Free energies along reaction coordinates & biased sampling ChE210D

> **Today's lecture:** biased ensembles in simulations that facilitate sampling along reaction coordinates, and general methods for extracting free energies along these coordinates

Overview of free energies and biased sampling

Free energies drive many important processes and are one of the most challenging kinds of quantities to compute in simulation. Free energies involve sampling at constant temperature, and ultimately are tied to summations involving partition functions. In order to measure free energies, generally there are two requirements:

- We must be able to **sample** the relevant molecular states along the direction that we would like to calculate a free energy. Often we need to introduce a **bias** to facilitate sampling.
- We must be able to relate the **frequency** of different molecular states to the underlying free energy of interest.

Thus we see that sampling (and biased sampling) is intimately related to the measurement of free energies. There are many kinds of free energies that we might compute, but here we first consider potentials of mean force.

Potentials of mean force (PMFs)

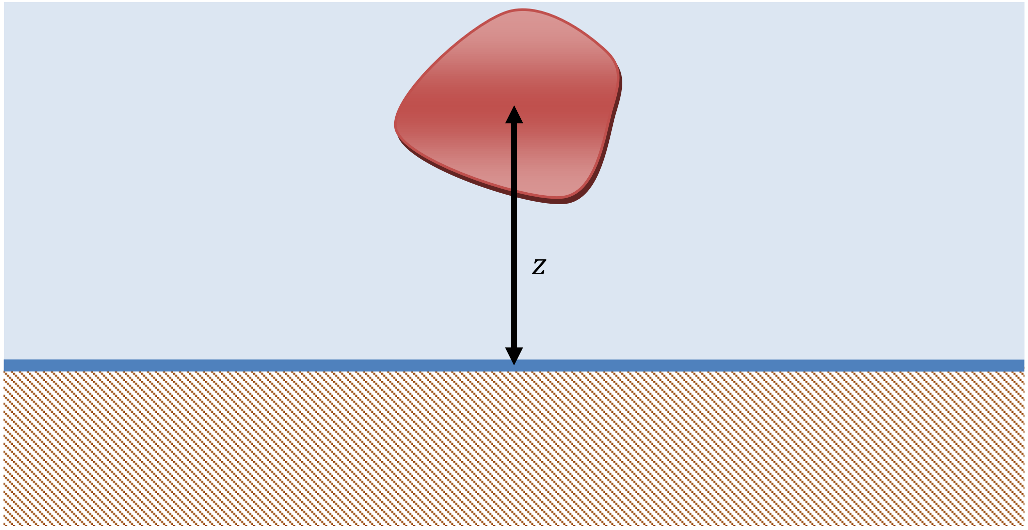
Oftentimes we would like to compute the free energy along some order parameter or reaction coordinate of interest. These are broadly termed **potentials of mean force**, for reasons we will see shortly. This perspective enables us to understand free-energetic driving forces in many processes. For the purposes of this discussion, we will notate a PMF by

$$F(\xi)$$

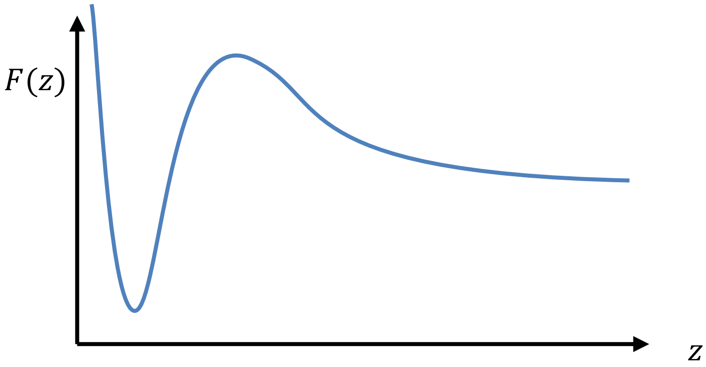
where ξ is the reaction coordinate of interest. This coordinate might be, for example:

- an intra- or intermolecular distance (or combination of distances)
- a bond or torsion angle
- a structural order parameter (e.g., degree of crystallinity, number of hydrogen bonds)

Let's motivate things using the example of a protein in aqueous solution interacting with a surface. The reaction coordinate might be the distance between the center of mass of the protein and the surface z:



The PMF along z, $F(z)$ would give the free energy of the system as a function of the protein-surface distance. It might look something like:



Remember that free energies dictate frequencies of different system states at equilibrium. Specifically, they relate to the probability of observing a particular molecular state:

$$\wp(z) \propto e^{-\beta F(z)}$$

Thus this curve would show us:

- the preferred distance at which the protein binds to the surface, from the value of z at the free energy minimum

- the free energy change upon binding, from the difference in free energy between the minimum and large values of z

- the barrier in free energy for binding and unbinding, from the height of the hump

Importantly, the free energy function does *not* just include the direct potential energy interactions between atoms in the molecule with atoms in the surface. It also includes the effects of all of the interactions in the solvent molecules. This may be crucial to the behavior of the system.

