Today's lecture: various kinds of Monte Carlo moves and how they are designed to facilitate otherwise challenging simulations

Considerations for Monte Carlo moves

Thermodynamics versus convergence kinetics

One advantage of MC simulations is that they permit great flexibility in the design and implementation of different MC move sets. There are in fact many different kinds of Monte Carlo moves. For every move, one chooses an acceptance criterion such that the desired ensemble probabilities $\wp_m$ are achieved. In other words, the choice of a move does not affect the ultimate equilibrium ensemble. Rather, it affects the rate, in MC steps, at which the simulation can converge to a stationary distribution. Thus, different MC moves affect the "kinetics" of a MC simulation, but not the thermodynamics.

In mathematical terms, the transition probabilities π_{mn} and π_{nm} can both either be large or small, resulting in frequent or infrequent transitions, respectively. For either case, their ratio can remain constant so as to produce the same equilibrium distribution:

$$\frac{\pi_{mn}}{\pi_{nm}} = \frac{\wp_n}{\wp_m}$$

Ultimately the choice of a move is motivated by the desire to have good convergence behavior in the simulation:

- fast equilibration times
- fast relaxation times and correlation times (so that shorter runs are then required to produce averages of good statistical quality)

Move considerations

To achieve these properties, one wants a simulation that moves very fast through configuration space, i.e., large π_{mn} values for $m \neq n$ and small π_{mm} values. This translates to MC moves that both have a high probability of acceptance *and* move the system far in configuration space. The efficacy of a MC move or set of moves can be assessed by computing correlation times.

There is a rigorous mathematical way to treat of convergence rates of Markov chains. If we were able to form the matrix π_{mn} for all pairs of states m and n, there we would be able to compute the rate of convergence of the MC simulation by finding the eigenvalues of the transition probability matrix. One of these eigenvalues will be one, corresponding to the stationary distribution. The others will determine how fast the simulation converges. Generally speaking, we want these other eigenvalues to be as small as possible for fast convergence. The choice of MC moves will affect the π_{mn} matrix and hence these eigenvalues.

Many systems are difficult to **sample**, meaning it is challenging to explore ergodically their configuration spaces. Such systems emerge when there are rare, low-energy configurations or when concerted atomic motions are required to evoke molecular movement. Systems that are challenging to sample include:

- dense liquids and solids
- low-temperature systems
- macromolecular or polymeric systems
- "associating" molecules with highly directional interactions (e.g., water with hydrogen bonds)
- "frustrated" systems whose inherent kinetics are very slow (e.g., supercooled liquids and glasses)
- systems that transition to, from, or between various "ordered" phases, such as crystals, nano- and meso-structured complex fluids, folded biomolecules, etc.

Oftentimes, multiple kinds of MC moves are used in the same simulation run. This can be to accomplish all of the fluctuations in the ensemble of interest, or simply to enhance the sampling. Keep in mind that the selection of a particular kind of move at each MC step must be done randomly, i.e., by drawing a random number. Not doing so ultimately breaks the underlying Markov chain because it destroys the Markov property.

In this lecture, we discuss a small number of advanced MC moves and move sets. We will not always present a detailed derivation of the acceptance criteria for each, but these are available in Leach and in Frenkel and Smit.

Biased moves

One way to enhance the efficiency of MC moves is to **bias** them. That is, we compute some information about the system that allows us to preferentially choose moves with a higher probability of acceptance. Such moves always modify the acceptance criterion. All of our derivations will be based on the fundamental detailed balance relation

$$\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} = \frac{\alpha_{21}\wp_2}{\alpha_{12}\wp_1}$$

Biased insertion

In GCMC simulations, it can be challenging to successfully insert a molecule in dense systems. One way around this would be to selectively insert the molecule at locations in the simulation box that are "empty", or that better accommodate insertion. For this kind of move, we need to re-derive the acceptance criterion for insertion, following the original GCMC derivation:

$$\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} = \frac{\alpha_{21}\wp_2}{\alpha_{12}\wp_1}$$

