JCTC Journal of Chemical Theory and Computation

ARTICLE

pubs.acs.org/JCTC

Block Covariance Overlap Method and Convergence in Molecular Dynamics Simulation

Tod D. Romo and Alan Grossfield*

Department of Biochemistry and Biophysics, University of Rochester Medical Center, Rochester, New York 14642, United States

Supporting Information

ABSTRACT: Molecular dynamics (MD) is a powerful tool for understanding the fluctuations of biomolecular systems. It is, however, subject to statistical errors in its sampling of the underlying distribution of states. One must understand these errors in order to draw meaningful conclusions from the simulation. This is becoming ever more critical as MD simulations of even larger systems are attempted. We present here a new method for determining the extent of convergence that relies on measures of the fluctuation space sampled by the simulation without any a priori knowledge of states or partitioning of the configuration space. This method reveals long correlation times, even for simple systems, and suggests caution when interpreting macromolecular simulations. We also compare this method with previous efforts to characterize the sampling of MD simulation.

1. INTRODUCTION

It has long been known that molecular dynamics (MD) is subject to statistical errors introduced by both the temporal sampling and the scale or length of the simulation. These statistical errors can lead to poor or erroneous estimations of the distribution of states and hence reduce the accuracy of values calculated from the simulation. Many biologically relevant questions that MD attempts to answer require understanding the statistical uncertainty in measurements derived from the simulation. Ideally, the best way to estimate the error is to run multiple independent simulations. However, this is not always practical, so we often wish to determine the error from a single simulation. The size of the statistical uncertainty (error bars) for these measurements is intimately related to the number of independent samples of the quantity in question that are present in the simulation. This number is implicitly dependent on the correlation time for the observable. Estimating the correlation time therefore leads to an estimate for the uncertainty in the measurement.

In the simplest case, determining the precision of measurements from the simulation focuses on a scalar quantity or an observable. In this case, block averaging[1] is the gold standard for determining the statistical error; it works by dividing the trajectory into blocks and computing the standard error of the observable for each block. As the block size is increased, the error estimated approaches the true error. However, methods that rely on a single observable can be misleading due to coupling between fast relaxation and other, slower processes.[2]

In order to safely estimate the standard error, one ought to focus on the slowest relevant relaxations known in the system. As such, a number of groups have attempted to develop measures of global sampling quality. A number of these are based on principal component analysis (PCA). Balsera et al.[3] examined the overlap between fluctuation directions from different sampling windows and found that the dominant modes changed, suggesting a lack of convergence. Amadei et al.,[4] in contrast, used the root-mean-squared inner product for the first 10 directions between two halves of a 2 ns long simulation for protein L and cytochrome c551. This measure was applied to pairs of subtrajectories of increasing size as well as consecutive 50 ps windows and suggested convergence occurring within the subnanosecond regime, when compared to short (nanosecond-scale) simulations. Hess[5] introduced a more detailed measure called the covariance overlap that considers both the directions of fluctuations and their relative magnitudes. He also presented the cosine content measure that compares the projection of the trajectory along a principal component to a cosine. The projection for a diffusive system will be more cosine-like than a system that has sampled multiple conformations. Using the covariance overlap between a trajectory of a protein and all subintervals as well as the cosine content, Hess found that while there was a suggestion of convergence on the order of 10 ns for the system used, a longer simulation was needed to accurately estimate the longest correlation times. Faraldo-Gómez et al.[6] used the covariance overlap to compare consecutive nanosecond blocks as well as blocks of different sizes (1, 2, 4, 8, and 16 ns) as well as the cosine content and found that some regions of a protein (such as the membrane-embedded domains of smaller proteins) can be well-sampled on the order of tens of nanoseconds. They also found that overall undersampling leads to imprecise B-factor predictions. Grossfield et al.[7] used the covariance overlap to characterize the similarity of fluctuations spaces between multiple 100 ns simulations of rhodopsin and found indications of a lack of convergence even with longer simulations. Sullivan et al.[8] introduced a measure of the configuration space sampled by the simulation and examined both the dimensionality of this space over time as well as the dependence of the phase space volume with increasingly large windows sampled from the trajectory. Using this method, transitions between conformational substates were found to occur on the nanosecond time scale, implying that longer time scales are required for adequate statistical sampling.

Received: April 20, 2011
Published: June 22, 2011

There are alternative approaches that do not directly consider the fluctuation directions. Smith et al.[9] compared a number of observables over time, such as intramolecular activation energy, root-mean-square distance, numbers of clusters, and numbers of hydrogen bonds. Lyman et al.[10,11] used a random partitioning of the conformation space of the system along with an analysis of the variance associated with this binning to determine an "effective sample size" that is related to how well it has sampled the distributions of states. This method was also applied to multiple independent simulations of rhodopsin.[7]

The method we have developed is in essence an extension of one used previously to analyze the convergence of membrane proteins[7] and is similar in spirit to the blocked covariance overlap used by Hess.[5] The underlying concept is that if a system is well sampled by MD, then the fluctuation space for sufficiently large subsets of the trajectory should be very similar. The blocked covariance overlap method combines the best aspects of two existing methods: block averaging is the best-of-breed method for assessing statistical error in a single variable, and covariance overlap is a powerful tool for assessing similarity of global fluctuations as a single scalar value.

Here, we present the results of applying this method to long, state of the art simulations of three different classes of biomolecules, ranging in size from a dipeptide to an integral membrane protein. In aggregate, we analyze nearly 29 μs of simulation time from all-atom MD. We show that this method indicates the quality of sampling in a simulation as well as gives an estimate of the rate of convergence, within certain limits. We also compare these results to previous efforts to characterize sampling quality, in particular, the decorrelation time[10] and the effective sample size[11] methods developed by Zuckerman and co-workers. Finally, we compare these results to the cosine content measure.[5]

2. METHODS

2.1. Block Average Root-Mean-Square Distance. We used block averaging[1] to assess convergence of the average structure. Briefly, the entire trajectory is first aligned to an optimal average structure using an iterative scheme.[7] We then divide the trajectory into contiguous blocks. We compute the average structure for each block, and the root-mean-square deviation (rmsd) between each average structure is calculated. The standard deviation for the rmsd at each block size is then plotted. The plateau, if present, indicates both the error in and the correlation time for the average structure.