For example, the direct pairwise interactions of an alkane with a silica surface will be the same regardless of whether the solvent is water or octanol. However, the net interaction of the alkane and surface will be very different in the two cases due to solvent energies and entropies, and this effect is exactly determined by the PMF.

Calculating the PMF

We can gain insight into the PMF by recognizing the connection between free energies and probabilities. Specifically, the PMF determines the relative probability of observing the protein at a distance z through a Boltzmann factor:

$$\wp(z) \propto e^{-\beta F(z)}$$

$$\wp(z) \rightarrow \frac{e^{-\beta F(z)}}{\int e^{-\beta F(z)} dz} = \frac{e^{-\beta F(z)}}{Z}$$

Inverting we then have

$$F(z) = -k_B T \ln \wp(z) - k_B T \ln Z = -k_B T \ln \wp(z) + z\text{-independent constant}$$

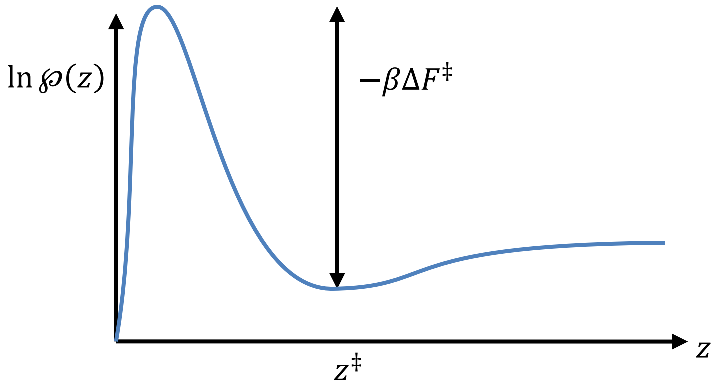
Thus if we were able to measure $\wp(z)$ to high accuracy, we would be able to determine the free energy along the coordinate z. In practice, we represent $\wp(z)$ as a histogram of counts of observations in our simulation, $H(z)$, such that

$$\wp(z) \approx \frac{H(z)}{\sum H(z)} = \frac{H(z)}{n_{\text{obs}}}$$

Then we have

$$\boxed{F(z) \approx -k_B T \ln H(z) + z\text{-independent constant}}$$

But the real challenge in simulation is measuring $F(z)$ to high accuracy. This is intimately connected to the form of $F(z)$ itself. Consider the following. If the solute interacts strongly and favorably with the surface as shown above, we might expect a probability distribution of the coordinate z to look something like:



If the free energy barrier for a surface-bound solute for escaping is fairly large, we will expect an extremely small probability for the system to adopt states other than the bound state. That is, we would almost never see a solute molecule far from the surface in the simulation. The molecule would spend all of its time near the surface it would be very difficult to measure $\wp(z)$ to good accuracy in low-probability regions. That is, the statistics of our estimate will be very poor:

$$H(z) = 0 \text{ for many z } \rightarrow \quad F(z) = \text{ill defined for those } z$$

Consider the following: we make 10,000 measurements of z in our simulation. If the free energy at the barrier is $\beta\Delta F^\ddagger = 10$, then the relative number of counts between the bound state and at $z^\ddagger$ is $\exp(10) = 22{,}000$. In this case we wouldn't expect any counts in histogram bins near the barrier!

Umbrella sampling and reweighting

How then do we ensure good sampling across the z coordinate in our simulations? The solution is to bias the sampling, and one of the most common and simplest approaches to doing so is the umbrella sampling approach.

Umbrella sampling

The **umbrella sampling** method was developed by Torrie and Valleau in 1977 and since has been one of the major approaches for performing simulations along predetermined reaction coordinates. It is a form of **biased sampling** that helps overcome the basic challenge above. It is readily applied to both molecular dynamics and Monte Carlo simulations. Here, we use the protein-surface for demonstrating the application of the approach; however, keep in mind that umbrella sampling can be applied to many, arbitrary reaction coordinates.

For the present example, we want to force the system to explore the entire range of z. Therefore, we construct multiple simulations where the z-coordinate is restrained to different regimes. The exact procedure is:

- Perform J simulations j of the same system.

- In each simulation, **restrain** the system to sample a small range of z values centered around z_j using (typically) a harmonic penalty or "rubber band":

  $$U_{\mathrm{j}}(\mathbf{r}^N) = U(\mathbf{r}^N) + \frac{k}{2}\left(z - z_j\right)^2$$

  where the bias is a form of energy weights that restrain the position,

  $$\eta_j(z) = \frac{k}{2}\left(z - z_j\right)^2$$

- Use a different target value z_j for each simulation. The z_j should span the entire range of interest.