Here, the move proposal probability α_{12} corresponds to the probability that we will insert a particle in state 1 that will correspond to the location of the additional particle in state 2. In the original acceptance move, we chose this location randomly

$$\alpha_{12} = \frac{d\mathbf{r}}{V}$$

In a biased insertion, we choose a location from a subset of the volume that corresponds to areas where we have identified a high-probability of insertion success. Such areas may be those that do not have any core overlaps with other particles. Let the total volume of this subset be ϵV where $\epsilon \leq 1$. Then,

$$\alpha_{12} = \frac{d\mathbf{r}}{\epsilon V}$$

All other quantities in the acceptance relation are the same as in the original insertion. Thus, the final acceptance criterion, using the Metropolis rule, is:

$$P_{12}^{\text{acc}} = \min\left[1, \frac{\epsilon V}{N+1} e^{-\beta\Delta U + \beta\mu'}\right] \qquad \text{for particle insertion}$$

For a particle deletion, the acceptance criterion is:

$$P_{12}^{\text{acc}} = \min\left[1, \frac{N}{\epsilon V} e^{-\beta\Delta U - \beta\mu'}\right] \qquad \text{for particle deletion}$$

In *both* cases, we have to compute the value of ϵ:

- For the insertion case, we compute ϵ before the insertion.

- For the deletion case, we have to compute ϵ *after* deletion. The reason we need to compute it after deletion is that it is the reverse move (P_{21}^{acc}) that incorporates ϵ into the

> acceptance criterion, not the forward one. Moreover, if the molecule being deleted does not lie within the ϵV volume found after deletion, we must reject the deletion move because there is zero probability that the reverse move will be performed.

How do we compute the available insertion volume ϵV? This can be a computationally intense procedure. One simple approach might be to divide the simulation box into cubic cells and identify those cells that are free of particles. Ultimately, however, if our biased moves are able to achieve much better acceptance rates than nonbiased moves, this computational cost might be worth it—the evolution of the system as a function of CPU effort might actually be faster.

Force bias

For dense systems, displacement moves tend to place particles on top of other particles, which results in a large increase in energy and chance for rejection. One way to improve displacement moves might be to bias them in the direction of the force, which corresponds to the direction of decreasing potential energies.

In the **force bias** approach, like single particle displacements, a particle is displaced in its x, y, z coordinates by random amounts in the range $[-\delta r_{\max}, \delta r_{\max}]$. However, displacements are not picked uniformly in this range. Rather, they are biased in the direction of the force. Take the x-coordinate. The probability of picking a displacement Δx is taken to be:

$$\wp(\Delta x) \propto \exp(\lambda \beta f_x \Delta x)$$

Here, f_x is the force on the displaced particle in the x direction, evaluated at the original configuration (before displacement). The term $-f_x \Delta x$ is a first-order expansion of the change in energy around the initial particle location, and thus this probability approximates the Boltzmann weights around the initial configuration. The parameter λ is used to tune how much we use the force to bias the move. For $\lambda = 0$, we recover usual single-particle displacements.

The probability above must be normalized within the window $[-\delta r_{\max}, \delta r_{\max}]$. Define

$$\begin{aligned} C_x &= \int_{-\delta r_{\max}}^{\delta r_{\max}} \exp(\lambda \beta f_x \Delta x)\, d\Delta x \\ &= \frac{\sinh(\lambda \beta f_x \delta r_{\max})}{\lambda \beta f_x} \end{aligned}$$

Then

$$\wp(\Delta x) = \frac{\exp(\lambda \beta f_x \Delta x)}{C_x}$$

Similar expressions exist for the y and z coordinates. To pick values of $\Delta x, \Delta y, \Delta z$, therefore, one must sample from a truncated exponential distribution of this form. There are a number of numerical methods for accomplishing this task.