2.2. Principal Component Analysis. The Cartesian coordinates for a structure at a given time point can be thought of as a $3N$ dimensional column vector, where N is the number of atoms, i.e., $[x_1, y_1, z_1, ..., x_N, y_N, z_N]^T$. The trajectory (or ensemble of structures) can then be represented by concatenating these column vectors together forming a $3N \times L$ conformation matrix A, where L is the number of snapshots in the trajectory. To compute the principal component analysis (PCA) of A, the average structure must be removed (i.e., the row-average of A is subtracted from A). The principal components are then calculated by finding the eigendecomposition of the covariance matrix AA^T, i.e., $U\Lambda U^T = AA^T$. The eigenvectors (columns of U) give the direction of fluctuations, in a least-squares sense, and the eigenvalues (diagonal elements of Λ) give the magnitude of the corresponding fluctuation. In the case of a protein, or other complete molecule, it is necessary to first remove global rotations and translations. We do this by computing the average structure from the trajectory (described above) and aligning each frame to it.

2.3. Covariance Overlap. For the proposed method to work, it is necessary to devise a mechanism to quantitatively compare two PCA results. This is typically done by comparing the subspaces (i.e., fluctuation directions for a set number of most significant modes) determined by the PCA. Here, we use the covariance overlap,[5−7] which measures not only the similarity in the directions of motion (eigenvectors) but also their relative importance (eigenvalues). This measure ranges from 0, where the fluctuations are completely dissimilar, to 1, where the fluctuations are identical. The covariance overlap between two PCA results is defined as

$$\Omega_{A,B} = 1 - \left[\frac{\sum_{i}^{N_{\text{modes}}} (\lambda_i^A + \lambda_i^B) - 2 \sum_{i}^{N_{\text{modes}}} \sum_{j}^{N_{\text{modes}}} \sqrt{\lambda_i^A \lambda_j^B} (\vec{v}_i^A \cdot \vec{v}_j^B)^2}{\sum_{i}^{N_{\text{modes}}} (\lambda_i^A + \lambda_i^B)} \right]^{1/2} \quad (1)$$

where λ_i^A is the i^{th} eigenvalue from the PCA for ensemble A and $\vec{v}_i^B$ is the i^{th} eigenvector from ensemble B.

2.4. Bootstrapping. Bootstrapping is a computational procedure used to determine the statistical error in a measurement.[12] The fundamental idea behind bootstrapping is that randomly drawing subpopulations from a distribution provides many different estimates for a statistical quantity. These estimates can then be used to estimate the error in the measure. It is particularly useful when the full distribution is unknown or complex but one has a statistical sampling of it.

In MD simulations, each structure is correlated to its neighbors in time. Bootstrapping can be used to remove this correlation from an observable. Structures are randomly drawn from the trajectory to create an ensemble and the observable calculated. This is repeated many times with the standard deviation in the distribution of averages computed becoming an estimate of the true uncertainty.

2.5. Block Covariance Overlap Method. The first step in the block covariance overlap method (BCOM) is to align the entire trajectory with an iterative alignment procedure[7] using specific atoms as reference points (in this work, nonhydrogen atoms for the small molecules and the transmembrane Cα atoms for the larger systems). Next, the conformation matrix is constructed from these atoms, and a PCA computed as described in Section 2.2. In direct analogy to block averaging, the BCOM is typically computed for a range of block sizes, up to half the size of the trajectory. Given a trajectory with L frames and a block size k, the trajectory is divided into L/k contiguous blocks. A PCA is then computed for each block. Within an individual block, there is no additional alignment performed since the trajectory as a whole is already in an optimal alignment. However, the average structure from the block is used in the subtraction for the PCA. The covariance overlap (eq 1) is then computed against the PCA for the entire trajectory. In essence, the full trajectory is treated as the gold standard. The average covariance overlap is then reported as a function of block size.

The block covariance overlap is then normalized by the value expected if the trajectory was totally uncorrelated. This value is determined by bootstrapping[13] the blocks, i.e., each block is created by randomly drawing (with replacement) samples from the entire trajectory, and the covariance overlap is computed between the PCA of the block and the overall trajectory. This

Table 1. Systems Used for Analyzing Convergence

system	length (ns)	conditions
dileucine	1000	500 K, implicit solvent
LfB6 #1	3164	323 K, explicit solvent
LfB6 #2	3127	323 K, explicit solvent
LfB6 #3	4275	323 K, explicit solvent
LfB6 #4	4275	323 K, explicit solvent
LfB6 #5	4285	323 K, explicit solvent
LfB6 #6	4285	323 K, explicit solvent
β_2AR	1023	310 K, explicit solvent and membrane
rhodopsin	1605	310 K, explicit solvent and membrane
CB2	1882	310 K, explicit solvent and membrane

procedure is repeated a set number of times (typically 50) for each block size, and the average covariance overlap is used to scale the block overlap data.

The inverse of the bootstrap-normalized blocked covariance overlap is a decaying curve that can generally be fit by a 3-exponential function, $f(t) = k_1e^{-t/t_1} + k_2e^{-t/t_2} + k_3e^{-t/t_3} + 1$, where t_1-t_3 are different correlation time constants. The 3-exponential function was determined to give the best fit to the observed data by fitting 1−4 exponential functions and examining the residual errors.

2.6. Cosine Content. The cosine content for the first mode was calculated based on the method presented by Hess.[5] In brief, the trajectory is aligned and divided into multiple contiguous blocks of a given size, as described above. The eigendecomposition is computed as previously described. The projection of the conformation matrix along the first eigenvector is then used to calculate the cosine content according to

$$c = \frac{2}{T}\left(\int_0^T \cos(\pi t/T)p(t)\mathrm{d}t\right)^2\left(\int_0^T p^2(t)\mathrm{d}t\right)^{-1} \quad (2)$$

where $p(t)$ is the t^{th} element of the projection vector and the average value over all contiguous blocks is reported.

2.7. Lightweight Object Oriented Structure Analysis Package. All analyses were performed using the Lightweight Object Oriented Structure analysis package (LOOS),[14,15] an object-oriented library for creating new analytical tools for MD that is implemented in C++ . LOOS uses Boost[16] and atlas[17,18] for additional functionality and high-performance linear algebra calculations. LOOS includes a powerful "selection expression" parser that enables tools to easily select which atoms to operate on. In addition, LOOS provides support for reading the native file formats for most major MD packages, including CHARMM,[19] NAMD,[20] Amber,[21] Gromacs,[22] and Tinker.[23] Also included with LOOS are over 50 analytical tools, including suites of programs for computing elastic network model solutions and the convergence analyses presented in this work. LOOS is freely available from SourceForge (http://loos.sourceforge.net).