- Measure the biased ensemble distribution $\wp_j(z)$ for each simulation using histograms. Let $c_j(z)$ denote the counts in simulation j for different z observations.

- Unweight and stitch together all of the simulation $\wp_j(z)$ to produce the true underlying free energy function $F(z) = -k_B T \ln \wp(z)$.

Form of the biasing potential

In each simulation, we need to bias the run to sample values of z near z_j. We want to energetically penalize configurations outside of this range so as to lower their probability weight. Many forms of this penalty can be chosen. The most common is a harmonic potential:

$$\eta_j(z) = -\beta\frac{k}{2}\left(z - z_j\right)^2$$

such that the effective weighted potential is given by

$$U_{\mathrm{j}}^{\mathrm{w}}(\mathbf{r}^N) = U(\mathbf{r}^N) + \frac{k}{2}\left(z - z_j\right)^2$$

Here a value of the force constant must be specified. Too small a value will not sufficiently bias the simulation. Too large a value will result in a very narrow distribution in the sampled z in each simulation; this will result in poor overlap between the z distributions in each simulation and will make hard to patch the unweighted results together with good statistical accuracy. In practice, it can be a trial-and-error process to determine good values for k.

The advantage of the harmonic potential is that it enables us to apply either MD or MC algorithms to measure the *biased* or *weighted* histogram $\wp_j^{\mathrm{w}}(z)$. In MD, the harmonic term will add an extra force to the solute in its z-coordinate. In MC, this term will modify the energy used in the acceptance criterion.

Reweighting concepts

Even though we run simulations with biases that hold the molecule near individual points in space, it is possible to recover the true (unbiased) distribution using **reweighting** concepts. Each simulation is connected to the unweighted distribution via the following considerations:

$$\wp_j(\mathbf{r}^N) \propto e^{-\beta U(\mathbf{r})-\beta\eta_j(z)}$$

such that

$$\wp_j^{\mathrm{w}}(\mathbf{r}^N) \propto \wp(\mathbf{r}^N)e^{-\beta\eta_j(z)}$$

or inverting

$$\wp(\mathbf{r}^N) \propto \wp_j^{\mathrm{w}}(\mathbf{r}^N)e^{\beta\eta_j(z)}$$

Integrating out all degrees of freedom except z,

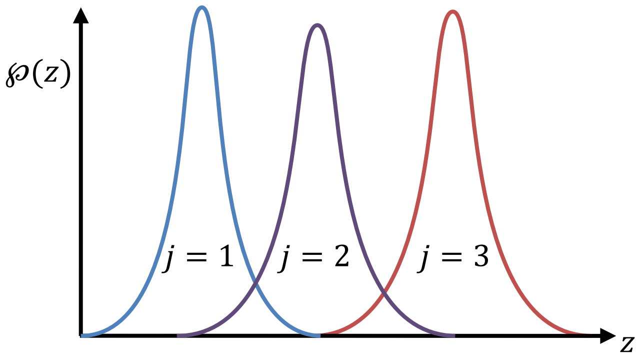
$$\wp(z) \propto \wp_j^{\mathrm{w}}(z)e^{\beta\eta_j(z)}$$

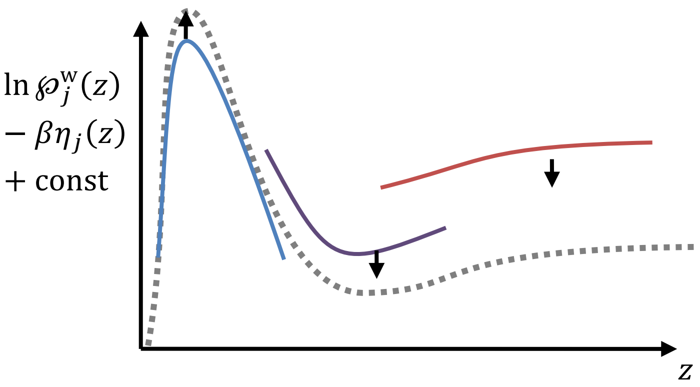
Taking the logarithm,

$$\boxed{F(z) = -k_B T \ln \wp_j^{\mathrm{w}}(z) - \eta_j(z) + \mathrm{const_j}}$$

This provides a way to measure the free energy, again by approximating each $\wp_j^{\mathrm{w}}(z)$ by a histogram. Notice that each simulation j should return, in principle, the same distribution $\wp(z)$. Also note that the constant term will be different for each umbrella.

