JCTC Journal of Chemical Theory and Computation

ARTICLE

pubs.acs.org/JCTC

On-the-Fly Identification of Conformational Substates from Molecular Dynamics Simulations

Arvind Ramanathan,[†,§] Ji Oh Yoo,[‡] and Christopher J. Langmead*[,†,‡]

[†]Lane Center for Computational Biology, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, United States

[‡]Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, United States

Supporting Information

ABSTRACT: We recently introduced a new method for discovering, characterizing, and monitoring spatiotemporal patterns in the conformational fluctuations in molecular dynamics simulation data (*J. Comput. Biol.* **2010**, *17* (3), 309−324). Significantly, our method, called Dynamic Tensor Analysis (DTA), can be performed as the simulation is progressing. It is therefore well-suited to analyzing long timescale simulations, which are critical for studying biologically relevant motions but may be too large for traditional analysis methods. In this paper, we demonstrate that the patterns discovered by DTA often correspond to functionally important conformational substates. In particular, we apply DTA to a 150 ns simulation of ubiquitin and discover patterns that provide unique insights into ubiquitin's ability to bind multiple substrates. Moreover, we take advantage of DTA's ability to identify patterns on different timescales and investigate how fast positional fluctuations may modulate slower, large-scale motions in functionally important regions. Our findings here suggest that DTA is well-suited to organizing, visualizing, and analyzing very large trajectories and discovering conformational substates.

INTRODUCTION

Proteins are intrinsically dynamic and exist in an ensemble of interconverting conformations. This ensemble can be partitioned into subsets of conformations, called *conformational substates*,[1,2] with similar structures and internal energies. The study of these conformational substates and their relevance to biological function remains an active area of research.[3−5] Multiple experimental techniques have been used to demonstrate the presence of conformational substates in proteins, including nuclear magnetic resonance (NMR),[6] neutron spectroscopy,[7,8] and X-ray crystallography.[5] These conformational substates, in turn, have provided valuable insights into biological function, such as enzyme catalysis[5,9,10] and protein folding.[11] However, experimental techniques are presently incapable of resolving many of the structural and dynamical details of individual conformational substates, or explaining how proteins transition between them.

Theoretical and computational modeling can effectively enhance our understanding about conformational substates by providing detailed atomistic information about their structural and dynamical features.[3] Molecular dynamics (MD) and Monte Carlo (MC) simulations are perhaps the most commonly used computational techniques for characterizing conformational dynamics in proteins.[12] Recent advances in hardware and software (e.g., refs 13−19) have dramatically decreased the cost of MD and MC simulations, and it is now possible to investigate conformational dynamics on microsecond and millisecond timescales. Such long simulations are well-suited to identifying and studying the conformational substates relevant to biological function. At the same time, the concomitant increase in the size of the resulting data sets make them more challenging to analyze and interpret. Thus, there is a need for new, automated methods that help biologists identify and characterize conformational substates, and the transitions between them.

Several techniques have been developed in order to identify conformational substates in MD data. These techniques are usually performed in an *offline* fashion (i.e., after all of the data are collected) and produce low-dimensional models of the data. Principal Component Analysis (PCA)[20−22]-based analyses, for example, have been used to describe conformational substates and their relevance in both folding pathways[23]and ligand/substrate binding.[24] More recently, Lange and Grubmüller introduced full correlation analysis,[25] which can capture both linear and nonlinear correlated motions from MD simulations. Another approach to characterizing conformational substates is the use of Markov State Models (MSMs), which partitions the conformational space sampled by the simulation into kinetically accessible substates.[26] Note that while each of these approaches detect and characterize conformational substates, they are performed only after simulations have completed. This is important because any analysis algorithm that runs in time polynomial in the number of simulation frames (e.g., clustering[27−29]) will face a serious computational challenge when presented with long timescale simulation data.

To overcome the aforementioned limitations, we have recently introduced an *online* algorithm to monitor and characterize collective distance fluctuations in protein simulations as they are progressing.[30,31] This algorithm, which performs *dynamic tensor analysis* (DTA),[32] represents the MD simulation trajectory as a time-evolving stream of multidimensional tensors. We have previously shown[30,31] that DTA can (a) identify constrained and flexible regions in a protein, (b) characterize the conformational couplings that exist between different parts of a protein, and (c) detect time points where collective behavior may have significantly changed. Our method is also flexible enough to allow the end-user to track specific structural features such as hydrogen bonds or hydrophobic interactions as they vary over time and to detect collective behavior as simulations are progressing.

