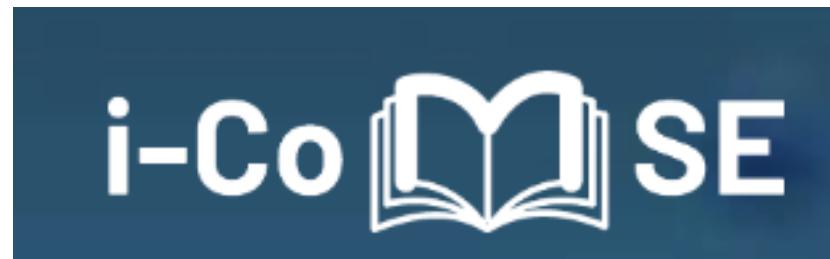


Enhanced Sampling Workshop: Day 1

Michael R. Shirts

Department of Chemical and Biological Engineering
University of Colorado Boulder

3rd i-CoMSE Workshop: Methods for Enhanced Sampling
March 20, 2023



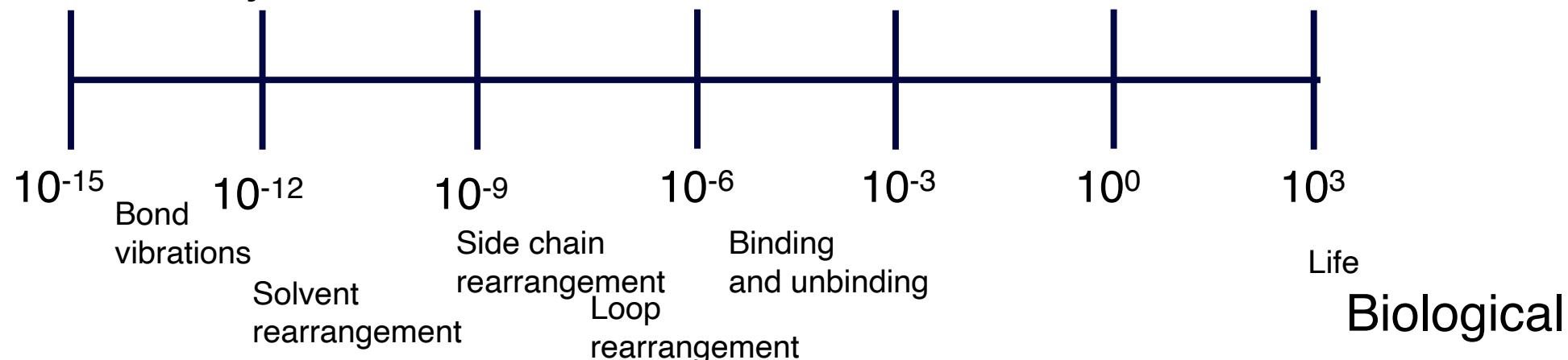
Overall Outline

- Part 1: Logistics
- Part 2: Why enhanced sampling?
- Part 3: How enhanced sampling?

Why enhanced sampling?

- The Multiple Time Scale Problem

- Interesting molecular processes involve multiple time scales!
- If it were just long time scale, we could just do that directly!



Catalysis

Enhanced Sampling for Thermodynamics

- Example: Calculating a binding affinity
- Break some physics laws, but not others
- Keep:
 - $p(x) \propto \exp(-\beta U(x))$
 - Thermodynamics
- Break:
 - Continuity of molecular trajectories
 - $F=ma$
 - Kinetics

Enhanced Sampling for Kinetics

- Break some physics laws, but not others
- Break:
 - $F=ma$
 - $p(x) \propto \exp(-\beta U(x))$ (maybe)
- BUT break in ways that one can recover the results

A complete overview of using simulations to calculate thermodynamics (in 5 min)

1. All thermodynamic properties are ratios of highly multidimensional integrals over coordinates

$$e^{-f(\vec{a}_1) + f(\vec{a}_2)} = \frac{\int e^{-u(\vec{a}_1, \vec{x})} d\vec{x}}{\int e^{-u(\vec{a}_2, \vec{x})} d\vec{x}}$$

$$f = \beta A \quad u(x) = \beta U(x)$$

x = all coordinates of the system

a = parameters of the Hamiltonian of the system (T,P, force field)

$$\langle O(\vec{a}) \rangle = \frac{\int O(\vec{a}, \vec{x}) e^{-u(\vec{a}, \vec{x})} d\vec{x}}{\int e^{-u(\vec{a}, \vec{x})} d\vec{x}}$$

O is some observable of the system: volume, radius of gyration, etc.