We have forced the different distributions to span a broad range of z solely to attain accurate statistics over the full range of z. We could stitch together the J estimates of $F(z)$ by shifting the unknown constants so as to obtain overlap in the common regions:





We need to have sufficient overlap between the distributions in each simulation in order to do this effectively. Clearly a more systematic approach would be desirable, since the error in the curves are likely largest where they overlap – in the tails of the probability distributions.

Statistically-optimal multiple histogram reweighting

Rather than visually shift these curves to overlap, we can resort to statistical methods. Ferrenberg and Swendsen in 1989 proposed an optimal way to stitch together different histograms by minimizing the statistical error in the computed free energies. It was later generalized by [Kumar et al., 1992] and was named the **Weighted Histogram Analysis Method (WHAM)**. The **so-called Multistate Bennet Acceptance Ratio** (**MBAR**) approach of Shirts and Chodera in 2008 is a more statistically rigorous approach to the same calculation, and their pymbar Python implementation is now community standard for automating the calculations.

Generally the best way to extract a free energy is through an approach called **multiple histogram reweighting.** Here we take a **maximum likelihood approach** to deriving the main results. Maximum likelihood estimation is a statistical method for parameterizing probability models that can be shown to have the lowest error versus any other approach in the limit of large sample sizes. It simply says the following: posit some form of the underlying distribution function for a random process. Then, given an observed set of events, the best parameters for that distribution are those that maximize the probability (likelihood) of the observed events.

Say that we make n *independent* observations of the distance z in each umbrella simulation j. In order to make a connection with the histogram, we will discretize all of z space into discrete values z_k separated by intervals δz. Before we start, let's define some notation:

- n – number of observations in each umbrella simulation
- i – index of observations ($i = 1, \dots, n$)
- j – index of umbrella ($j = 1, \dots, J$)
- k – index of discrete distance values
- F_k – discrete free energy values at each distance
- η_{jk} – bias in umbrella j at distance k
- z_{ij} – ith observation of z in umbrella j
- c_{jk}– count of histogram entries for umbrella j and distance bin k

With these notations, we construct the total probability or likelihood L of making the nJ observations from the different umbrella simulations in terms of a yet-unknown unknown density of states function:

$$\begin{aligned} L &= \prod_{i=1}^{n} \prod_{j=1}^{J} \wp_j(z_{ij}) \\ &= \prod_{i=1}^{n} \prod_{j=1}^{J} \left(\frac{e^{-\beta F(z_{ij}) - \beta \eta_j(z_{ij})}}{Z_j} \right) \end{aligned}$$

Here, the Z_j can be seen as the probability normalization constants and are formally given by

$$Z_j = e^{-\beta A_j} = \int e^{-\beta F(z) - \beta \eta_j(z)} dz$$

$$\boxed{Z_j = e^{-\beta A_j} = \sum_k e^{-\beta F_k - \beta \eta_{jk}}}$$

where the A_j are the free energies of each umbrella.

With all of these considerations, we can rewrite the above probability in terms of histograms:

$$L = \prod_{k} \prod_{j=1}^{J} \left(\frac{e^{-\beta F_k - \beta \eta_{jk}}}{Z_j} \right)^{c_{jk}}$$

Note that the sum over observations was converted to a sum over discrete distance indices, with the c_{jk} accounting for the number of observations at each distance. Note also:

- F_k → (unbiased) free energy at discrete distance k
- η_{jk} → bias for umbrella j at distance k

Taking the logarithm,

$$\ln L = \sum_{k} \sum_{j=1}^{J} c_{jk} \left[-\beta F_k - \beta \eta_{jk} + \beta A_j \right]$$

According to the maximum likelihood approach, we want to maximize the likelihood of the observations with respect to any adjustable parameters in our probability model. The observations are the histogram counts c_{jk} and the parameters are the yet unknown values F_k. (Note that the A_j values are not independent, because they can be computed from the F_k and η_{jk}.) Therefore, we take the derivative with respect to F_k and set it equal to zero:

$$0 = \frac{\partial \ln L}{\partial F_k} \quad \text{for all } k$$

Evaluating,

$$0 = \sum_{j=1}^{J} \left(-\beta c_{jk} + n\beta \frac{\partial A_j}{\partial F_k} \right)$$

The latter derivative is

$$\begin{aligned} \beta \frac{\partial A_j}{\partial F_k} &= -\frac{\partial \ln Z_j}{\partial F_k} \\ &= -\frac{1}{Z} \frac{\partial Z_j}{\partial F_k} \\ &= \beta \frac{e^{-\beta F_k - \beta \eta_{jk}}}{Z_j} \\ &= \beta e^{-\beta F_k - \beta \eta_{jk} + \beta A_j} \end{aligned}$$

Substituting back in:

$$0 = \sum_{j=1}^{J} \left(c_{jk} - n e^{-\beta F_k - \beta \eta_{jk} + \beta A_j} \right) \quad \text{for all } k$$