2.8. Structural Decorrelation Time and Effective Sample Size. For comparison purposes, we also applied two previously published methods developed by Zuckerman and co-workers for assessing the convergence of MD simulations, which we believe represent the state of the art in the field. The first method uses the "decorrelation time" as described by Lyman et al.[10] Briefly, this method partitions the conformational space of the trajectory using a set of randomly drawn "reference" structures

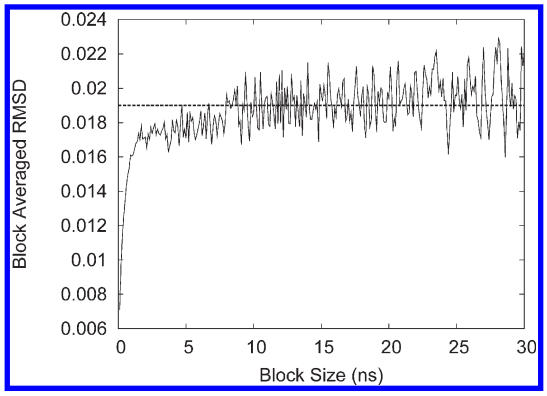

Figure 1. Block averaging of the rmsd between each block and the optimal global average for the 1 μs dileucine trajectory.

and compares the variance of the histograms of subsets of the ensemble (using the reference structures) with the expected variance if the structures were uncorrelated. The second method, described by Zhang et al.,[11] uses the same tesselation of configuration space but then clusters the reference "bins" based on the rates of exchange between the different clusters. This gives an estimate for the number of uncorrelated conformations found in the ensemble from which a decorrelation time can be estimated. Implementations of both tools are available from the Zuckerman lab Web site (http://www.ccbb.pitt.edu/Faculty/zuckerman/software.html) and are also available in the convergence suite in LOOS.[14]

2.9. Model Systems. Several different systems were used to assess convergence, ranging from a dipeptide to a set of G protein-coupled receptors, and are listed in Table 1. The first system was a "toy" system based on Lyman et al.,[10] consisting of a dileucine in implicit water at 500 K simulated in Tinker[23] for 1 μs, with structures saved every 100 ps . The second system is a short hexapeptide derived from lactoferrin B in explicit water;[24,25] NAMD[20] was used on a BlueGene/P[26] to simulate this peptide in the NVT ensemble at 323 K using CHARMM22[19,27,28] with CMAP[28] parameters. Snapshots were saved every 1 ns. We ran multiple simulations, with 2 lasting approximately 3.1 μs each and 4 other simulations that are each approximately 4.3 μs long. For both the LfB6 and the dileucine systems, all heavy atoms were used in performing alignments and for computing the PCA.

Three different G protein-coupled receptors (GPCRs) were used as the larger test systems. The details of their construction have been previously described, but in brief, each consisted of the GPCR embedded in a lipid bilayer along with explicit solvent. The first system is a 1.02 μs all-atom simulation of β_2AR,[29] the second is a 1.6 μs simulation of dark-state rhodopsin,[30] and the final system is an approximately 1.9 μs simulation of the CB2 canabannoid receptor.[31] For these larger systems, only transmembrane α carbons were considered for analysis.

3. RESULTS AND DISCUSSION

3.1. Convergence of the Average Structure. As a trajectory evolves over time, the estimate for the average structure will change. This can be the result of new conformational substates being found or from changes in the relative population of the existing substates. This variation is particularly important for methods such as PCA that depend upon the average structure as a reference point. This dependence can be easily demonstrated

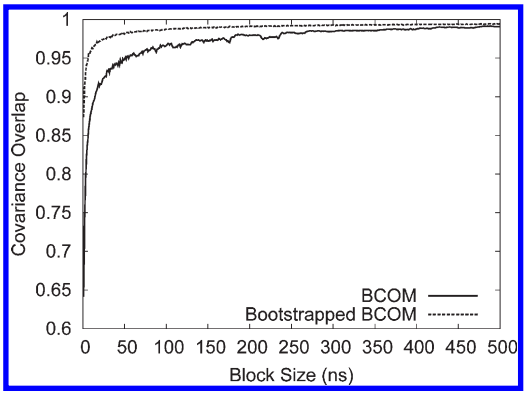

Figure 2. BCOM and BBCOM for the 1 μs dileucine trajectory.

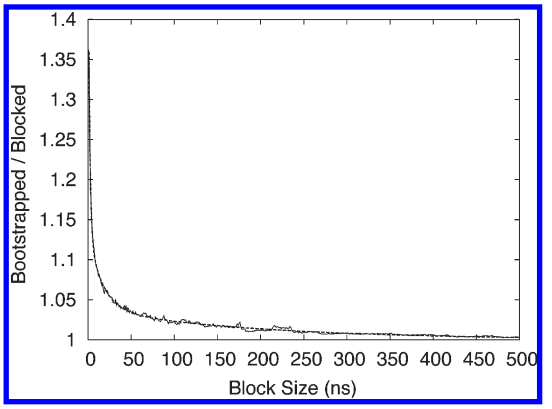

Figure 3. Inverse of bootstrap-normalized BCOM for the 1 μs dileucine trajectory. A three-exponential function is fit to the curve giving three different correlation times. The correlation times are t_1 = 1.7 ns, t_2 = 16.0 ns, and t_3 = 194.3 ns.

by randomly perturbing the average structure used in computing a PCA and then calculating the covariance overlap between the perturbed and the unperturbed modes. An rms perturbation as small as 0.1 Å in the β_2AR average structure reduces the covariance overlap from 1 to 0.92 and an overlap of 0.94 for rhodopsin and CB2. Therefore, we must first ask at what point is it reasonable to believe that the average structure has converged such that its statistical errors no longer perturb the PCA results significantly. For the case of the 1 μs dileucine simulation, Figure 1 shows the block averaged rmsd. The rmsd error in the average structure does not plateau until around 10 ns with an error of 0.019 Å . This is a higher uncertainty than would be suggested based on a plot of the running average rmsd (≈0.002 Å) for the simulation (see Figure SI1A, Supporting Information). The running average rmsd for the transmembrane Cα's of rhodopsin is also shown in Figure SI1B, Supporting Information. It appears to show a convergence at approximately 1.4 μs with an error of 0.025 Å . It is important to note however that the rmsd beyond the first several hundred nanoseconds is quite small in both cases, and it would be tempting to determine convergence of the average structure at this point. The rmsd between each structure of the rhodopsin simulation and the starting crystal structure is shown in Figure SI2, Supporting Information. Here, there is a short plateau at 0.5 μs where the rmsd does not increase, followed by an increase and a longer plateau at about 1 μs. This "classical" measure of convergence therefore suggests that the simulation has "converged" by 1 μs.

3.2. Convergence and Correlation Times. The PCA of an MD trajectory is defined by both the fluctuations of the system as well as how well these are sampled. To illustrate this, imagine a simple model system that has two distinct states. If the system stays in one well, then the average structure and the fluctuations about that average will be different from the case where the system transitions between the two wells. Similarly, once the simulation samples the true statistical distribution of states (e.g., twice as many samples in state one as in state two), then the average structure and the fluctuations about that average will be different still. Only once the fluctuation subspaces cease to change, can we consider the simulation well sampled.