A complete overview of using simulations to calculate thermodynamics (in 5 min)

Use molecular simulations to perform Monte Carlo integration of these ratios of integrals

$$\langle O \rangle_i = \int O(\vec{x}) p_i(\vec{x}) d\vec{x} = \frac{1}{N} \sum_{n=1}^N O(\vec{x}_n)$$

\vec{x}_n sampled from $p_i(x)$

- This is why when you calculate average energy from a simulation trajectory, you don't use Boltzmann factors - they were automatically included by running the simulation!

A complete overview of using simulations to calculate thermodynamics (in 3 slides)

- Generate samples from $p_i(x)$ using:
 - Markov Chain Monte Carlo (MCMC)
 - Given the current state,
 - assuming it has the right distribution,
 - select a new state that preserves this given probability distribution.
 - Eventually*, will converge to the right distribution
- Molecular dynamics
 - ACTUALLY a type of MCMC, in the direction of force, with 100%* acceptance
 - Apply Newton's equation of motion: $dx^2/dt = -m_i^{-1} \nabla U(x)$:
 - solve the coupled PDE by as an initial value problem.
- Whatever the method, it will have long correlation times

A complete overview of using simulations to calculate thermodynamics

3. Identify improved ways to accelerate sampling
through the multidimensional space of interest

Umbrella sampling

Replica exchange

Hamiltonian replica exchange

Accelerated MD

Multicanonical simulation

Adaptive biasing force

Weighted ensemble

Metadynamics



4. Identify improved ways to estimate multidimensional
integrals given a set of molecular simulation data

Free energy perturbation

Bennett acceptance ratio

WHAM

MBAR . . .

"Taco Bell" formulation of enhanced sampling

Limited number of stat mech ingredients



"Taco Bell" formulation of enhanced sampling

Combinatorially large number of methods



"Taco Bell" formulation of enhanced sampling

Limited number of ingredients

Construct bias along a collective variable



Introduce dynamics along a auxiliary variable



Interpret a derivative as a expectation value



Assume things are Gaussian



Express as a maximum likelihood problem



Accumulate partition function in some DOF adaptively



Integrate out degrees of freedom



Use importance sampling to calculate unsampled averages



Let's understand the Boltzmann distribution some more

- It's often much clearer to work with **probability** than with energies.

$$p_i(x) \propto e^{-\beta U_i(x)}$$

$$p_i(x) = \frac{e^{-\beta U_i(x)}}{Q_i}$$

Means "proportional to"

"Normalizing constant" or
"partition function"

$$p_i(x) = e^{\beta A_i - \beta U_i(x)}$$

Where $A_i = -k_B T \ln Q_i$, the free energy

We just need to do some sampling

- Our goal: sample representative x_i from all of the distribution:

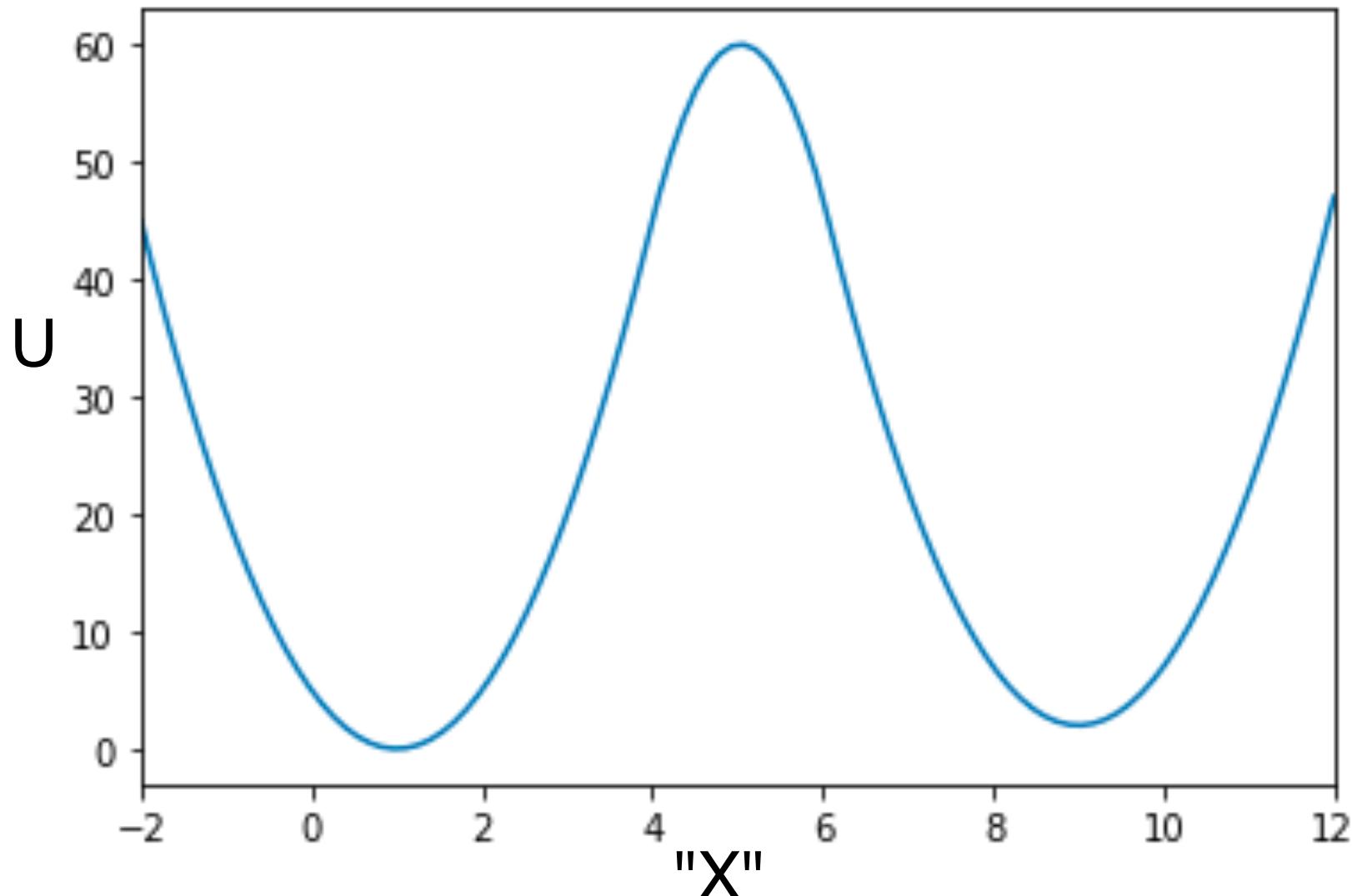
$$p_i(x) = e^{\beta A_i - \beta U_i(x)}$$

- Problem: the distribution looks like this:



- 10,000 atoms means it's a 30,000 dimensional problem
- Only a vanishingly small fraction of the x_i have high probability (low energy)

A toy problem (let's go to the notebook!)



Reminder: Markov Chain Monte Carlo

- We want a rule for flow between states, that preserves the probability AT the states

$$\frac{p_i}{p_j} = \frac{T_{j \rightarrow i}}{T_{i \rightarrow j}}$$

Probability we are in state i Chance of going from j to i

Probability we are in state j Chance of going from i to j

The diagram illustrates the detailed balance equation for a Markov chain. It shows the ratio $\frac{p_i}{p_j}$ equated to the ratio of transition probabilities $T_{j \rightarrow i}$ over $T_{i \rightarrow j}$. Four arrows point from descriptive labels to the corresponding terms in the equation:

- An arrow points from "Probability we are in state i" to p_i .
- An arrow points from "Chance of going from j to i" to $T_{j \rightarrow i}$.
- An arrow points from "Probability we are in state j" to p_j .
- An arrow points from "Chance of going from i to j" to $T_{i \rightarrow j}$.