Once we have proposed a move, we need to find the appropriate acceptance criterion. Notice that the selection of displacement values now modifies our move proposal probabilities α_{12} and α_{21}. In fact, these two probabilities are no longer symmetric. Let

$$\wp(\Delta\mathbf{r}) = \wp(\Delta x)\wp(\Delta y)\wp(\Delta z)$$

This is the probability of picking a displacement vector $\Delta\mathbf{r}$. Notice that this probability *depends on the initial configuration*, because this is where the forces are evaluated. Then,

$$\alpha_{12} = \wp_1(\Delta\mathbf{r})$$

For the reverse move, a force-bias move will use the forces at the new configuration, not the original one. Therefore,

$$\alpha_{21} = \wp_2(-\Delta\mathbf{r})$$

As a result,

$$\frac{\alpha_{21}}{\alpha_{12}} = \frac{\wp_2(-\Delta\mathbf{r})}{\wp_1(\Delta\mathbf{r})}$$

Therefore, the final acceptance criterion with the Metropolis rule is modified from usual single particle displacements as

$$\begin{aligned} P_{12}^{\text{acc}} &= \min\left[1, \frac{\wp_2(-\Delta\mathbf{r})}{\wp_1(\Delta\mathbf{r})} e^{-\beta\Delta U}\right] \\ &= \min\left[1, \exp\left(-\beta\Delta U - \ln\frac{\wp_2(-\Delta\mathbf{r})}{\wp_1(\Delta\mathbf{r})}\right)\right] \\ &= \min\left[1, \exp\left(-\beta\Delta U - \lambda\beta(\mathbf{f}_1 + \mathbf{f}_2)\cdot\Delta\mathbf{r} + \ln\frac{C_{1,x}C_{1,y}C_{1,z}}{C_{2,x}C_{2,y}C_{2,z}}\right)\right] \end{aligned}$$

Notice that we have to compute two forces and two sets of C values: one at the initial configuration and one at the final.

By tuning the parameter λ, the force bias approach can result in acceptance ratios that are much higher than typical particle displacements. However, the additional cost of evaluating the forces can reduce the overall efficiency of the simulation in real CPU time. Thus, typical speedups of only 2-3 are found for this approach.

Orientational bias

For rigid molecules that have strongly orientational-dependent interactions, simple orientational moves typically have low acceptance rates. Such systems include molecules with strong dipoles or hydrogen bonding, like water. To increase the acceptance ratio of these moves, we can bias orientations to energetically-favorable conformations. This procedure works well if we can decompose the potential energy into two parts: a positional part depending only on the center-of-mass position of the molecule and a computationally less expensive orientational component depending only on the orientation, for fixed position:

$$U = U^{\text{pos}} + U^{\text{or}}$$

The **orientational bias** algorithm works as follows:

1. Perform a usual displacement move on the center-of-mass position of the rigid molecule.
2. Generate K trial orientations at the new position.
3. Select one of these conformations, say i, with a probability proportional to the Boltzmann factor,

$$\wp(i) = \frac{e^{-\beta U_i^{\text{or}}}}{\sum_{j=1}^{K} e^{-\beta U_j^{\text{or}}}}$$

4. Determine whether or not to accept the selected conformation *and displacement* using an acceptance criterion.

To determine the acceptance criterion, we impose detailed balance as before. However, we must consider the probability of the reverse move in this case, in which the current orientation is selected from K possible configurations generated at the proposed state. To do so, we must generate $K - 1$ random additional orientations at the current state to simulate the kind of move we would perform in reverse.

