Simple Biasing Methods

One of the strongest features of Monte Carlo simulation is the freedom it provides to devise any type of trial move to enhance sampling of the ensemble. So far we have encountered just the basic trial moves needed to perform simulations of the more popular ensembles (canonical; isothermal-isobaric; grand-canonical). These trial moves are *unbiased* in the sense that no information about the current configuration enters into their prescription; these details enter only at the trial-acceptance stage. *Biased* trials are made in a way that does use information about the configuration, and if designed well the consequence is that such trials are more likely to be accepted. Note that biased sampling, in the sense we mean it here, does not involve any change in the limiting distribution. The biasing we introduce in the trial is offset somewhere else, usually in the formulation of reverse trial move and the acceptance probabilities. This contrasts with biasing techniques that do attempt to modify the limiting distribution. In these methods the biasing is removed by modifying the averages taken in the simulation. No modifications of the averages are needed in the techniques we presently consider, which modify the transition probabilities while maintaining detailed balance for a given limiting distribution.

We now turn to a discussion of how clever trial moves can be devised to perform better sampling. It is appropriate to begin by considering what constitutes "better sampling", so we first define and examine some performance measures of a MC simulation. To make things concrete we return to our elementary discussion of a Markov process that samples just a few states. We can look to see what features of the transition-probability matrix yield better values of the performance measures. We then follow with a few examples of some specific biasing schemes in Monte Carlo simulation.

Performance measures

The performance of a Monte Carlo simulation usually concerns two issues: the rate of convergence to equilibrium, and the reproducibility of the averages. The rate of convergence can be addressed in terms of the question: What is the likelihood of being in a particular state after a run of finite length n? Is this likelihood close to the likelihood in the limiting distribution? Remember that the transition probability for a multi-step move is given as a power of the transition probability matrix governing the single-step moves. For a simple three-state process

$$\pi_1^{(n)} = \pi_1^{(0)} \Pi^n \equiv \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \pi_{11}^{(n)} & \pi_{12}^{(n)} & \pi_{13}^{(n)} \\ \pi_{21}^{(n)} & \pi_{22}^{(n)} & \pi_{23}^{(n)} \\ \pi_{31}^{(n)} & \pi_{32}^{(n)} & \pi_{33}^{(n)} \end{pmatrix} = \begin{pmatrix} \pi_{11}^{(n)} & \pi_{12}^{(n)} & \pi_{13}^{(n)} \end{pmatrix}$$

Although the probability distribution after *n* steps depends on the initial state, the key feature is the behavior of the matrix Π^n and thus Π itself, so we focus our analysis on it. The eigenvectors and eigenvalues of Π are given by

$$\Pi \Phi = \Phi \Lambda \tag{1.1}$$

where Φ is a matrix with columns formed from the eigenvectors of Π , and Λ is a diagonal matrix with elements given by the corresponding eigenvalues of Π . Equation (1.1) is manipulated to a similarity transform for Π

$$\Pi = \Phi \Lambda \Phi^{-1}$$

From this it is easy to obtain a useful expression for Π^n

$$\Pi^n = \Phi \Lambda^n \Phi^{-1}$$

The matrix Λ^n is also a diagonal matrix with elements given by powers of the eigenvalues, λ^n . It can be shown that, because Π is a probability matrix (non-negative elements with rows that sum to unity), one eigenvalue will be unity and all other eigenvalues will be of magnitude less than unity. The unit eigenvalue dominates for large *n*, and the rate at which the other eigenvalues vanish for large *n* characterizes the rate of convergence of the simulation. Thus a good form for the transition probability matrix has sub-dominant eigenvalues with magnitudes as close to zero as possible.

Illustration 1 gives some examples of different transition-probability matrices that all correspond to the same limiting distribution. The matrix labeled "Inefficient" has subdominant eigenvalues of 0.96. This can be compared with the matrix formed from the Metropolis recipe (when in state *i* choose one of the other two states with equal probability to be trial state *i*, and accept the transition with probability min[1, π_i/π_i]), which has one zero sub-dominant eigenvalue and one equal to -0.5. According to our analysis the Metropolis scheme will converge more quickly to the limiting distribution than the "Inefficient" algorithm. Also present in this example is the matrix formed from the Barker algorithm (which we haven't discussed in detail because it is not widely used). Our convergence measure indicates the Barker algorithm to be comparable to Metropolis—Barker has two subdominant eigenvalues of magnitude between the two Metropolis values. Also included in this example is a very efficient scheme, which prescribes a transition from state 2 to states 1 or 3 (chosen with equal likelihood), immediately followed (with probability 1) by a move back to state 2. In this case the system never stays in the same state twice. The example is anomalous because the system never forgets its initial state; if starting in state 2, for example, after an even number of trials, no matter how many, it will always be back in state 2. Associated with this long memory is a second, negative eigenvalue of unit magnitude whose influence does not decay for large *n*.