Received: September 17, 2010
Published: February 10, 2011

In this paper, we will demonstrate the ability of DTA to identify and characterize conformational substates in an online fashion by applying it to a 150 ns simulation of protein ubiquitin. We compare and contrast the conformational substates discovered from our technique to those identified using simple metrics (e.g., root-mean squared deviations [RMSD]) and offline PCA-based methods. Dynamic tensor analysis reveals the presence of several well-defined conformational substates that are not directly evident from RMSD-based metrics. The conformational substates we identify are distinct in their collective fluctuations and directly relevant to substrate binding. Finally, we will demonstrate that it is possible to analyze the conformational landscape on different timescales using DTA. Our multiscale analysis of the ubiquitin trajectory provides unique insights into how the motions of the binding regions may be modulated to achieve optimal binding conformation. These experiments demonstrate the utility of DTA as an approach for identifying and characterizing conformational substates from molecular dynamics simulation data.

METHODS

Molecular Dynamics Simulations. Detailed molecular dynamics simulations were performed on human ubiquitin (PDB id: 1UBQ). The initial crystal structure was processed using the Maestro software (Schrodinger Inc.), and the OPLS/AA forcefield[33,34] was used for simulations. The protonation state of each residue was determined at pH 7.0, and missing protons were added. The structure was then placed in a pre-equilibrated rectangular box of water, parametrized using SPC,[35] such that the distance between the protein and the boundaries of the box was at least 10 Å. The final box size was 51.3 Å × 51.3 Å × 51.3 Å. Prior to equilibration, the solvent and proteins were energy minimized using both steepest descent (50 steps) and a conjugate gradient until the overall root-mean-square (RMS) of the gradients was less than 0.25 kcal/mol/Å. The system was then equilibrated using a standard protocol involving multiple steps of energy minimization and small MD runs to allow the solvent molecules and then the solute atoms to relax. Temperature ramps were used to gradually bring the system to 300 K. Next, an NPT (constant number of particles N; constant pressure P; constant temperature T ensemble) simulation at 300 K was performed to make sure that the system was stable. A small simulation run for 1.2 ns using an NVE (constant number of particles N; constant volume V; constant energy E) ensemble was then performed to allow the system to fully equilibrate.

The production run was performed using Desmond,[36] under an NVE ensemble with periodic boundary conditions. The RESPA integrator was used for solving Newton's laws of motion. Hydrogen atoms were constrained via the SHAKE algorithm.[37] Long-range electrostatics were computed using the particle mesh ewald (PME)[38] technique. The production run lasted a total of 150 ns (excluding the initial 1.2 ns run), with frames being stored every 10 ps. A total of 15 000 snapshots were stored and used for further analysis.

Dynamic Tensor Analysis for Protein Simulations. Recently, our group introduced a novel online analysis tool for mining spatiotemporal patterns from MD simulations.[30,31] Our technique encodes the MD trajectory as a time-ordered sequence of tensors. Tensors are a generalization of matrices beyond two dimensions and can be used to encode both spatial and temporal dynamics. In contrast, PCA takes as input a covariance matrix, which is a time-averaged representation of the data. Our method then performs Dynamic Tensor Analysis (DTA) to identify and track spatiotemporal patterns in the data. DTA was first introduced in the context of analyzing streams of router data from computer networks.[32,39]

A detailed description of the algorithm is provided in Ramanathan et al.[31] Here, we summarize briefly how DTA works (see flowchart in Figure 1) and then describe how conformational substates are identified. Given a collection of tensors $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, ...$\boldsymbol{X}_T$, each of dimension $n_1 \times n_2 \times \ldots \times n_M$, DTA will find orthogonal matrices $\mathbf{U}_i$, one for each dimension, that minimizes the *dynamical deviation*, η, which is defined as follows:

$$\eta = \sum_{t=0}^{T} || \boldsymbol{X}_t - \boldsymbol{X}_t \prod_{i=1}^{M} \times_i (\mathbf{U}_i \mathbf{U}_i^T) ||_F^2 \quad (1)$$

Here, $||\mathbf{X}||_F^2$ is the square of the *Frobenius norm* of tensor $\boldsymbol{X}$, which is defined as

$$||\mathbf{X}||_F^2 = \sum_{i_1=1}^{n_1} \ldots \sum_{i_M=1}^{n_M} \boldsymbol{X}(i_1,\ldots i_M)^2 \quad (2)$$

and is equivalent to the sum of the inner product operation in matrices.