With these considerations, the acceptance criterion can be determined. A detailed derivation is available in Frenkel and Smit. Here we only present the final result.

$$P_{12}^{\text{acc}} = \min\left[1, \frac{w_2}{w_1}\exp(-\beta \Delta U^{\text{pos}})\right]$$

The quantity w is a weight and it depends on the K random orientations generated at the new state 2 as well as the $K - 1$ orientations generated at the current state 1:

$$w_1 = e^{-\beta U_1^{\text{or}}} + \sum_{j=2}^{K} e^{-\beta U_j^{\text{or}}}$$

$$w_2 = \sum_{j=1}^{K} e^{-\beta U_j^{\text{or}}}$$

The first term in w_1 corresponds to the original orientation at state 1. The other terms in the sums for w_1 and w_2 correspond to the $K-1$ and K random orientations generated at the center of mass positions for 1 and 2, respectively.

Hybrid MC

For nonrigid molecules and particularly for complex heterogeneous systems, displacement moves can become extremely inefficient because of the way in which they distort bond lengths and angles. Such systems include macromolecules like proteins and polymers. In these cases, it is very difficult to identify intelligent move sets that are able to propagate the system in configurational space.

On the other hand, molecular dynamics methods are quite able to evolve such systems in time since they provide deterministic pathways on a system's potential energy surface. Thus, we might think about using such schemes to generate trial configurations in Monte Carlo simulations.

The **hybrid Monte Carlo method** uses short MD trajectories to generate trial configurations. While it may seem computationally expensive to implement MD within an MC simulation, this approach can sometimes be one of the few viable ways of implementing moves with nonzero acceptance probability in very complex systems. This approach was developed by Duane et al in 1987. It entails the following steps:

1. Generate a set of initial momenta according to a Boltzmann distribution at the temperature of interest. This amounts to picking each velocity component randomly from a Gaussian distribution such that the total joint probability distribution follows:

$$\begin{aligned} \wp(\mathbf{p}^N) &\propto e^{-\beta K(\mathbf{p}^N)} \\ &= e^{-\beta \sum_i \frac{\mathbf{p}_i^2}{2m_i}} \end{aligned}$$

2. Perform a short molecular dynamics trajectory for a fixed time t using a given fixed time step δt. It is very important that the integrator be **reversible**.

3. Accept or reject the move according to an acceptance criterion.

The acceptance criterion for this kind of move is developed in a similar manner as before. We must recognize that the configuration produced at the end of the trajectory is *uniquely* specified by the initial set of momenta chosen. That is, there is a one-to-one correspondence between the initial momenta and final configuration. If there were not, the time evolution in molecular dynamics would not be deterministic.

The probability that we will move to a new configuration is given by the probability that we pick the right set of momenta that will lead us to it after the MD integration procedure

$$\begin{aligned}\alpha_{12} &= \wp(\mathbf{p}_1^N)\\ &\propto e^{-\beta K(\mathbf{p}_1^N)}\end{aligned}$$

This can be show rigorously using Louiville's equation. For the reverse move, we must consider the probability that we will pick a set of momenta leading us back to the original conformation after the same amount of integration time. This is just,

$$\begin{aligned}\alpha_{21} &= \wp(\mathbf{p}_2^N)\\ &\propto e^{-\beta K(\mathbf{p}_2^N)}\end{aligned}$$

Note that $\mathbf{p}_2^N$ is the set of momenta at configuration 2 that must be picked to return us to the current configuration with a molecular dynamics trajectory. Since the dynamics are reversible, we must have

$$\mathbf{p}_2^N(0) = -\mathbf{p}_1^N(t)$$

That is, the momenta after integrating a time t from state 1 must be negated at state 2 to bring us back to state 1. This is only possible if the trajectory is **reversible**, and thus we require our integrator to have this property. Note that this implies $K_1(t) = K_2(0)$. That is, the kinetic energy at the end of the MD trajectory is equal to that at the beginning of the reverse move.