An issue separate from the rate of convergence is the reproducibility of the results. This is characterized by the variance of an average, taken over many independent realizations of the Markov process. In our simple example we don't have any particular averages in mind, but we can characterize the reproducibility of a process by examining the variance in the state occupancies. The question we address is: If an M-step Markov process is

repeated many times independently, how much variation do we observe in the fraction of time spent in any given state? From one M-step run to another (where M is large), is the fractional occupancy reproducible? Through propagation of error the variance in any average can be expressed in terms of the variances and covariances of the occupancies. For example the variance in some property U defined on a two-state system would be

$$\sigma_U^2 = U_1^2 \overline{\sigma_1^2} + U_2^2 \overline{\sigma_2^2} + 2U_1 U_2 \overline{\sigma_1 \sigma_2}$$

where the σ_1 and σ_2 terms form variances and covariances of the occupancies of states 1 and 2. We cannot make a too general statement about how to minimize σ_U^2 , because the covariance can be negative. If we know U₁ and U₂, the values of the property U in states 1 and 2, we could rewrite the right-hand side as a sum of squares, and then declare a good outcome as one that minimizes each term individually. Lacking a general way to do this, we will instead consider the effect of the transition probabilities on the occupancy variances and covariances themselves. All else being equal, it is better that all variances and covariances be close to zero.

Fortunately, there exists a very nice formula that can give us these occupancy variances for a given transition probability matrix:

$$M\overline{\sigma_i^2} = \pi_i^2 + 2\pi_i s_{ii} - 1 \quad \text{variance}$$
$$M\overline{\sigma_i\sigma_j} = \pi_i \pi_j + \pi_i s_{ij} + \pi_j s_{ji} \quad \text{covariance}$$

where s_{ij} are the elements of the matrix

$$S = (\mathbf{I} - \Pi + \Phi)^{-1} - \Phi$$

where **I** is the identity matrix. Note that the right-hand side of the variance equation is independent of M, indicating that the group on the left is also independent of M. This is consistent with our previous experience with the 1/M decay of the variance in a Monte Carlo process. The matrix of covariances determined by this formula is included with the examples in Illustration 1. In terms of the average magnitude of the terms, the matrix labeled "Most efficient" gives the best performance (variance of order 0.1), while the "Inefficient" algorithm is clearly inferior (with variances of order 10). Whereas the convergence measure was equivocal regarding the merits of the Metropolis algorithm versus the Barker approach, by this measure Metropolis is markedly better (explaining why it is more widely used).

A simple look at the good versus the poor transition-probability matrices gives some indication of what is desired in a good algorithm. Clearly it is better to minimize the diagonal elements, meaning it is desirable to keep the process moving regularly from one state to the next. But this in itself is not sufficient to ensure good performance. Consider the 4-state process displayed in Illustration 1. Here all the diagonal elements are zero, but the convergence and variance measures are still poor. The problem is that there is a bottleneck from states 1 and 2 to states 3 and 4. A good algorithm promotes movement

of the system among a wide variety of states, and does not let the system get trapped into sampling a small subset of all the relevant states (*i.e.*, those with non-negligible probabilities in the limiting distribution).

You can experiment with other transition probability matrices using the applet in Illustration 2.