The covariance overlap (eq 1) is a powerful tool for determining how similar the conformational spaces sampled by two trajectories are. In contrast to the subspace overlap,[32] which only considers the similarity of directions, the covariance overlap also includes the relative significance (i.e., power) of each mode. The covariance overlap also considers all modes, rather than an arbitrary subset. This is a more stringent test on whether the subspaces are similar, since it requires a better sampling of the underlying fluctuations in order for both the directions and the power spectra to match.

If we wish to say that a trajectory has "converged," then it is reasonable to expect that the fluctuations from different, large subsets of the trajectory should be similar, i.e., have a covariance overlap that approaches 1. The point at which convergence has been achieved can then be determined by choosing successively smaller subsets and determining at what point the covariance overlap diverges from 1. In practice, the subsets are picked as contiguous blocks of a given size spanning the trajectory. This method is what we call the BCOM. It is, in effect, a quantification of the statistical error present in the MD simulation due to finite and discrete sampling. The overlap for each block is then normalized by the overlap value expected were there no correlations via bootstrapping, and the inverse of the resulting curve is fit to a three exponential function. As the block sizes increases, this ratio will decay toward 1, where the blocks are long enough that they are effectively uncorrelated.

Hess[5] notes that autocorrelation functions of principal components can be fitted with a double exponential function using a fast and a slow correlation time. Indeed, such a hierarchy has been seen previously in PCA analysis for large systems and manifests as a "beads on a string" when visualizing the fluctuation phase space.[29,33] Within a bead, or conformation state, there is a short correlation time as the local well(s) are explored. The transition between different beads occurs at a much slower time scale. In contrast, our method results in three different correlation times, typically at different scales (i.e., fast, medium, and slow scale). While we could hypothesize that the third time scale is the time to get a sampling of the distribution of states, we have no underlying model to justify a triple exponential; fitting with two exponentials leaves a clear residual, while the residual from triple exponential appears random.

The covariance overlap from the BCOM and the bootstrapped BCOM (BBCOM) for the dileucine system is shown in Figure 2. Both the BCOM and the BBCOM approach 1 after 500 ns. The BCOM curve rapidly increases to 0.95 around 50 ns and then slowly increases thereafter. The bootstrapped curve, in contrast, is very close to its peak value for random sets containing as many points as 25 ns blocks. These curves illustrate the effect of correlation within the blocks. The ratio of the BBCOM to the

Table 2. Fitting of Three-Exponential Curve to BCOM Results for Different Model Systems

model	k_1	t_1 (ns)	k_2	t_2 (ns)	k_3	t_3 (ns)
dileucine	0.42	1.7	0.10	16.0	0.04	194.3
LfB6	0.19	32.9	0.17	199.9	0.12	1316.3
β_2AR	0.76	23.3	0.51	248.9	0.19	2481.2
rhodopsin	2.18	10.0	0.89	47.0	1.24	804.5
CB2	1.69	7.3	0.98	39.1	1.15	934.7

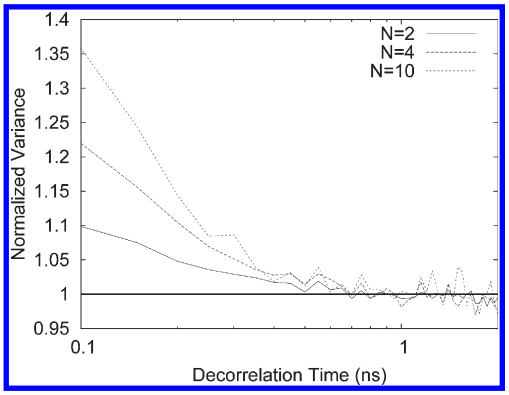

Figure 4. Decorrelation time estimation by plotting $\sigma^2_{obs}(t)$ for three different step sizes: 2, 4, and 10. The time point where the curves reach 1 is the approximate decorrelation time.

Table 3. Decorrelation Times As Estimated by Variance Plots (τ_d) and from Automated Effective Sample Size Analysis (τ'_d) [a]

model	τ_d (ns)	τ'_d (ns)
dileucine	0.7−0.9	0.45−0.53
LfB6	40−55	35.6−51.0
β_2AR[b]	75−90	37.2−46.7
rhodopsin[b]	130−150	60.3−70.9
CB2[b]	140−175	76.7−91.2
β_2AR[c]	90−100	60.5−78.2
rhodopsin[c]	140−180	120.0−150.5
CB2[c]	170−230	160.8−188.7

[a] All analyses used 20 replicates. The τ_d for dileucine and LfB6 used step sizes of 2, 4, and 10, while all other models used 2, 3, and 4. [b] 20 bins were used and not all clustering for τ'_d resulted in 2 top-level clusters. [c] 10 bins were used and not all clustering for τ'_d resulted in 2 top-level clusters.

BCOM is shown in Figure 3, along with the best three-exponential fit. The correlation times, determined from the three-exponential fit, are shown in Table 2. It is quite clear from this graph that the dileucine system has converged since the ratio decays to 1 by 500 ns. What is striking in this figure is the magnitude of the longer time scale for such a simple system—nearly 200 ns. This is a measure of the simulation time required to accurately determine the relative populations of the different states available to dileucine. It is worth noting that this time is far longer than the average lifetime of any single state. Rather, it reflects the fact that even a molecule as simple as dileucine has a significant number of states available to it.

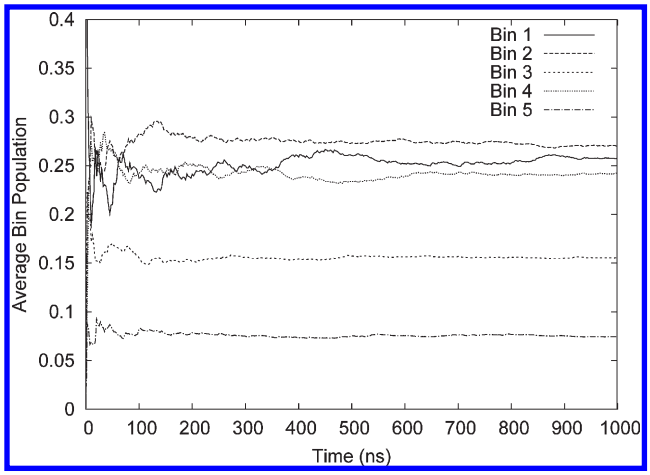

Figure 5. The running average of the population for each bin over time using 5 bins for the dileucine simulation.