This can be solved for F_k:

$$e^{\beta F_k} = \left[\sum_{j=1}^{J} c_{jk}\right]\left[n\sum_{j=1}^{J} e^{-\beta\eta_{jk}+\beta A_j}\right]^{-1}$$

Notice that this equation now provides us with a recipe for computing the free energy (the LHS) based on histogram data. We can simplify slightly:

$$e^{\beta F_k} = \frac{c_{tot,k}}{n}\left[\sum_{j=1}^{J} e^{-\beta\eta_{jk}+\beta A_j}\right]^{-1}$$

Here, we have made the definition

$$c_{tot,k} = \sum_{j=1}^{J} c_{jk}$$

such that it gives the *total* number of histogram counts at distance z_k, in any umbrella

Even though this expression gives us a way to determine the PMF from multiple umbrella simulation results, note that the RHS involves the free energy at every temperature. This free energy depends on F_k:

$$A_j = -\beta^{-1}\ln\sum_{k} e^{-\beta F_k-\beta\eta_{jk}}$$

Iterative reweighting solution

How can we solve for the discrete PMF free energy values F_k? Ferrenberg and Swendsen suggested an iterative solution:

1. An initial guess is made for the J values of A_j. One can simply choose $A_j = 0$ for all j.

2. The (discretized) free energy at every distance is computed using

$$\boxed{\beta F_k = \ln c_{tot,k} - \ln n - \ln\sum_{j=1}^{J} e^{-\beta\eta_{jk}+\beta A_j} \qquad \text{for all } k}$$

3. The free energies are recalculated using

$$\boxed{-\beta A_j = \ln\sum_{k} e^{-\beta F_k-\beta\eta_{jk}} \qquad \text{for all } j}$$

4. Steps 2 and 3 are repeated until the free energies no longer change upon each iteration. In practice, one typically checks to see if the free energies change less than some fractional tolerance.

5. According to the above equations, the function F_k and the values A_j can only be determined to within an additive constant. Thus, with each iteration, one typically sets one of the free energies equal to zero in one umbrella, such as $A_{j=1} = 0$, and similarly $F_{k=1} = 0$.

Thus, the combination of umbrella sampling and statistical reweighting allows us to:

- Explicitly force exploration of the full reaction coordinate through biasing potentials.

- Rigorously remove the effect of the biases and compute the "true" PMF to high accuracy, in a statistically optimal way. That free energy function $F(z)$ is approximated by the discrete-distance values F_k.

Property averages through configurational weights

What if we want to calculate other observables in the system, such as the protein orientation or hydration level as a function of z? We can easily do by **reweighting** the umbrella simulations, in effect removing the bias post-simulation. We define a **weight** associated with each configuration i in the trajectory at each umbrella j:

$$w_{ij} \propto e^{-\beta F(z_{ij}) - \beta \eta_j(z_{ij})}$$

In practice, the logarithm of the weights are maintained in the computer rather than the weights themselves, so as to maintain precision.

Any property depending on the configurational coordinates can be determined by a sum over *all* of the nJ observations using the weights:

$$\langle X \rangle = \frac{\sum_{j=1}^{J} \sum_{i=1}^{n} w_{ij} X_{ij}}{\sum_{j=1}^{J} \sum_{i=1}^{n} w_{ij}}$$

Here, X_{ij} is the value of the property X for the ith configuration observed in umbrella j. For example, one can compute the mean potential energy,

$$\langle U \rangle = \frac{\sum_{j=1}^{J} \sum_{i=1}^{n} w_{ij} U_{ij}}{\sum_{j=1}^{J} \sum_{i=1}^{n} w_{ij}}$$

Or we could find the mean potential energy at a given distance by restricting the sums to those distances of interest.

$$\langle U \rangle_z = \frac{\sum_{j=1}^{J}\sum_{i=1}^{n} w_{ij} U_{ij}\, \delta_{z,z_{ij}}}{\sum_{j=1}^{J}\sum_{i=1}^{n} w_{ij} \delta_{z,z_{ij}}}$$

However, keep in mind that X can be *any* property of interest. Notice that the reweighting approach uses *all* of the data, from *each* umbrella simulation. Thus, it provides an incredibly general way to determine averages in a statistically *optimal* manner.

Formal details of reweighting

Histograms in simulations

Histograms are concerned with computing property **fluctuations,** and as we saw can be tied to free energies (and also entropies). Let's talk a bit about how we build a histogram. For the purposes of illustration, consider a histogram in distance per the protein-surface example. In our simulation, we would measure the distribution of the variable z using a long simulation run and many **observations** of the instantaneous value of z.

The distance is a continuously-varying variable. Therefore, the underlying $\wp(z)$ is a continuous probability distribution. However, in the computer we must measure a discretized version of this distribution.

