such paths are very hard to find. (R. Elber, Curr. Opin. Struct. Biol. **15, 151 2005).**
- Therefore, we look for different perturbations which have a chance of resulting in smaller per unit energy variations.
- What if we actually change the atoms themselves? This will allow us to make smaller steps in energy steps at the morphing step.



Morphing for Dummies



Alchemy

Alchemy

$$\mathbf{x}_n = (1-\lambda)\,\mathbf{a}_n + \lambda\,\mathbf{b}_{\sigma(n)}$$

$$0 \le \lambda \le 1$$

Minimize the distances !

Least-Squares Morphing Problem

$$\boxed{\min_{\sigma\in\Pi_N}\sqrt{\frac{1}{N}\sum_{n=1}^{N}\left\|\mathbf{a}_n-\mathbf{b}_{\sigma(n)}\right\|^2}}$$

$$\Pi_N=\{N\text{-permutations}\}$$

$$\sum_{n=1}^{N}\left\|\mathbf{a}_n-\mathbf{b}_{\sigma(n)}\right\|^2=\sum_{n=1}^{N}\left\|\mathbf{a}_n\right\|^2+\sum_{n=1}^{N}\left\|\mathbf{b}_{\sigma(n)}\right\|^2-2\sum_{n=1}^{N}\mathbf{a}_n\cdot\mathbf{b}_{\sigma(n)}$$

$$\boxed{\max_{\sigma\in\Pi_N}\sum_{n=1}^{N}\mathbf{a}_n\cdot\mathbf{b}_{\sigma(n)}}$$

$$P_{ij}(\sigma)=\begin{cases}1, & j=\sigma(i)\\ 0, & \text{otherwise}\end{cases}\qquad \mathbf{A}=\left(\mathbf{a}_1\cdots\mathbf{a}_N\right)\qquad \mathbf{B}=\left(\mathbf{b}_1\cdots\mathbf{b}_N\right)$$

$$\boxed{\max_{\sigma\in\Pi_N}\operatorname{Tr}\left[\mathbf{A}\,\mathbf{P}(\sigma)\,\mathbf{B}^T\right]}$$

Linear-Programming Solution

Original problem - Combinatorial search

$$\text{P1}: \max_{\mathbf{P}\in\Omega_N} \text{Tr}\left[\mathbf{A}\,\mathbf{P}\,\mathbf{B}^T\right] \qquad \Omega_N = \{N\times N \text{ permutation matrices}\}$$

Birkoff's theorem: $\Omega_N = \{\text{Vertices of } \Gamma_N\}$

Fundamental theorem of LP:

Solution of P2 $\in$ $\{\text{Vertices of } \Gamma_N\}$

Relaxed problem - Linear programming

$$\text{P2}: \max_{\mathbf{W}\in\Gamma_N} \text{Tr}\left[\mathbf{A}\,\mathbf{W}\,\mathbf{B}^T\right] \qquad \Gamma_N = \{N\times N \text{ bistochastic matrices}\}$$

$$W_{ij} \geq 0 \qquad \sum_i W_{ij} = 1 \qquad \sum_j W_{ij} = 1$$

Another way to look at it – it is a linear assignment problem!

Define $A = \left\{ a_j^i \right\}_{i=1,p;j=1,N}$ $B = \left\{ b_j^i \right\}_{i=1,p;j=1,N}$

$$\max \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} \left\langle a_i, b_j \right\rangle$$

$$\sum_{j=1}^{N} w_{ij} = 1, i = 1,2,\ldots,N; \ \sum_{i=1}^{N} w_{ij} = 1, j = 1,2,\ldots,N$$

$$\left\{ w_{ij} \right\}_{i,j=1,2\ldots,N} \in F_N, w_{ij} \in \{0,1\}, i,j = 1,2,\ldots,N$$

$$\max \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} \left\langle a_i, b_j \right\rangle$$