In order to further assess the quality of sampling, we also performed a decorrelation time and an effective sample size analysis, as described in Section 2.8. The normalized variance versus decorrelation time is shown in Figure 4 for three different step sizes through the trajectory. Each curve converges to 1 at ≈0.7−0.9 ns. Similarly, the decorrelation time estimated from the effective sample size (also described in Section 2.8) is 0.45− 0.53 ns. The decorrelation times and their estimates for each system are shown in Table 3. It is important to remember that these two methods are not measuring the same thing as the BCOM; both of these methods hinge on the rate of interconversion between individual states, which BCOM measures as the convergence of their populations. Examining the pairwise (all-to-all) rmsd map (Figure SI3A, Supporting Information), it is apparent that dileucine has substates that last approximately 1 ns. Indeed, the pairwise rmsd map is a very simple test that can be used to give a qualitative assessment of how well sampled the trajectory is.[2,29] However, the off-diagonal blocks, despite having a low rmsd, are not guaranteed to be structurally similar.

While the decorrelation time (or the rate of transitions between states) can be quite rapid, the convergence of the distribution of observables may take much longer. For example, the dileucine trajectory can be partitioned as above but using 5 bins. A running average of the bin populations is computed and graphed over time in Figure 5. The majority of the changes in populations have smoothed out in the first 200 ns or so, with some variation in bins 1 and 4 that continue until nearly 1 μs.

3.3. LfB6. The LfB6 hexapapetide (RRWQWR-NH_2) is a slightly larger model system that was simulated in explicit solvent. As with the dileucine model, all nonhydrogen atoms were considered in the analysis. The average inverse bootstrap-normalized BCOM plot for all 6 LfB6 simulations is shown in Figure 6 with the fit parameters listed in Table 2. There is a short correlation time of 33 ns with longer correlation times of 200 ns and 1316 ns. When fit individually, half of the simulations have long correlation times exceeding the largest block size. This accounts for the variation in the BCOM plot at 1 μs and beyond and is evidence of a lack of convergence despite being a small system that was simulated for far longer than is typical for full proteins.

The decorrelation time plots for LfB6 are shown in Figure 7. The curves for the different step sizes are averaged over all six simulations. This figure yields an estimated decorrelation time of

40− 55 ns. The average decorrelation time derived from the effective sample size analysis (using 20 replicates) is approximately the same, 35.6− 51 ns. This time scale also matches the short correlation time from the BCOM analysis.

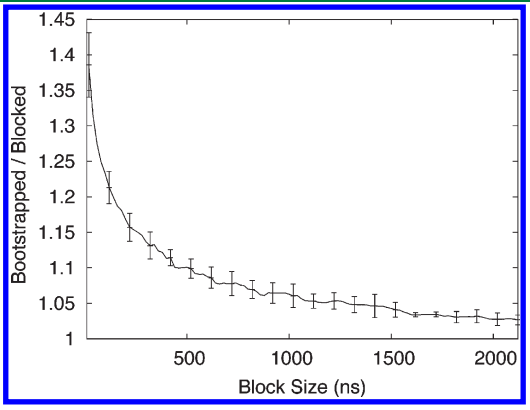

Figure 6. The inverse of the averaged, bootstrap-normalized BCOM plot for all six LfB6 simulations. The error bars are the standard deviation of the normalized BCOM from all simulations. The correlation times are t_1 = 32.0 ns, t_2 = 199.9 ns, and t_3 = 1316.3 ns. The t_1 correlation time closely matches the decorrelation time estimated by other methods.

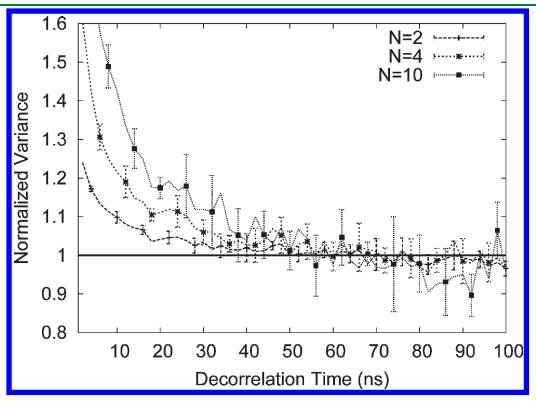

Figure 7. Decorrelation time estimation from $\sigma^2_{obs}(t)$ for all 6 LfB6 simulations averaged together. The error bars are the standard deviation across all six simulations. The decorrelation time is estimated as being between 40 and 55 ns.

Figure 8A shows the running average of the cluster populations for a 20 bin partitioning of the first LfB6 trajectory. Here, the bins are across the x axis with time along the y axis. Color indicates the deviation of the running average from the overall average population for each column. Qualitatively, what we find is that the cluster populations continue changing until approximately 1−1.5 μs, although there are again variations up to the full 3 μs of the simulation. The time scale of the convergence of these cluster populations corresponds to the long time scale predicted by the BCOM analysis of 1.3 μs.

3.4. GPCRs. The three GPCR simulations represent an application of these methods to large systems of biological interest, with state of the art trajectories ranging from 1 to nearly 2 μs. In order to make the convergence criterion more lenient, only the transmembrane α carbons were considered in the analysis, since incorporating the fluctuations of the flexible loops and the terminii would vastly expand the configuration space of the systems. The BCOM curve and fit for β_2AR is shown in Figure 9A. The best exponential fit to the data is again a three-exponential giving three well-separated correlation times of approximately 23 ns, 250 ns, and 2.5 μs. It is important to note that the longest correlation time, 2.5 μs, is longer than the largest block size used in the analysis. Moreover, the final ratio is above 1.2, indicating that the longest blocks still do not appear uncorrelated, which in turn suggests that the system is poorly converged. Interestingly, a previous analysis of the phase space formed by the first three principal components for β_2AR[29] found the presence of "beads", indicative of conformational substates, with an average duration of 252 ns. This time scale is virtually identical to the medium time scale found by the BCOM.