	Limiting distributi	$\pi = (0.25 \ 0.5 \ 0.5)$).25)
Inefficient	$\Pi = \begin{pmatrix} 0.97 & 0.02 & 0.01 \\ 0.01 & 0.98 & 0.01 \\ 0.01 & 0.02 & 0.97 \end{pmatrix}$	$\lambda = (1 0.96 0.96)$	$\Sigma = \begin{pmatrix} 9.2 & -6.1 & -3.1 \\ -6.1 & 12.2 & -6.1 \\ -3.1 & -6.1 & 9.2 \end{pmatrix}$
Barker	$\Pi = \begin{pmatrix} 0.42 & 0.33 & 0.25 \\ 0.17 & 0.66 & 0.17 \\ 0.25 & 0.33 & 0.42 \end{pmatrix}$	$\lambda = (1 0.33 0.17)$	$\Sigma = \begin{pmatrix} 0.30 & -0.25 & -0.05 \\ -0.25 & 0.50 & -0.25 \\ -0.05 & -0.25 & 0.30 \end{pmatrix}$
Most efficient	$\Pi = \begin{pmatrix} 0 & 1 & 0 \\ 0.5 & 0 & 0.5 \\ 0 & 1 & 0 \end{pmatrix}$	$\lambda = \begin{pmatrix} 1 & 0 & -1 \end{pmatrix}$	$\Sigma = \begin{pmatrix} 0.125 & 0 & -0.125 \\ 0 & 0 & 0 \\ -0.125 & 0 & 0.125 \end{pmatrix}$
Metronolis	$\Pi = \begin{pmatrix} 0.0 & 0.5 & 0.5 \\ 0.25 & 0.5 & 0.25 \\ 0.5 & 0.5 & 0.0 \end{pmatrix}$	$\lambda = \begin{pmatrix} 1 & 0 & -0.5 \end{pmatrix}$	$E = \begin{pmatrix} 0.10 & -0.125 & 0.02 \\ -0.125 & 0.25 & -0.125 \\ 0.02 & -0.125 & 0.10 \end{pmatrix}$
	Limiting distribution	$\pi = (0.25 0.25 0.25$	0.25)
Lots of movement $1 \Leftrightarrow 2; 3 \Leftrightarrow 4$ Little movement $(1,2) \Leftrightarrow (3,4)$	$\Pi = \begin{pmatrix} 0 & 0.99 & 0.01 & 0 \\ 0.99 & 0 & 0 & 0.01 \\ 0 & 0.01 & 0 & 0.99 \\ 0 & 0.01 & 0.99 & 0 \end{pmatrix}$	$\lambda = (1 0.98 -0.99 -0.9$	9) $\Sigma = \begin{pmatrix} 6.2 & 6.2 & -6.2 & -6.2 \\ 6.2 & 6.2 & -6.2 & -6.2 \\ -6.2 & -6.2 & 6.2 & 6.2 \\ -6.2 & -6.2 & 6.2 & 6.2 \end{pmatrix}$

Biasing the underlying Markov process

Let us look again at the Metropolis formula for detailed balance

$$\pi_i \tau_{ij} \min(1, \chi) = \pi_j \tau_{ji} \min(1, 1/\chi)$$
(1.2)

With unbiased sampling it often happens that τ_{ij} is small while π_j/π_i (and thus χ) is large, indicating that the system is eager to accept a particular trial if only it could find it. Alternatively sometimes τ_{ij} is non-negligible but χ is very small, causing commonly encountered trials to be rarely accepted. In these situations biasing methods can be used to steer the system toward the moves it is eager to accept while moving it away from trials that won't likely be kept.

Equation (1.2) requires

$$\chi = \frac{\pi_j \tau_{ji}}{\pi_i \tau_{ij}}$$

As discussed above, one of our aims in enhancing performance is to keep the process moving, which means we would like for χ (and hence $1/\chi$) to be close to unity. In this situation the underlying transition probabilities themselves satisfy detailed balance, and the system moves into a state with probability in direct proportion to its ensemble probability. If candidate states included all in the ensemble, then this perfect situation is no different than direct importance sampling of the ensemble. If it could be achieved there would be no need for the Markov process. But it cannot, and we must settle for something less. As in the usual Metropolis scheme, we restrict transitions to states that are not wildly different from the current one, but we use biasing to select intelligently from these states. Another use of biasing is to expand the base of states considered by a single trial. If we can get the system to move into a state that is wildly different from the current one, even if it has a small but non-negligible probability of being accepted, we are likely to improve the performance of a simulation.