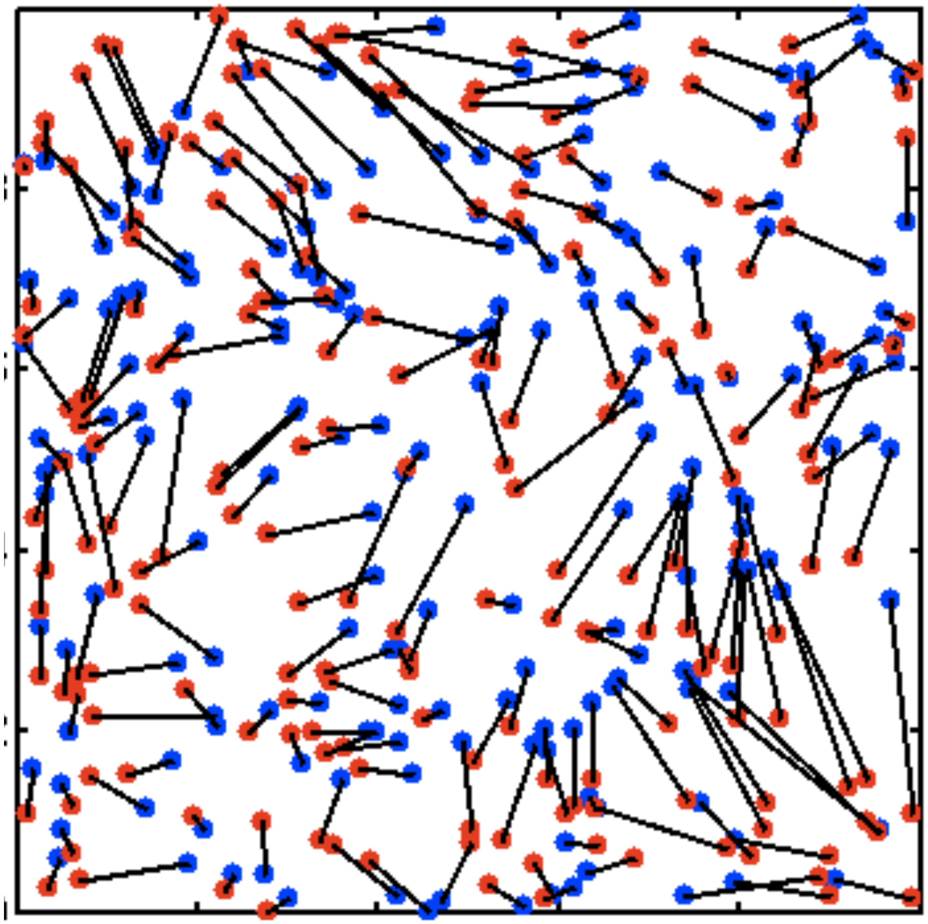
$$\sum_{j=1}^{N} w_{ij} = 1, i = 1,2,\ldots,N; \ \sum_{i=1}^{N} w_{ij} = 1, j = 1,2,\ldots,N$$

$$w_{ij} \geq 0, i,j = 1,2,\ldots,N$$

It is a linear assignment problem !