- We specify a minimum and maximum value of the energy that defines a **range** of distances in which we are interested. Let these be $z_{\min}$ and $z_{\max}$.

- We define a set of m **bins** into which the energy range is discretized**.** Each bin has a **bin width** of

$$\delta z = \frac{z_{\mathrm{max}} - z_{\mathrm{min}}}{m}$$

- Let the variable k be the **bin index**. It varies from 0 to $m-1$. The average distance of bin k is then given by

$$z_k = z_{\mathrm{min}} + \left(k + \frac{1}{2}\right)\delta z$$

- We create a **histogram** along the distance bins. This is simply an array in the computer that measures **counts** of observations:

$$c_k = \text{counts of } z \text{ observations in the range } [z_k - \delta z/2\,, z_k + \delta z/2)$$

To find the bin to which a given z value belongs, we use

$$k = \mathrm{int}\left(\frac{z - z_{\min}}{\delta z}\right)$$

Here the int function returns the integer part of its argument. For example, $\mathrm{int}(2.6) = 2$. For simplicity, we often write the histogram array using the distance, rather than the bin index,

$$c(z) \to c_k \qquad \text{where } k = \mathrm{int}\left(\frac{z - z_{\min}}{\delta z}\right)$$

To create a histogram in simulation, we perform a very large number of observations n from a long, equilibrated molecular simulation. At each observation, we update:

$$c(z) \leftarrow c(z) + 1$$

This update is only performed if $z_{\min} \leq z < z_{\max}$. Otherwise, the distance would be outside of the finite range of interest. However, we still need to keep count of all distances, whether or not inside the range, in order to properly normalize our histogram.

We can **normalize** to determine a discretized approximation to the true underlying *continuous* distribution $\wp(z)$:

$$\tilde{\wp}(z)\delta z = \frac{c(z)}{n}$$

where n is the *total number of observations,* including those outside of the defined range. On the LHS we include the bin width so as to approximate the *continuous* probability differential $\wp(z)dz$. Thus,

$$\tilde{\wp}(z) = \frac{c(z)}{n\delta z}$$

In the limit of an infinite number of observations from an infinitely long, equilibrated simulation, this approximation converges to the true one in the following manner:

$$\tilde{\wp}(z_k)\delta z = \int_{z_k - \delta z/2}^{z_k + \delta z/2} \wp(z) dz$$

This equation simply says that we sum up all of the underlying probabilities for the continuous energies within a bin to obtain the observed, computed probabilities. As the bin width goes to zero, we have

$$\lim_{\delta z \to 0, n \to \infty} \tilde{\wp}(z) = \wp(z)$$

Notice that there are two components to this limit:

- We need an infinite number of observations.
- We need an infinite number of bins.

These two limits "compete" with each other: as we increase the number of bins, we need more observations so that we have enough counts in each bin to have good statistical accuracy. Practically speaking, we must choose a finite bin width that enables us to balance the length of the run with statistical accuracy in each bin. Typically, for the example above,

- The bin width is chosen to be of the order of a characteristic distance. We might choose $1\ \text{Å}$. For a Lennard-Jones system, we might choose 0.1σ.
- The simulation is performed long enough to achieve on the order of ~1000 or more average counts per bin, that is, $n \geq m \times 1000$.

Statistical considerations

Keep in mind that the computation of a histogram is subject to the same statistical considerations as simple simulation averages. That is, the histogram needs to be performed for many correlation times to reach good statistical accuracy. It can be shown that the expected squared error in the histogram bin $c(z)$ goes as

$$\sigma_{c(z)}^2 \propto c(z)$$

This implies that the error goes as the square root as the number of counts. Similarly, the expected squared error in the corresponding estimate $\tilde{\wp}(z)$ goes as:

$$\sigma_{\tilde{\wp}(z)}^2 \propto \frac{\wp(z)}{n}$$

The relative error in $\tilde{\wp}(z)$ is given by:

$$\frac{\sigma_{\tilde{\wp}(z)}}{\tilde{\wp}(z)} \propto \frac{1}{\sqrt{n\wp(z)}}$$

Notice that the relative error is higher for lower values of $\wp(z)$, i.e., at the tails of the distribution.

Multidimensional histograms

In this example, we considered only a histogram of distance. However, it is possible to construct histograms of many kinds of simulation observables. We could, for example, measure a histogram of both distance and an angle characterizing orientation of the protein, or distance and the number of waters in the protein's hydration shell.

We compute **joint distributions** using **multidimensional** histogram arrays. For example,

$$\tilde{\wp}(z,\theta)\delta z\delta\theta = \frac{c(z,\theta)}{n}$$

Note that any continuous variables will require discretization and specification of a bin width. Discrete variables, on the other hand, do not require such a definition because the underlying distribution itself is discrete. A histogram for the number of hydration waters would follow:

$$\tilde{\wp}(N_w) = \frac{c(N_w)}{n}$$

Moreover, many kinds of distributions require the specification of a minimum and maximum observable value.