- We keep the ratio of "stuff" in state i and j equal by making sure the ratio of stuff going IN and OUT is equal
- **Detailed balance**

Reminder: Markov Chain Monte Carlo

$$\frac{p(x_i)}{p(x_j)} = \frac{e^{\beta A - \beta U(x_i)}}{e^{\beta A - \beta U(x_j)}} = e^{\beta(U(x_j) - U(x_i))}$$

Any rule that satisfies this works!

$$\frac{T_{x_j \rightarrow x_i}}{T_{x_i \rightarrow x_j}} = e^{\beta(U(x_j) - U(x_i))}$$

A rule: Step 1: Propose a move randomly and isotropically in any direction

Step 2: If $U(x_j) < U(x_i)$, then move there

If $U(x_j) > U(x_i)$, then move with probability: $e^{\beta(U(x_j) - U(x_i))}$

Checking the rule

Step 1: Propose a move randomly and isotropically in any direction

This is really important, because otherwise, the change of proposing a move from i

$$T_{i \rightarrow j} = \alpha(i \rightarrow j)P(i \rightarrow j)$$

Chance of going from i to j Chance of proposing a move from i to j Chance of accepting a move from i to j

Step 2:

$$\frac{T_{j \rightarrow i} = \alpha(j \rightarrow i)P(j \rightarrow i)}{T_{i \rightarrow j} = \alpha(i \rightarrow j)P(i \rightarrow j)} = \frac{P(j \rightarrow i)}{P(i \rightarrow j)} = \frac{1}{e^{\beta(U(x_i) - U(x_j))}}$$

if $U(x_i) < U(x_j)$	if $U(x_i) > U(x_j)$
$\sqrt{e^{\beta(U(x_j) - U(x_i))}}$	$e^{\beta(U(x_j) - U(x_i))}$

Back to the notebook

Importance sampling can allow us to remove a bias

$$\begin{aligned}\langle O \rangle_i &= \int O(\vec{x}) p_i(\vec{x}) d\vec{x} = \int O(\vec{x}) p_i(\vec{x}) \frac{p_j(\vec{x})}{p_j(\vec{x})} d\vec{x} \\ &= \int O(\vec{x}) p_j(\vec{x}) \frac{p_i(\vec{x})}{p_j(\vec{x})} d\vec{x} \quad \langle O \rangle_i = \frac{1}{N} \sum_{n=1}^N O(\vec{x}_n) \frac{p_i(\vec{x}_n)}{p_j(\vec{x}_n)} \\ &\qquad\qquad\qquad \text{x}_n \text{ sampled from } p_j(x)\end{aligned}$$

Sampling from $p_j(x)$, observables from $p_i(x)$

If $p_i(x)/p_j(x)$ is similar in magnitude as x_n varies, then we can avoid generating the entire Monte Carlo chain

If $p_i(x)/p_j(x)$ varies too much, then it becomes very noisy!

How can we remove a bias?

$$\frac{p_i(x)}{p_j(x)} = \frac{e^{\beta A_i - \beta U_i(x)}}{e^{\beta A_j - \beta U_j(x)}} = e^{\beta \Delta A - \beta \Delta U(x)}$$

For physical problems, we know these distributions, and can do importance

$$\frac{p_i(x)}{p_j(x)} = \frac{e^{\beta A_i - \beta U_i(x)}}{e^{\beta A_j - \beta U_j(x)}} = e^{\beta \Delta A - \beta \Delta U(x)}$$

$$O(\vec{x}) = 1$$

$$\langle O \rangle_i = \frac{1}{N} \sum_{n=1}^N O(\vec{x}_n) e^{\beta \Delta A - \beta \Delta U(\vec{x}_n)}$$

$$e^{-\beta \Delta A} = \frac{1}{N} \sum_{n=1}^N e^{-\beta \Delta U_i(\vec{x}_n)}$$