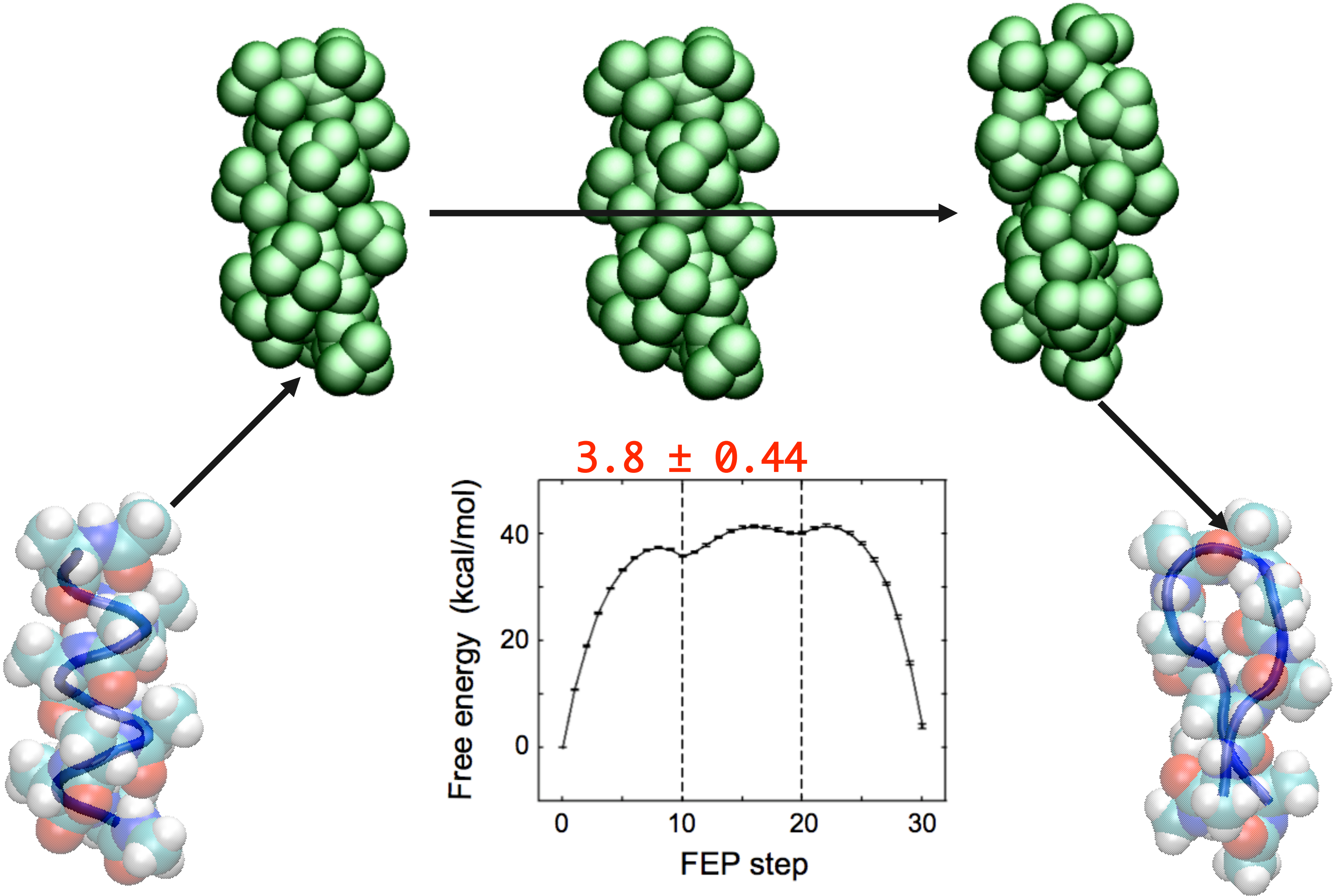
But we had to identify this in the LS formulation!!