Insertion bias in GCMC simulations

As our first example we will consider the molecule insertion/deletion trials in grandcanonical Monte Carlo simulation. Earlier we presented the algorithm and derived transition probabilities for these trials. An important problem encountered in these simulations arises when the system density is high, which corresponds to a large value of the imposed chemical potential μ . The difficulty is described by Illustration 3. The insertion trial specifies that a new molecule be placed at a random position in the simulation volume. At high density the most likely outcome of such a trial is an overlap with another molecule. The result is a high-energy configuration that is sure to be rejected. Indeed, for hard spheres near the freezing density the probability of random insertion without overlap is about 10⁻⁷. Complementing this problem is a similar one with deletion. A molecule selected at random is likely to be nicely nestled in the energy minima of its neighbors, so its removal is going to be accompanied by a substantial



increase in the energy. Compounding the issue is the fact that the chemical potential is large, and the acceptance of the removal trial is proportional to $\exp[-\beta\mu]$. Consequently deletion trials are as likely to be rejected as insertion trial (of course this must be so or there would be a net removal of molecules). Experiment with the applet given in Illustration 4 (GCMC simulation applet) to see how infrequently insertions and removals are completed at high chemical potential.

One way to remedy this problem is to bias the insertion trials to regions of the simulation volume that do not result in overlap. Finding such regions is a nontrivial problem that we



Non-overlap region

will set aside for the purposes of this example. The concept is described by Illustration 5. The green regions describe non-overlap positions for placement of the inserted molecule. Let us designate the volume of this region as εV , where V is the system volume and ε is small if the density is large. An insertion trial consists of the placement of a molecule at a point selected uniformly within this region. The deletion trial is conducted as in the unbiased case.

The trial moves and their component probabilities are summarized in Illustration 6. The only difference from the unbiased case is the presence of the ε term in the molecule placement probability. With the bias in the trial, acceptance parameter χ becomes

$$\chi = \frac{\varepsilon V \lambda}{(N+1)} e^{-\beta \Delta U}$$

For large λ the fractional volume available for insertion ϵ is small, but their product may yield a χ of order 1.

Event	Probability
[reverse event]	[reverse probability]
Select insertion trial	¹ / ₂
[select deletion trial]	[¹ / ₂]
Place molecule at r _{N+1}	dr/(εV)
[delete molecule N+1]	[1/(N+1)]
Accept move	$\min(1,\chi)$
[accept move]	[min(1,1/ χ)]

Now we come to an important point. For both the insertion and deletion trials, the acceptance parameter χ is as given here. Insertion is accepted with probability min[1, χ], while deletion is accepted with probability min[1,1/ χ]. Note therefore that both the insertion and deletion trials need to know ε . For an insertion trial, ε is computed naturally in the course of identifying the insertable region. In deletion, ε is not needed to conduct the trial, but the acceptance probability does depend upon it. The ε computed for this purpose (*i.e.* deciding acceptance of a deletion trial) is that corresponding to the resulting configuration, in the absence of the deleted particle. Remember that the acceptance of this deletion move is based in part on the likelihood that the reverse move will be made; and of course this re-insertion trial begins with, and thus uses an ε for, the configuration obtained if the deletion trial is accepted. This general circumstance is one commonly encountered in biasing algorithms—the acceptance of a given trial requires a (perhaps nontrivial) calculation related to how the reverse trial would be completed.

Derivative bias

One way to bias a move is to use local information obtained from derivatives to lead the system in a favorable direction. An example of this concept is found in the force bias method, in which the direction of movement of an atom is made through consideration of

the force currently acting on it. As with the unbiased case, movement of the atom is still within a finite region centered on its present position. Through adjustment of a parameter, the degree of bias can be selected continuously from no bias at all to a complete bias such that the movement is selected practically deterministically in the direction of the force. In the latter case, if the step size is not too large the energy is practically unchanged by the move, and the technique begins to resemble molecular dynamics. In fact, one of the strengths of the force-bias approach is that it increases the effectiveness of multi-atom moves, which again form the basic step of a molecular dynamics simulation. A related scheme is the virial-bias method for NPT simulations. Here the volume-change trial is conducted with consideration of the difference in the "internal pressure" (the virial) and the imposed ensemble pressure. Note that acceptance of any derivative-bias trial depends on the value of the derivative both before and after the trial move. Details of the methods are reviewed in the text by Allen and Tildesley.