The decorrelation time and effective sample sizes for β_2AR are shown in Table 3. In this case, the analyses were repeated using two different numbers of bins for the partitioning of configuration space: 10 and 20 bins. There were an insufficient number of frames to support larger step sizes (i.e., N = 10) in the decorrelation time plots, so step sizes 2−4 were used instead. The estimated decorrelation times for β_2AR are 90− 100 ns for 10 bins and 75− 90 ns for 20 bins. The effective sample size hierarchical clustering did not generally result in two top-level states (this can occur in this analysis when there are states that do not interconvert during the trajectory, which is itself a sign that the trajectory has not converged). In addition, the effective sample size (N_{eff}) per bin was only slightly greater than 1, suggesting that the system is not converged and that the resulting

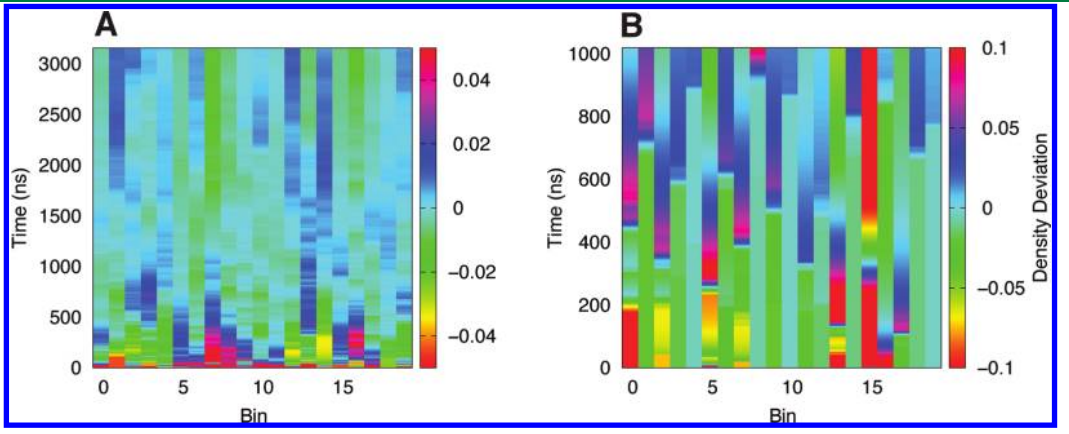

Figure 8. A visualization of the change in cluster populations over time for the first LfB6 trajectory (panel A) and for β_2AR (panel B). Color represents the deviation from the column average.

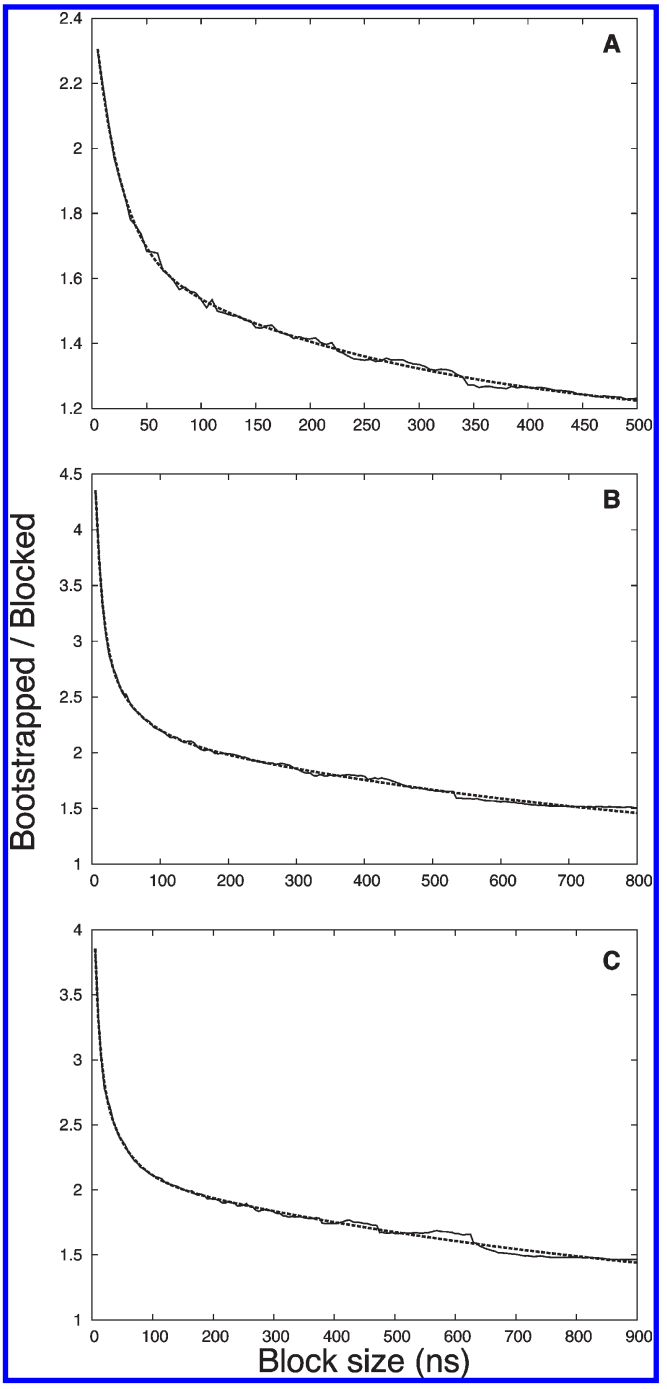

Figure 9. Inverse of the bootstrap-normalized BCOM for the simulation of β_2AR (panel A), rhodopsin (panel B), and CB2 (panel C). The longest correlation times are 2481, 804, and 935 ns, respectively, and are approximately equal or greater than the largest block size, indicating a lack of convergence.

statistics are suspect. The estimated decorrelation times based on N_{eff} are 60− 78 and 37− 47 ns for 10 and 20 bins, respectively. The running average of the cluster populations for β_2AR is shown in Figure 8B; many of the bins show sharp population changes well into the simulation, such as bin 1 at nearly 700 ns, while bin 5 continues to drift until the 1 μs mark, and bin 8 shows a sudden change at almost 1 μs. The extent of convergence for CB2 can be seen in the all-to-all rmsd plot in Figure SI3B, Supporting Information. The only significant cross-peak, indicating revisiting of conformational substates, occurs at around 1.4 μs, although there are broad regions with some self-similarity, such as the first 0.4 μs and 0.6− 1.1 μs. In addition, one can qualitatively discern small blocks along the diagonal whose size is approximately 100 ns. This again reinforces the challenges of adequately sampling a large protein system, even with start of the art, microsecond-scale simulations.

The BCOM, decorrelation time, and N_{eff} for rhodopsin and CB2 are also given in Tables 2 and 3. In each case, the BCOM curve never comes close to 1. Rhodopsin plateaus at approximately 1.7 (Figure 9B), and CB2 reaches about 1.6 (Figure 9C), indicating neither system is converged. Moreover, the long correlation time exceeds the largest block size used in the analysis, further suggesting that the systems have not converged. The decorrelation time and N_{eff} analyses are somewhat more complicated given the differences between 10 and 20 bins. While the change in decorrelation time is not that significant, the N_{eff} varies considerably. In the 10 bin case, the number of top clusters found is smaller than in the 20 bin case, and it is likely that the larger numbers are more reflective of the "true" decorrelation time. Nevertheless, the decorrelation times for all of the GPCRs are quite long, approaching 100 ns or longer.