Using detailed balance in the canonical ensemble,

$$\begin{aligned}\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} &= \frac{\alpha_{21}}{\alpha_{12}}\frac{\wp_2}{\wp_1}\\ &= \left(\frac{e^{-\beta K_2}}{e^{-\beta K_1}}\right)\left(\frac{e^{-\beta U_2}}{e^{-\beta U_1}}\right)\\ &= e^{-\beta \Delta E}\end{aligned}$$

where $\Delta E = K_2 + U_2 - K_1 - U_1$. Here, ΔE is the change in energy from the beginning to the end of the short trajectory. Thus an acceptance criterion *for the entire MD trajectory from 1 to 2* would be:

$$P_{12}^{\text{acc}} = \min\left[1, e^{-\beta\Delta E}\right]$$

For very small time steps δt, the total energy is highly conserved, $\Delta E \approx 0$, and such a move will always be accepted. This is because of the Boltzmann sampling of initial velocities and the fact that Newton's equations follow a constant energy hypersurface. Thus, small time steps lead to what looks like a long MD trajectory with periodic velocity resampling from a Boltzmann distribution. This is identical to the Andersen thermostat discussed in earlier lectures.

Because of the 100% acceptance of hybrid moves, it may not seem advantageous to use this approach in favor of a straightforward MD simulation. However, one advantage of the acceptance criterion above is that it still holds rigorously true for large δt and when the energy is *not* conserved well. That is, we can treat δt as a sort of maximum displacement parameter and tune its value so as to achieve ~50% acceptance. By tuning δt up, energy will no longer be well conserved and we will have nonzero ΔE.

In practice, the choice between a molecular dynamics run with a thermostat and the hybrid MC approach typically tips towards the former due to the straightforward nature of the dynamical evolution. However, in particular cases, hybrid MC may offer benefits. For example, one may evolve a system in short MD trajectories according to an approximate potential that is much faster to compute than the actual force field. The acceptance criterion above always corresponds to the original potential, but any force field can be used for the dynamical evolution. One example might be a system using the Ewald summation in its potential: here, one might use simple screened Coulomb interactions to guide the MD trajectory, which are much faster to compute.

Approximate potential or multiple "time" step MC moves

One way to improve the efficiency of a MC simulation is not to increase the acceptance rate of the moves themselves, but to speed up the simulations as a whole in real CPU time so that longer trajectories can be performed. This is possible if one can identify an approximate potential that is much faster to compute than the true force field, but that behaves similarly. In such cases, one can perform a MC simulation that rigorously converges to the ensemble of the true (expensive) force field while allowing much faster computations due to the approximate force field.

Let the true force field be U and the approximate potential be $\widetilde{U}$. Then, the basic procedure is:

1. Make n MC moves using the approximate potential $\widetilde{U}$. Accept or reject them as would be done otherwise if $\widetilde{U}$ were the true potential.

2. At the end of the short n-step mini-trajectory, accept or reject *the entire trajectory generated with $\widetilde{U}$ using the true potential U.*

3. Repeat steps 1 and 2. Each n-step mini-trajectory is in essence one MC move using the true potential. Therefore, the trajectory used for property averages corresponds to the configurations generated after each acceptance or rejection of the mini-trajectories.

We need to determine the acceptance criterion for a mini-trajectory. To do so, we need to know the probability of going from an initial state 1 at the beginning of the trajectory to a final state 2 at the end of it, after n steps. Denote this probability as $\pi_{12}^{(n)}$. Assuming that each step within the trajectory obeys detailed balance, it can be shown that

$$\tilde{\wp}_1 \pi_{12}^{(n)} = \tilde{\wp}_2 \pi_{21}^{(n)}$$

That is, the mini-trajectories obey detailed balance as well. Here, the tilde characters indicate that we are sampling from a probability distribution that corresponds to the approximate potential,

$$\tilde{\wp}_m \propto e^{-\beta \tilde{U}_m}$$

The above result is an important one because it allows us to compute the ratio of transition probabilities of the mini-trajectory:

$$\frac{\pi_{21}^{(n)}}{\pi_{12}^{(n)}} = \frac{\tilde{\wp}_1}{\tilde{\wp}_2}$$

Now, we consider one mini-trajectory as a MC step using the true potential. We pick an acceptance criterion for this step using the usual detailed balance equation:

$$\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} = \frac{\alpha_{21} \wp_2}{\alpha_{12} \wp_1}$$