Connection to statistical mechanics

The power of histograms is that they allow us to measure fluctuations in the simulation that can be used to extract underlying partition functions. That is, we measure $\wp(z)$ in simulation and then we post-process this discretized function to make connections to free energies and entropies.

The rigorous form of the distribution $\wp(z)$ is given by:

$$\wp(z) = \frac{\int e^{-\beta U}\delta[z - \hat{z}(\mathbf{r}^N)]d\mathbf{r}^N}{Z(T,V,N)} = \frac{e^{-\beta F(z)}}{Z(T,V,N)}$$

where $Z(T,V,N)$ is the configurational partition function,

$$Z(T,V,N) = \int e^{-\beta U}d\mathbf{r}^N = \int e^{-\beta F(z)}dz$$

Potentials of mean force (PMFs)

Formally, a potential of mean force (the free energy) along some reaction coordinate ξ is given by a partial integration of the partition function. In the canonical ensemble, we begin with the configurational part of the Helmholtz free energy,

$$\begin{aligned} F(\xi) &= A(T,V,N,\xi) \\ &= -k_B T \ln Z(T,V,N,\xi) \\ &= -k_B T \ln \int e^{-\beta U(\mathbf{r}^N)}\delta[\xi - \hat{\xi}(\mathbf{r}^N)]d\mathbf{r}^N \end{aligned}$$

Here, $\hat{\xi}(\mathbf{r}^N)$ is a function that returns the value of the order parameter for a particular configuration $\mathbf{r}^N$. The integral in this expression entails a delta function that filters for only those Boltzmann factors for configurations with the specified ξ.

One can think of the PMF as the free energy when the system is constrained to a given value of ξ. Notice that we have the identity

$$\int e^{-\beta F(\xi)} d\xi = e^{-\beta A}$$

The potential of mean force is so-named because its derivative gives the average force along the direction of ξ at equilibrium. We proceed to find the derivative of the PMF:

$$\begin{aligned} \frac{dF(\xi)}{d\xi} &= -k_B T \frac{d}{d\xi} \ln \int e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N \\ &= -k_B T \frac{\frac{d}{d\xi} \int e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N}{\int e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N} \end{aligned}$$

To make progress, we need the mathematical identity

$$\frac{d}{da} \int g(x) \delta(x-a) dx = \int \frac{dg(x)}{dx} \delta(x-a) dx$$

This allows us to pull the derivative inside the integral:

$$\begin{aligned} \frac{dF(\xi)}{d\xi} &= -k_B T \frac{\int \left(\frac{d}{d\xi} e^{-\beta U(\mathbf{r}^N)}\right) \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N}{\int e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N} \\ &= -k_B T \frac{\int \left(-\beta \frac{dU}{d\xi} e^{-\beta U(\mathbf{r}^N)}\right) \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N}{\int e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N} \\ &= -\frac{\int f_\xi e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N}{\int e^{-\beta U(\mathbf{r}^N)} \delta\left[\xi - \hat{\xi}(\mathbf{r}^N)\right] d\mathbf{r}^N} \end{aligned}$$

Here, the term f_ξ gives the force along the direction of ξ,

$$\begin{aligned} f_\xi &= -\frac{dU}{d\xi} \\ &= -\frac{d\mathbf{r}^N}{d\hat{\xi}} \cdot \nabla U \\ &= \frac{d\mathbf{r}^N}{d\hat{\xi}} \cdot \mathbf{f}^N \end{aligned}$$

The remainder of the terms in the PMF equation serve to average the force for a specified value of ξ. Thus,

$$\frac{dF(\xi)}{d\xi} = -\langle f_\xi(\xi) \rangle$$

Theory of biased ensembles and reweighting

When we perform a simulation with an artificial bias, we are simulating in **biased ensembles.** Such ensembles have some benefits:

- We can achieve broader sampling of particular states that are rarely visited, in order to compute properties of them. The usual ensembles give rise to sharply peaked distribution functions whose relative width scales as $N^{-1/2}$; biased ensembles allow us to broaden these fluctuations.

- We can achieve **enhanced sampling** of states in a way that promotes faster equilibration times.

- By designing artificial ensembles that visit frequently states that are normally rarely-sampled, we can compute free energies of them to much higher accuracy than would normally be the case. We saw this was particularly useful with umbrella sampling

Ultimately, reality exists in standard ensembles. Therefore, any simulation of a system that we perform in a biased ensemble must somehow be related back to results for that system in usual ensembles. To do this, we can use **reweighting** techniques.