Least-Squares Permutation



200 points

Least-Squares Morphing

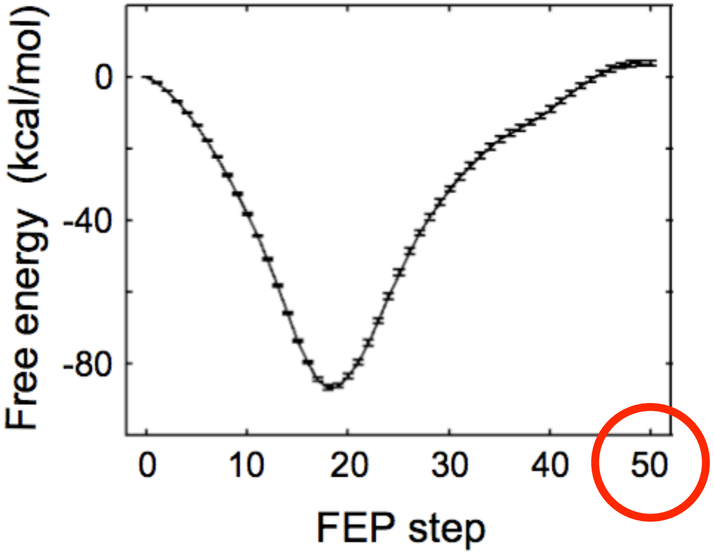


3.8 ± 0.44

Direct vs. Least-Squares Morphing

Direct

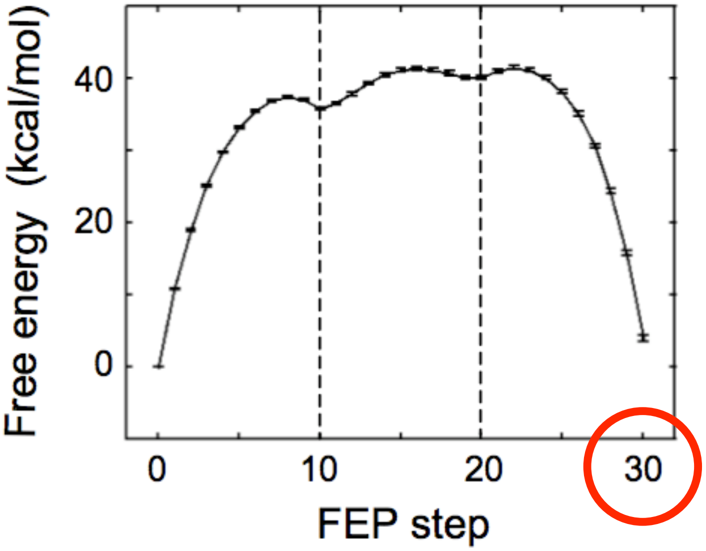
RMS distance = 8.4 Å



3.6 ± 0.69

Least-squares

RMS distance = 2.1 Å



3.8 ± 0.44

Discussion of the results.

- Each one of the steps in the molecular dynamics simulation is done with NAMD.
- NAMD is enormously expensive. One free energy perturbation step (FEP) takes *20 CPU hours* for the deca-alanine.
- In this case, dummying the atoms takes 10 FEP, our least-squares morphing takes 10 FEPs, and the un-dummying of the atoms takes another 10 FEPs. Compare with 50 FEP steps for the original step. We save 600 CPU hours. (Morphing with LP takes 1-2 seconds).
- We solve 2 linear programming – linear assignment problems. There are better ways to do linear assignments, but, give the small computational cost, it is not worth to do it.
- But, more importantly, we can compute a more accurate path 0.44 versus 0.69 kcal/mol.

About NAMD– Molecular Dynamics Software

- Why it is difficult: very expensive potential – CHARM 22.

$$\begin{aligned}U(\vec{R}) = {} & \sum_{\text{bonds}} K_{\text{b}}(b-b_0)^2 + \sum_{\text{UB}} K_{\text{UB}}(S-S_0)^2 + \\ & \sum_{\text{angle}} K_\theta(\theta-\theta_0)^2 + \sum_{\text{dihedrals}} K_\chi(1+\cos(n\chi-\delta)) + \\ & \sum_{\text{impropers}} K_{\text{imp}}(\varphi-\varphi_0)^2 + \\ & \sum_{\text{nonbond}} \epsilon\left[\left(\frac{R_{\text{min}_{ij}}}{r_{ij}}\right)^{12} - \left(\frac{R_{\text{min}_{ij}}}{r_{ij}}\right)^{6}\right] + \frac{q_i q_j}{\epsilon_1 r_{ij}}\end{aligned}$$

•Simulations done at constant temperature, using the Langevin thermostat and Langevin-piston barostat.

•Time step: 1fs, it is run for 1ns (1000 steps !), the trajectories sampled at 100fs are used as samples for estimating the integral.

Why computing a path of small error is not trivial

- We note that getting a good path is still a matter of heuristics.
- We are interested in the overall error, not just the asymptotic error estimate for one segment, which may have the usual Monte Carlo behavior.
- Therefore it is not clear how the estimate behaves with more segments – therefore the cost of reducing the error for the original approach to the level we have obtained is hard to fathom.
- We have to some extend added a new capability to molecular dynamics.