Here, the α's are given by the transition probabilities for the mini-trajectory. Thus:

$$\begin{aligned}\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} &= \frac{\pi_{21}^{(n)} \wp_2}{\pi_{12}^{(n)} \wp_1} \\ &= \frac{\tilde{\wp}_1 \wp_2}{\tilde{\wp}_2 \wp_1}\end{aligned}$$

In the canonical ensemble,

$$\begin{aligned}\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} &= e^{-\beta(U_2 - U_1) + \beta(\tilde{U}_2 - \tilde{U}_1)} \\ &= e^{-\beta \Delta U + \beta \Delta \tilde{U}}\end{aligned}$$

With the Metropolis rule,

$$P_{12}^{\mathrm{acc}} = \min\left[1, \exp\left(-\beta\Delta U + \beta\Delta\widetilde{U}\right)\right]$$

Here, keep in mind that the subscripts 1 and 2 correspond to the configurations *before* and *after* the mini-trajectory, not the configurations in between. This expression is used to accept or reject the entire mini-trajectory. Note that it enforces a sort-of "correction" step, in which deviations from the true potential are used to decide whether to accept or reject the trajectory. If the true potential equals the approximate one, then the mini-trajectories are always accepted because $\Delta U = \Delta\widetilde{U}$.

For this approach to be effective, one must be able to find an approximate potential that is (1) computationally faster than the true one, and (2) a good approximation to the true potential. If the latter is not the case, then the mini-trajectories will have a high rate of rejection.

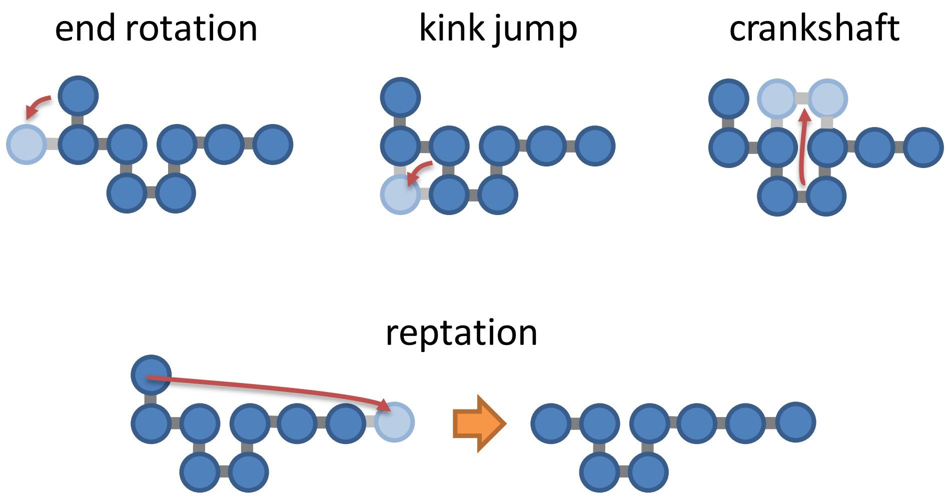
Such an approach might be used for systems that have long-ranged electrostatic interactions. Here, the true potential might involve an Ewald sum, while the approximate one might involve the bare (untreated) or a screened Coulombic interaction. Note that the parameter n must be chosen. This might be tuned to balance a high acceptance ratio and efficiency of movement through configuration space.

Chain molecules on lattices

For polymeric systems, it can be very difficult to develop efficient MC moves due to the intertwining nature of the chains that prevents substantial movement through phase space in single-atom displacement. In these cases, a number of specialized moves have been developed. These moves were originally applied to **lattice polymer** systems, in which each monomer is constrained to sit on a cubic lattice point. The discussion of these moves is easiest to explain in the lattice case, and will be our focus here; however, there have been extensions of such moves to continuum classical systems. We will not address the latter, but details can be found in Frenkel and Smit and in the literature.