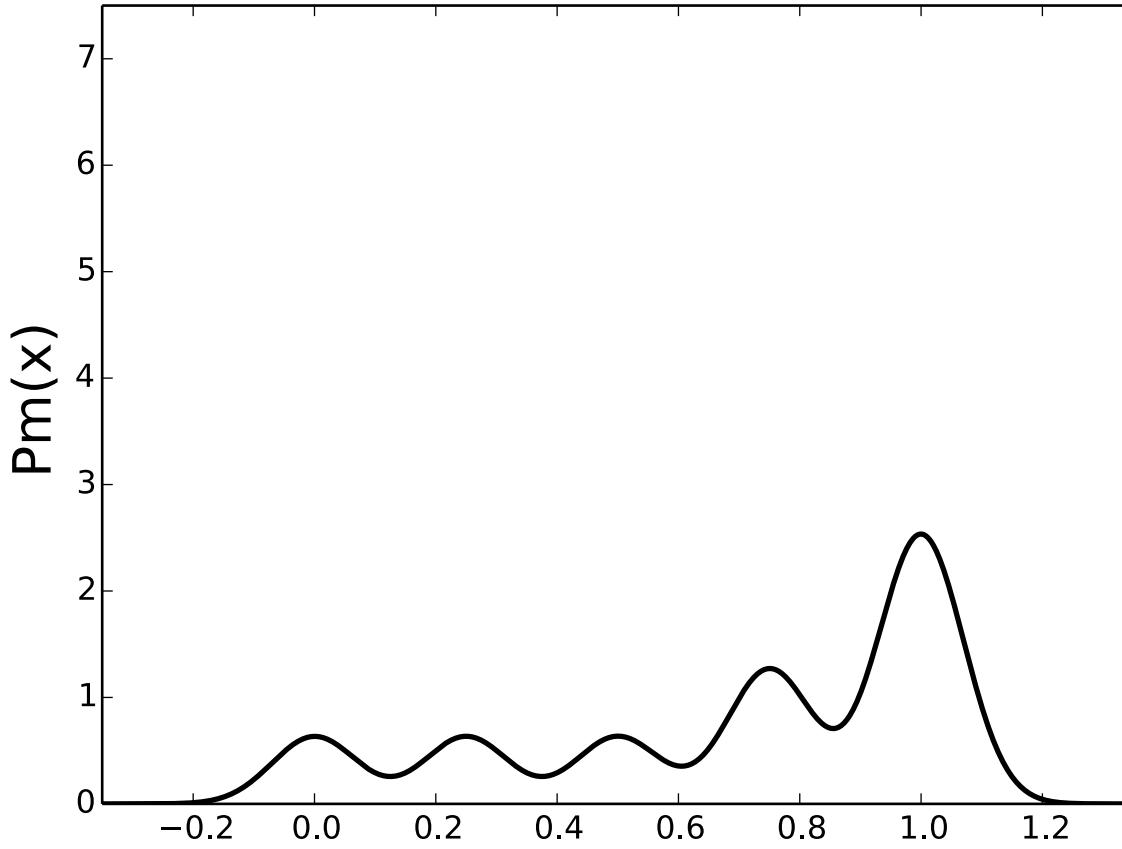
$$\langle O \rangle_i = \frac{\frac{1}{N} \sum_{n=1}^N O(\vec{x}_n) e^{-\beta U(\vec{x}_n)}}{\frac{1}{N} \sum_{n=1}^N e^{-\beta U(\vec{x}_n)}}$$

Zwanzig

How do we remove multiple biases?

- The next few slides will be cleaned up some more.

We can perform importance sampling from multiple probability distributions



Different numbers of samples?
 $N = [10, 10, 10, 20, 40]$

$$p_m(\vec{x}) = \frac{1}{\sum_{k=1}^5 N_k} \left[p_1(N_1)p_1(\vec{x}) + p_2(N_2)p_2(\vec{x}) + p_3(N_3)p_3(\vec{x}) + p_4(N_4)p_4(\vec{x}) + p_5(N_5)p_5(\vec{x}) \right]$$

$p_m(x)$ is called a ‘mixture distribution’

Importance sampling from the mixture distribution gives lowest variance estimate

$$e^{-\beta A_i} = \frac{1}{N} \sum_{n=1}^N \frac{-\beta U_i(\vec{x}_n)}{\sum_k \frac{N_k}{N} p_k(\vec{x}_n)} \quad \left| \quad \langle O \rangle_i = \frac{1}{N} \sum_{n=1}^N O(\vec{x}_n) \frac{e^{\beta A_i - \beta U_i(\vec{x}_n)}}{\sum_k \frac{N_k}{N} p_k(\vec{x}_n)} \right.$$

- Multistate Bennett Acceptance Ratio (MBAR)
 - Shirts and Chodera, *J. Chem. Phys.*, 129:124105 (2008)
- Reduces to:
 - Bennett acceptance ratio (bridge sampling) for two states
 - Weighted histogram analysis method (WHAM) in the limit of zero-width histogram bins
- Is the maximum likelihood estimate of all data collected
- Provably the lowest variance reweighting estimator
- DOES require calculating the ratio of probabilities (i.e. potential energy differences) between all states

pymbar: A Python implementation of reweighting

- Fast, robust, and efficient
- Includes many examples
 - Potentials of mean force
 - Solvation free energies
 - Force bias single molecule experiments
- Automated setup.py and installation through conda
- <https://github.com/choderalab/pymbar>