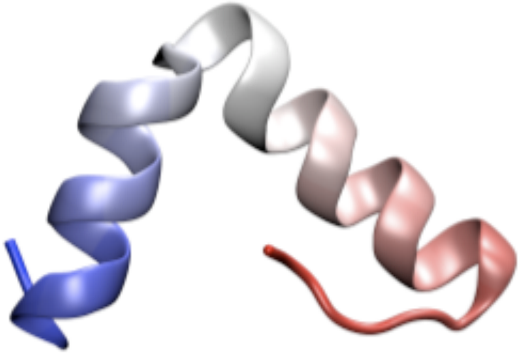
WW Domain

Direct morphing

RMS distance = 11.3 Å
100 FEP steps
12.9 ± 3.2 kcal/mol

Least-squares morphing

RMS distance = 3.4 Å
50 + 30 FEP steps
13.3 ± 1.1 kcal/mol

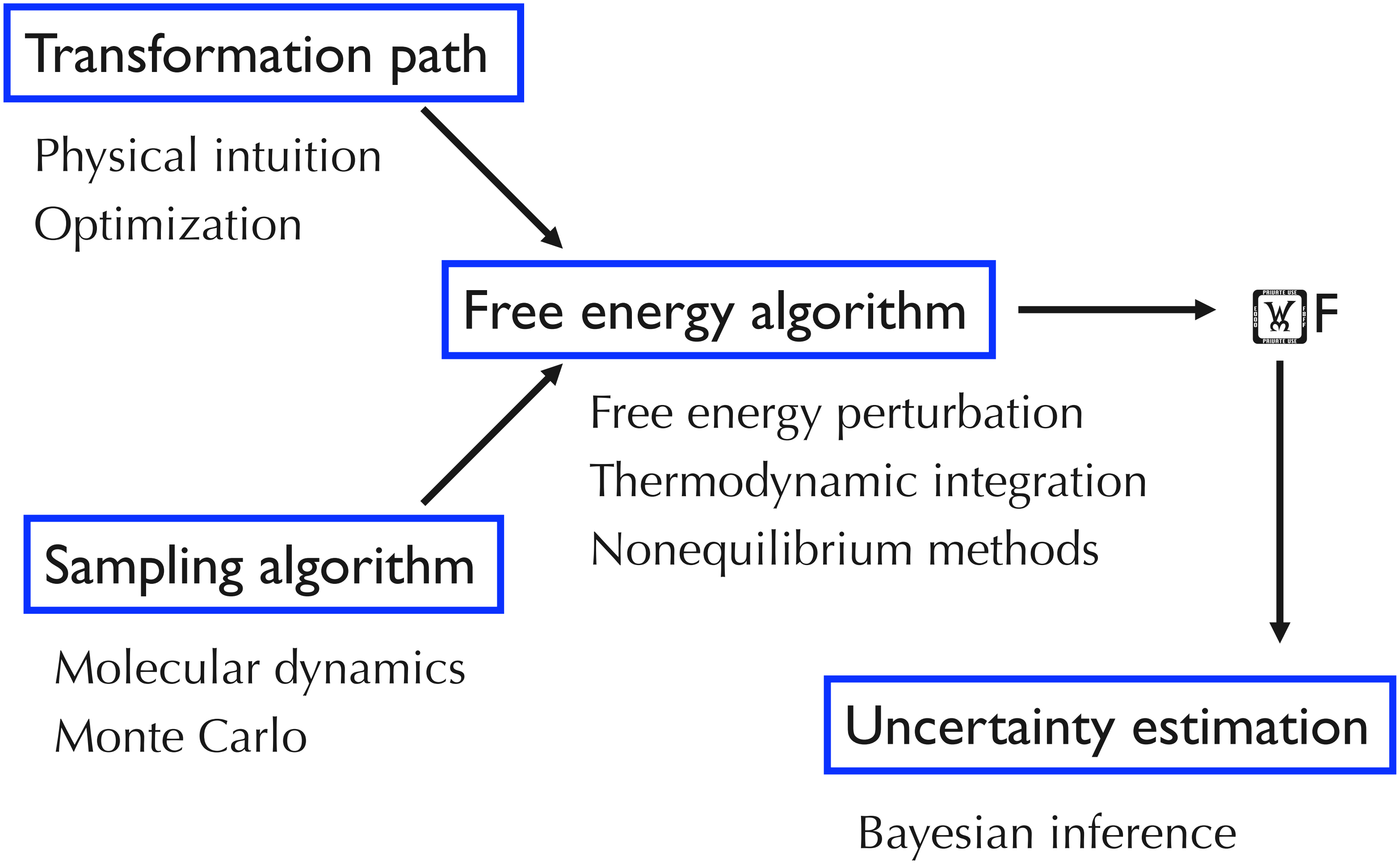


To our knowledge, this is the first time the WW domain protein has been computed at all with this low of an error estimate.

Conclusion

- Morphing can result in much sharper estimates of free energy differences between different conformations.
- We have shown that least-square morphing obtains an excellent free energy perturbation path.
- We have shown that the path can be obtained in polynomial time, by using linear programming – linear assignment.
- We have obtained 100s of CPU hour computational time savings, with much more accurate FE difference estimates.

What can applied math do for FEC ?



Transformation path

Physical intuition

Optimization

Free energy algorithm

ΔF

Free energy perturbation

Thermodynamic integration

Nonequilibrium methods

Sampling algorithm

Molecular dynamics

Monte Carlo

Uncertainty estimation

Bayesian inference