Displacement-type moves

There are a number of moves for displacing multiple monomer sites at once in a polymeric system. These are shown in the figure below:



These moves are all symmetric and thus the usual Metropolis criterion can be applied. Note that the reptation move resembles snake-like motion in the polymeric system.

Configurational bias

Insertion moves pose major challenges for polymeric systems. The configurational bias method enables one to incrementally insert monomers of a polymer into a system in a way that avoids overlaps with existing molecules on the lattice. This results in a modification of the acceptance criterion.

Here is how the approach works:

1. Choose a random lattice site to place the first monomer of the polymer.

2. Look around this site at the z neighbors surrounding it. Here, z is the coordination number of the lattice (2 for 1D, 4 for 2D, and so on). Count the number of free sites. Let this be z_2. The 2 subscript indicates the second monomer.

3. Randomly pick one of the z_2 sites to place the second monomer.

4. Continue this process until all M monomers have been placed down.

5. Accept or reject the insertion.

The acceptance criterion is given by the probability for proposing the move given. The initial site is chosen with probability

$$\wp^{(1)} = \frac{1}{V}$$

where V is the number of lattice sites. The second site is chosen with probability

$$\wp^{(2)} = \frac{z_2}{z}$$

The third and all higher order sites are chosen with probability

$$\wp^{(i)} = \frac{z_i}{z-1}$$

The denominator in this case accounts for the fact that the site of the previous monomer is not available for selection.

The total move proposal probability is given by the product of all of these individual probabilities:

$$\alpha_{12} = \prod_i \wp^i$$

The reverse move corresponds to deletion of the inserted particle,

$$\alpha_{21} = \frac{1}{N+1}$$

Therefore, the acceptance probability is given by:

$$\begin{aligned}\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} &= \frac{\alpha_{21}\wp_2}{\alpha_{12}\wp_1} \\ &= \frac{1}{N+1}\left[\prod_i \wp^{(i)}\right]^{-1}\frac{\wp_2}{\wp_1} \\ &= \frac{V}{N+1}\frac{z(z-1)^{M-2}}{\prod_{i=2}^{M} z_i}\ e^{-\beta\Delta U}\end{aligned}$$

The above approach can be enhanced if we also incorporate energetics in the selection of locations to place monomers. At each step in the polymer build process, we can choose a subsequent neighboring lattice site with probabilities proportional to the energies that a monomer would have if sitting at that site. At each step:

1. Compute the energies u_j of interactions of the ith monomer placed at the z or $z-1$ possibilities for neighboring sites j.

2. Pick one of the j sites with a probability equal to

$$\wp^{(i,j)} = \frac{e^{-\beta u_j}}{w_i} \qquad w_i \equiv \sum_j e^{-\beta u_j}$$

3. Compute the total **Rosenbluth weight**:

$$W = \prod_{i=2}^{M} w_i$$

4. Accept/reject the move according to:

$$\begin{aligned}
\frac{P_{12}^{\text{acc}}}{P_{21}^{\text{acc}}} &= \frac{V}{N+1}\left[\prod_{i=2} \wp^{(i)}\right]^{-1} e^{-\beta \Delta U} \\
&= \frac{V}{N+1}\left(\prod_{i=2} w_i\, e^{\beta u_i}\right) e^{-\beta \Delta U} \\
&= \frac{V}{N+1}\left(\prod_{i=2} w_i\right) \\
&= \frac{V}{N+1} W
\end{aligned}$$

The third line comes from the fact that $\sum_i u_i = \Delta U$.

Configurational bias deletions involve a similar acceptance criterion. In these cases, one must account for the Rosenbluth weight of the deleted molecule in the acceptance criterion. It is computed as before, retracing the insertion of the polymer from the beginning.

Displacement-type moves can be accomplished using a similar approach, in which part of a polymer molecule is deleted and the monomers are then reconstructed using a configurational bias procedure.