While the time scales found by the BCOM and the decorrelation and N_{eff} analysis diverge for the GPCR systems, all methods indicate that there are quite long time scales involved and that the number of statistically independent configuration samples, even in a multimicrosecond simulation, is small. This divergence is also not entirely unexpected considering that the BCOM is using a very different approach from the decorrelation time and N_{eff} analysis. Moreover, since the per bin N_{eff} is very close to 1, indicating insufficient sampling quality, the estimated decorrelation times are suspect.

3.5. Cosine Content. The cosine content for contiguous blocks along the first mode is shown in Figure 10. The average cosine content across all blocks of a given size is plotted, and the error bars are the standard deviation. Figure 10A illustrates the cosine content for a converged simulation, the first LfB simulation. The cosine content reaches 0 around 1 μs, suggesting convergence on a slightly longer time-scale than determined by BCOM (≈900 ns, data not shown). In contrast, Figure 10B shows the cosine content for the rhodopsin simulation, a nonconverged simulation. The cosine content begins high and increases with larger block sizes. Hess suggests that, in practice, the cosine content can be a useful negative indicator of conformational sampling.[5] In a simpler form, the cosine content of the first mode for the entire trajectory can be a simple test to determine whether a system is undersampled. For example, the cosine content for β_2AR, rhodopsin, and CB2 are 0.78, 0.87, and 0.9, respectively. In contrast, the average cosine content for the LfB simulations is 0.003.

3.6. Known Unknowns: Suggestion for Practical Applications of BCOM. There are two tantalizing questions for the BCOM analysis: How reliably can it indicate when a system is not converged, and can it predict how much longer is required to run until convergence is achieved? Philosophically, this is a difficult prospect since it is difficult to know what is not known. It is always possible that increasing the simulation time will reveal a new conformation state, and there is no way to know that this state exists solely by using the previously seen configuration space. In tests where the trajectories were arbitrarily truncated and the BCOM performed, the reliability of using the long

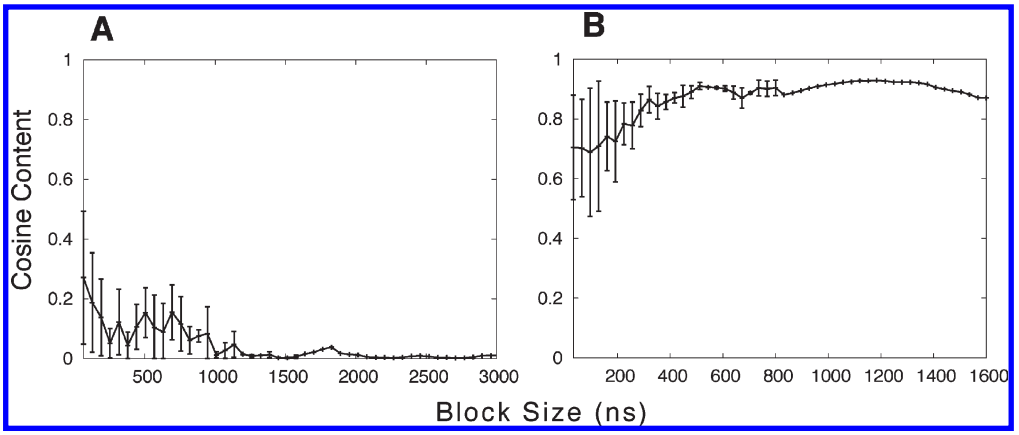

Figure 10. The average cosine content as a function of block size for the first LfB simulation (panel A) and rhodopsin (panel B). The error bars are the standard deviations across all contiguous blocks of a given size.

time-scale coefficient and the BCOM ratio is low until at least the average structure begins to converge. Indeed, when the simulation is so small that it stays within a single well, it may in fact be well sampled within this restricted conformation space—in effect, convergence is in the eye of the beholder. An alternative interpretation is that if the simulation is far too short, then there is no way to know this. Once the trajectories become sufficiently large, however, more credence can be given to these indicators. This is not conclusive proof, however. In some cases, the trajectory was truncated before a new state was discovered, resulting in an apparent convergence (data not shown). In some respects, however, the BCOM was not incorrect in that given the known configuration space, the simulation did appear to be converged. However, once the trajectory was extended and the new state was seen, it was apparent that the simulation was not close to convergence. It bears repeating that there is no way to prove a simulation (or even an ensemble of simulations) has converged to the correct answer. Rather, one can only demonstrate that the system has not converged or that it may be converged.

It is important to bear in mind that none of these methods replace the simpler methods of assessing the simulation's sampling, such as visually inspecting the all-to-all rmsd of the system, examining the convergence of the average structure, and the time series of observables of interest. While it has been repeatedly shown that these measures are not necessarily indicative of good sampling quality, they are a simple test that represents a minimum threshold that any simulation must pass before more sophisticated methods are employed. Similarly, much can be learned by plotting the projection of the system along the first several principal components, and this is typically a computationally inexpensive analysis.

4. CONCLUSIONS

We have devised a new method for assessing both the quality of sampling and the rate of convergence for a molecular dynamics simulation. The method relies on the similarities of the subspace sampled by the simulation and defined by the system's fluctuations. This method differs from other approaches in that it relies on fitting the resulting curves to a three-exponential function rather than a graphical interpretation. In addition, more information is considered in the analysis by using the covariance overlap, utilizing a wide range of block sizes for partitioning the trajectory and normalizing the resultant overlaps by a bootstrapped block sample. Moreover, the hierarchical nature of the different correlation times indicates a longer time scale for sufficient sampling of the "known" configuration space than is suggested by other measures. Armed with this knowledge of the correlation times in the simulation, we can now make assertions regarding the effective sample size for any observable statistics and hence the statistical errors that are present in those quantities.