Derivative-bias schemes can hasten the convergence of a MC simulation, but not dramatically. Improvements are of the order of a factor of 2 to 4. Enhancements of this type are welcome, but perhaps not with the added effort needed to implement them. One might prefer to let the simulation run 2 to 4 times longer instead. The most important biasing schemes are those that open up the possibility to do simulations that simply could not be accomplished otherwise. The simple schemes we are presently considering do not all reach that level of performance, although some are more effective than others are worth implementing. In a subsequent chapter we will examine advanced biasing methods that do make the impossible possible (so to speak).

Association bias

Simulations of dilute but strongly interacting molecules are problematic. Their strong interactions (hydrogen bonding, for example) leads to configurations in which the molecules form associated clusters. On the other hand, because they are dilute there is an entropic advantage in having the molecules dissociated and spread roughly uniformly over the large available volume. So both monomers and oligomeric clusters are (or can be, depending on the temperature) prevalent in such systems. The problem that this presents to simulation is the difficulty a molecule has in making transition between being dissociated and being associated. Unassociated molecules must find another cluster before association can occur; the rate at which this happens is roughly proportional to the (small) fraction of the total volume that is occupied by the molecules (more precisely, it is proportional to the fraction of the volume which corresponds to a new bonded configuration). An associated molecule enjoys strongly favorable energetic interactions with its associates, and it is difficult to accept the movement of a molecule from this comfortable position off to some place as an unbonded monomer. Association bias methods aim to remedy this problem.

A simple association-bias scheme is described by Illustration 7. The bias displaces an unassociated molecule into a small region (of volume ϵV) centered on the position of another molecule. In the reverse move an associated molecule is selected and placed at random anywhere in the simulation volume V. The forward and reverse trial probabilities are dissected in Illustration 8. The association trial includes several terms. First is the probability that the given molecule is selected. Since this choice is made considering only the unassociated molecules, the probability is $1/N_u$, where N_u is the number of unassociated molecules in the present configuration. Then another molecule is



selected at random from all other molecules in the system; for any given molecule this probability is 1/(N-1). The first molecule is then placed at a random position selected uniformly in the bias region about the second molecule; the probability of selecting any position here is proportional to $1/(\epsilon V)$. Almost. At this point we must now consider that this very same position might have instead been selected by choosing a different second molecule. This question can be answered easily. Once the position is selected, we look to see if it belongs in the bias volume of any other molecules. We call *n* the number of such molecules found. So the total probability that the first molecule is placed in the new position is given by the expression recorded in Illustration 8. We should also consider the possibility that the molecule is placed in the new position via a standard unbiased atom

displacement trial, which might also be attempted as part of the overall MC sampling. For clarity of the example we will omit this consideration.

The dissociation trial begins by selecting an associated molecule, with probability $1/(N_a+1)$, where N_a is the number of associated molecules in the configuration before the association trial is attempted (so if the attempt succeeds, there will then be N_a+1 associated molecules). The selected molecule is placed at a random position in the volume, with probability proportional to 1/V of selecting the position originally held before the association trial. For these probabilities, detailed balance specifies the acceptance parameter to be

$$\chi = \frac{N_u}{n(N_a+1)} (N-1) \varepsilon e^{-\beta \Delta U}$$

Event	Probability
[reverse event]	[reverse probability]
Select molecule k	1/N _u
[select molecule k]	[1/(N _a +1)]
Move to r ^{new}	n/((N-1)εV)
[move back to r ^{old}]	[1/V]
Accept move [accept move]	$\min(1,\chi)$ $[\min(1,1/\chi)]$

There are a few points to make regarding the algorithm:

- It is necessary to define explicitly what constitutes an "association". Typically this is defined via an energetic criterion (the pair energy between two molecules is below some (negative) value) or a geometric criterion, *i.e.*, two molecules lie in some well-defined geometric region (e.g., a cone) about one another.
- It is necessary to keep track of the number of associated and unassociated molecules, and to be able to select one in either category at random. Alternatively, it is acceptable to choose a molecule at random, then decide whether to do an association or dissociation trial based on its present association state. It should not be taken for granted that a change like this in this algorithm leads to the same acceptance probabilities; a bit of thought is needed to show that the trial probabilities are equivalent.