Keep in mind that the specification of the ensemble is entirely separate from the specification of the system evolution. We can use molecular dynamics or Monte Carlo to perform the sampling. Here, however, we will consider the case of a MC simulation specifically.

General formalism and connection to statistical mechanics

For the sake of simplicity, we will use a discrete notation to indicate the probability distribution, as we did in our initial discussion of MC methods:

$$\wp_m$$

where m denotes one configurational microstate of the system, e.g., a set of coordinates $\mathbf{r}^N$.

Consider a MC simulation in the canonical ensemble, where we specify

$$\wp_m \propto e^{-\beta U_m}$$

We now want to perform a *biased* simulation in this ensemble where we introduce a **weighting factor** for the microstates in the exponential:

$$\boxed{\wp_m^{\mathrm{w}} \propto e^{-\beta U_m + \eta_m}}$$

The weighting function η_m modifies the canonical distribution so that some configurations have higher or lower probabilities than would normally be expected. It gives a value that is dependent on the particular configuration of interest. This function can take a variety of forms.

We can think of the weighting function as giving rise to an effective "weighted" potential energy function:

$$e^{-\beta U_m + \eta_m} \equiv e^{-\beta U_m^{\mathrm{w}}}$$

or,

$$U^{\mathrm{w}} \equiv U - k_B T \eta$$

Notice that if $\eta_m = \mathrm{const}$, then we recover the usual canonical distribution because its effect is removed via the normalization condition:

$$\wp_m^{\mathrm{w}} = \frac{e^{-\beta U_m + \eta_m}}{\sum_{m'} e^{-\beta U_{m'} + \eta_{m'}}}$$

Comparing the weighted and unweighted ensembles,

$$\boxed{\frac{\wp_m}{\wp_m^{\mathrm{w}}} \propto e^{-\eta_m}}$$

Imagine that we perform a MC simulation in the weighted ensemble. For symmetric moves, our acceptance criterion would follow

$$\begin{aligned} \frac{P_{12}^{\mathrm{acc}}}{P_{21}^{\mathrm{acc}}} &= \frac{\wp_2^{\mathrm{w}}}{\wp_1^{\mathrm{w}}} \\ &= e^{-\beta(U_2 - U_1) + \eta_2 - \eta_1} \\ &= e^{-\beta \Delta U + \Delta \eta} \end{aligned}$$

With the Metropolis form,

$$P_{12}^{\mathrm{acc}} = \min\left[1, e^{-\beta \Delta U + \Delta \eta}\right]$$

Ultimately, the MC simulation would generate a trajectory of configurations according to the distribution $\wp_m^{\mathrm{w}}$. We could compute the average of any property X in the weighted ensemble,

$$\langle X \rangle_{\mathrm{w}} = \frac{1}{n} \sum X_i$$

where the index $i = 1, \dots, n$ is an index over the trajectory configurations.

What if we wanted to compute the average in the *unweighted* ensemble (the usual canonical average)? We could still perform a simulation in the weighted ensemble, but our expression for the unweighted average would need to take into account differences in $\wp_m^{\mathrm{w}}$ and $\wp_m$:

$$\langle X \rangle_{\mathrm{u}} = \frac{\sum X_i \frac{\wp_i}{\wp_i^{\mathrm{w}}}}{\sum \frac{\wp_i}{\wp_i^{\mathrm{w}}}}$$

Substituting from above,

$$\boxed{\langle X \rangle_{\mathrm{u}} = \frac{\sum X_i \, e^{-\eta_i}}{\sum e^{-\eta_i}}}$$

Here, η_i gives the value of the weighting function for each configuration included in the trajectory i. The denominator in this expression serves as a normalization for the unweighting expression.

Practical and statistical issues

To perform such an unweighting of weighted simulation results, we need keep lists of the values A_i and η_i for each of the trajectory configurations. We also must rearrange the exponential so that we don't run out of precision:

$$\langle X \rangle_{\mathrm{u}} = \frac{\sum X_i \, e^{-\eta_i + \eta_{\min}}}{\sum e^{-\eta_i + \eta_{\min}}} \qquad \eta_{\min} = \min_i \eta_i$$

Keep in mind that the error in this expression is determined by the correlation times of the observable in the weighted ensemble.

The error is also highly dependent on whether the distribution of X in the unweighted ensemble is similar to the distribution in the weighted ensemble. If $\langle X \rangle_{\mathrm{u}}$ is rarely sampled in the weighted simulation, then the statistics of this unweighting procedure will produce a poor estimate of the unweighted average.

Typically, however, one explicitly selects a weighting function that allows the simulation to explore a very wide range of X to accumulate good statistics, much wider than runs in conventional ensembles. With multiple histogram reweighting, we run many different simulations and perform the reweighting over all of them at once, like we did with umbrella sampling.