All of the methods investigated in this work suffer when the simulation is far too short. That is, they require a minimum sampling quality before one can hope to determine how well converged or sampled the system is. Even in the case where the simulation is converged, there is appreciable variation in the numbers obtained for the correlation and decorrelation times, although the general time scales are similar. Absent a priori knowledge of what the fluctuation space should look like, there is no method known to the authors that can determine an "unknown unknown," that is, states that should have been seen but have not yet been visited by the system. Moreover, many of these methods for assessing sampling quality are new, and it is not yet clear under what conditions they perform well and those under which they fail. Given this, we strongly recommend that all available methods, ranging from the simple rmsd plots and the cosine content along the first few modes to the structural decorrelation analyses and BCOM, be used to assess sample quality and convergence. These tests are computationally inexpensive (particularly relative to the cost of running an all-atom simulation), and implementations are freely available as part of LOOS. Only in concert, combining N_{eff} with BCOM for example, can we hope to assert what the statistical error is and whether or not a simulation is well sampled. We must emphasize again that despite the improvements in these methods for assessing sampling quality, there is no substitute for visually observing the time series, be it torsions or projections along a principal component to look for multiple transitions.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information. Three figures: The first shows the rmsd between the running average structure for time t and $t + 1$ for dileucine and rhodopsin. The second shows the rmsd between the structure at time t and the starting crystal structure for rhodopsin. The final figure shows the all-to-all rmsd for each structure in the dileucine simulation with every other structure in

the same simulation as well as for the CB2 simulation. This information is available free of charge via the Internet at http://pubs.acs.org/.

AUTHOR INFORMATION

Corresponding Author

*E-mail: alan_grossfield@urmc.rochester.edu.

ACKNOWLEDGMENT

We would like to thank Dan Zuckerman for his insightful discussions and reading of this manuscript. We would also like to thank IBM Watson for the BlueGene supercomputer and Michael Pitman for the GPCR simulations. We are also grateful to the Center for Research Computing at the University of Rochester for providing computing systems and personnel to enable the research presented in this manuscript.

REFERENCES

(1) Flyvbjerg, H.; Petersen, H. *J. Chem. Phys.* **1989**, *91*, 88–103.
(2) Grossfield, A.; Zuckerman, D. M. *Ann. Rep. Comp. Chem.* **2009**, *5*, 23–48.
(3) Balsera, M.; Wriggers, W.; Oono, Y.; Schulten, K. *J. Phys. Chem.* **1996**, 2567–2572.
(4) Amadei, A.; Ceruso, M. A.; Nola, A. D. *Proteins* **1999**, *36*, 419–24.
(5) Hess, B. *Phys. Rev. E* **2002**, *65*, 031910.
(6) Faraldo-Gómez, J. D.; Forrest, L. R.; Baaden, M.; Bond, P. J.; Domene, C.; Patargias, G.; Cuthbertson, J.; Sansom, M. S. P. *Proteins* **2004**, *57*, 783–91.
(7) Grossfield, A.; Feller, S. E.; Pitman, M. C. *Proteins* **2007**, *67*, 31–40.
(8) Sullivan, D. C.; Kuntz, I. D. *Proteins* **2001**, *42*, 495–511.
(9) Smith, L. J.; Daura, X.; van Gunsteren, W. F. *Proteins* **2002**, *48*, 487–96.
(10) Lyman, E.; Zuckerman, D. M. *J. Phys. Chem. B* **2007**, *111*, 12876–82.
(11) Zhang, X.; Bhatt, D.; Zuckerman, D. M. *J. Chem. Theory Comput.* **2010**, *6*, 3048–3057.
(12) Efron, B. *Ann. Stat.* **1979**, *1*, 1–26.
(13) Efron, B.; Tibshirani, R. J. *An Introduction to the Bootstrap*; CRC Press LLC: Boca Raton, FL, 1998; pp 45−57.
(14) Romo, T. D.; Grossfield, A. *Conf. Proc. IEEE Eng. Med. Biol. Soc.* **2009**, *1*, 2332–5.
(15) Romo, T. D.; Grossfield, A. *LOOS: Lightweight Object Oriented Structure analysis*; Grossfield Lab, University of Rochester Medical School: Rochester, NY; http://loos.sourceforge.net. Accessed May 20, 2011).
(16) *BOOST C++ Libraries*; http://www.boost.org.
(17) Whaley, R. C.; Dongarra, J. In Proceedings from Ninth SIAM Conference on Parallel Processing for Scientific Computing, San Antonio, TX, March 22−24, 1999; SIAM: Philadelphia, PA, 1999; (in CD-ROM).
(18) Whaley, R. C.; Petitet, A. *Software: Practice and Experience* **2005**, *35*, 101–121.
(19) Brooks, B.; Bruccoleri, R.; Olafson, B.; States, D.; Swaminathan, S.; Karplus, M. *J. Comput. Chem.* **1983**, *4*, 187–217.
(20) Phillips, J. C.; Braun, R.; Wang, W.; Gumbart, J.; Tajkhorshid, E.; Villa, E.; Chipot, C.; Skeel, R. D.; Kalé, L.; Schulten, K. *J. Comput. Chem.* **2005**, *26*, 1781–802.
(21) Case, D. A.; Cheatham, T. E.; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. *J. Comput. Chem.* **2005**, *26*, 1668–88.
(22) Spoel, D. V. D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. C. *J. Comput. Chem.* **2005**, *26*, 1701–18.
(23) Ponder, J. *Tinker*, version 4.2; Department of Biochemistry and Molecular Biophysics, Washington University School of Medicine: St. Louis, MO, 2007.
(24) Tomita, M.; Takase, M.; Bellamy, W.; Shimamura, S. *Acta Paediatr. Jpn.* **1994**, *36*, 585–91.
(25) Romo, T. D.; Bradney, L. A.; Greathouse, D. V.; Grossfield, A. *Biochim. Biophys. Acta, Biomembr.* **2011**, *1808*, 2019–30.
(26) Allen, F.; et al. *IBM Syst. J.* **2001**, *40*, 310.
(27) MacKerell, A. D., Jr.; Brooks, C. L., III; Nilsson, L.; Roux, B.; Won, Y.; Karplus, M. *CHARMM: The Energy Function and Its Parameterization with an Overview of the Program*; John Wiley and Sons: Chichester, 1998; Vol. 1; pp 271−277.
(28) A. D. MacKerell, J.; Feig, M.; C. L. Brooks, I. *J. Comput. Chem.* **2004**, *25*, 1400–15.
(29) Romo, T. D.; Grossfield, A.; Pitman, M. C. *Biophys. J.* **2010**, *98*, 76–84.
(30) Grossfield, A.; Pitman, M. C.; Feller, S. E.; Soubias, O.; Gawrisch, K. *J. Mol. Biol.* **2008**, *381*, 478–86.
(31) Hurst, D. P.; Grossfield, A.; Lynch, D. L.; Feller, S.; Romo, T. D.; Gawrish, K.; Pitman, M. C.; Reggio, P. H. *J. Biol. Chem.* **2010**, *285*, 17954–17964.
(32) Hess, B. *Phys. Rev. E* **2000**, *62*, 8438–48.
(33) Clarage, J. B.; Romo, T.; Andrews, B. K.; Pettitt, B. M.; Phillips, G. N. *Proc. Natl. Acad. Sci. U.S.A.* **1995**, *92*, 3288–92.