- Both association and dissociation trials require knowledge of N_u and N_a.
- As described, there is a good chance that the association trial will lead to an overlap rather than to a favorable associated configuration. Despite this, the biasing move may still yield a dramatic improvement in convergence. The overlap condition may occur, say, 50% of the time, yielding then the other 50% of the time a decent association. This contrasts with the unbiased case that may lead to an association 0.01% of the time (for example), depending on how dilute the system is. Regardless, it is simple to modify the algorithm to perform a more intelligent association trial (defining for example the bias volume as a hollow shell rather than a solid cube centered on another molecule).
- An alternative approach to performing an association bias trial involves identifying all points in the simulation volume that correspond to an association state, and then choosing a point uniformly within this volume. Even if the association volume about each molecule is a simple shape, this approach can lead to a very complex set of calculations. It requires one to add up the volume of all such regions (simple enough) then to subtract the volume where these regions overlap, add back in triple overlap volumes, and so on.

Using an approximation potential

As our last example we consider an approach based on an approximation potential. This biasing scheme has a different character than the examples previously presented. The idea is to generate a new configuration in the Markov process by performing a short Markov subprocess using transition probabilities that are based on a simpler molecular model. The simple model is chosen to approximate the true model while being less computationally demanding. For example, the true model may involve a quantummechanical treatment, or three-body interactions, or an Ewald sum, all expensive calculations that might be approximated for this purpose by a simple pairwise additive potential. The acceptance probabilities for the subprocess (based on the approximate potential) can be computed much more quickly than the those for the true Markov process, so the subprocess can be used to move the system a considerable distance from the present configuration at less expense. The configuration at the end of the subprocess forms the trial configuration for the true Markov process. If the trial is rejected, the entire sequence of moves completed by the subprocess is discarded. The approach is depicted in Illustration 9. Note that averages are taken only for those configurations obtained after making an acceptance decision using the true potential.



An acceptance criterion must be applied at the end of the subprocess, and to derive it we need to determine the transition probabilities for multi-step subprocess. Each elementary step in the subprocess is governed by transition probabilities that obey detailed balance for the approximate potential:

$$\pi^a_i\pi^a_{ij}=\pi^a_j\pi^a_{ji}$$

where the superscript a indicates the limiting distribution and the transition probabilities for the Markov process based on the approximate potential. It is possible to show that the multi-step transition probabilities also obey this same detailed balance equation

$$\pi_i^a \pi_{ij}^{a(n)} = \pi_j^a \pi_{ji}^{a(n)} \tag{1.3}$$

where the (n) indicates the n-step transition probability. The proof of this statement is easily made using matrix manipulations. Note that the condition of detailed balance may be written compactly as

$$P\Pi = \left(P\Pi\right)^T \tag{1.4}$$

where *P* is a diagonal matrix with elements given by the limiting-distribution transition probability, and T indicates a matrix transpose. Remembering that the *n*-step transition probabilities are the elements of the matrix Π^n , then the proof merely has to show that Eq. (1.4) implies

$$P\Pi^n = \left(P\Pi^n\right)^T$$

We leave this proof as an exercise.

The trial probabilities for our "true" Markov process are the *n*-step transition probabilities for the approximate potential

$$\tau_{ij} = \pi_{ij}^{a(n)}$$

In the Metropolis scheme detailed balance for the true process requires

$$\pi_i \pi_{ij}^{a(n)} \min(1, \chi) = \pi_j \pi_{ji}^{a(n)} \min(1, 1/\chi)$$

In light of Eq. (1.3), we have

$$\pi_i \left(\frac{\pi_j^a}{\pi_i^a} \pi_{ji}^{a(n)} \right) \min(1, \chi) = \pi_j \pi_{ji}^{a(n)} \min(1, 1/\chi)$$

Which yields for the acceptance parameter χ

$$\chi = \frac{\pi_i^a}{\pi_j^a} \times \frac{\pi_j}{\pi_i}$$

For example, if the simulation were conducted in the canonical ensemble, the limiting distributions are of the form $\pi_i = e^{-\beta U_i} / Z_N$, so the acceptance parameter is

$$\chi = \frac{e^{-\beta U_i^a} / Z_N^a}{e^{-\beta U_j^a} / Z_N^a} \times \frac{e^{-\beta U_j} / Z_N}{e^{-\beta U_i} / Z_N}$$
$$= \exp\left[-\beta (\Delta U - \Delta U^a)\right]$$

where ΔU is the energy difference between the trial and original configurations. We note again our good fortune in the cancellation of the partition functions. If the approximation potential gives a good description of the real potential, the argument of the exponential will be small and χ will be close to unity, indicating that the n-step move will probably be accepted.