Informally, eq 1 is the difference between the actual data, $\mathbf{X}_t$, and the approximation of $\mathbf{X}_t$ in the space spanned by orthogonal matrices $\mathbf{U}_i$, denoted by $\mathbf{X}_t \prod_{i=1}^{M} \times_i (\mathbf{U}_i \mathbf{U}_i^T)$. In computer science and machine learning literature, η is often referred to as the error of reconstruction.[40] Here, $\mathbf{Y}_t = \mathbf{X}_t \prod_{i=1}^{M} \times_i \mathbf{U}_i$ is called the *core tensor*. The tensor−matrix multiplication operator, $\mathbf{X}_t \prod_{i=1}^{M} \times_i \mathbf{U}_i$, is defined as

$$\mathbf{X} \prod_{i=1}^{M} \times_i \mathbf{U}_i = \mathbf{X} \times_1 \mathbf{U}_1 \ldots \times_M \mathbf{U}_M \quad (3)$$

As illustrated in the flow-chart (Figure 1), DTA takes as input (i) the new incoming tensor $\mathbf{X}_t$ such that $1 \le t \le T$, (ii) the eigenvalues $\mathbf{S}_i^{(t-1)}|_{i=1}^{M}$, and (iii) the eigenvectors $\mathbf{U}_i^{(t-1)}|_{i=1}^{M}$ computed from the preceding call to DTA on tensor $\mathbf{X}_{t-1}$. If there are no previous eigenvalues/eigenvectors (i.e., at $t = 0$), then the only input is the first tensor, and the eigenvalues/eigenvectors will be computed for use in subsequent calls to DTA.

The algorithm proceeds by minimizing the variance in every dimension i, for $1 \le i \le M$. First, the input tensor is *unfolded* (or matricized) along the selected dimension d. Given $\mathbf{X}_t \in \mathbf{R}^{n_1 \times \ldots \times n_M}$, unfolding in dimension d involves constructing the $(\prod_{i \neq d} n_i) \times n_d$ matrix $\mathbf{X}_{(d)}$ such that each row is a vector in $\boldsymbol{R}^{n_d}$ obtained by holding d fixed and varying the remaining indices. Next, the variance matrix associated with dimension d from the previous call to DTA, $\mathbf{C}_d^{(t-1)}$, is reconstructed using the eigenvectors and eigenvalues from the preceding tensor, $\mathbf{X}_{t-1}$. The variance of the unfolded incoming tensor is, by definition, $\mathbf{X}_d^T \mathbf{X}_d$. At every time step, the variance estimates are updated according to the rule

$$\mathbf{C}_d^{(t)} \leftarrow \lambda \mathbf{C}_d^{(t-1)} + \mathbf{X}_d^T \mathbf{X}_{(i)} \quad (4)$$

where, λ is called the *memory parameter*. This parameter controls the degree to which previous tensors influence the current estimate of the variance. When $\lambda = 0$, only the present tensor at time t is considered to be relevant, and all of the past tensors are ignored for updating the variance matrix. By restricting our attention to $\lambda = 0$, we will be capturing an instantaneous description of the landscape.

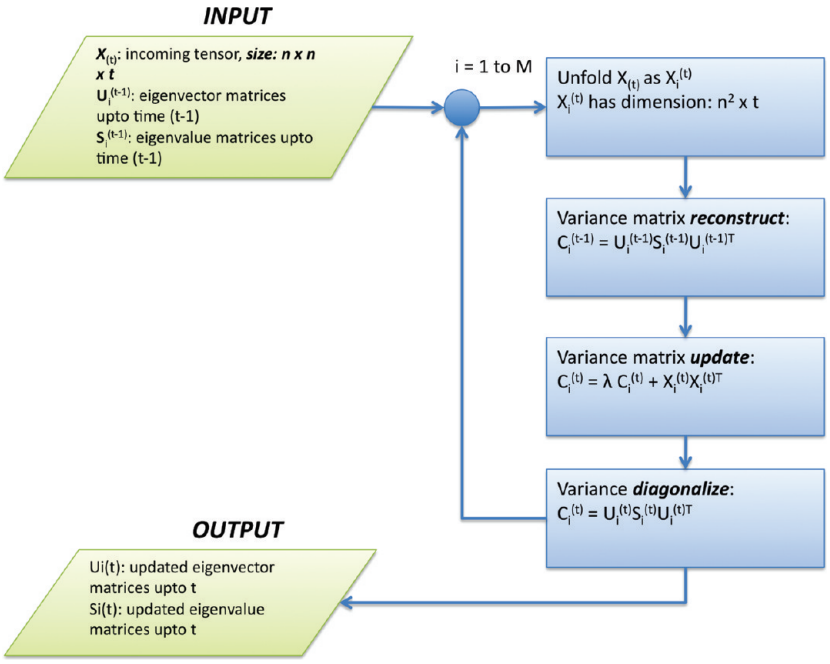

Figure 1. Schematic of dynamic tensor analysis used to capture spatiotemporal correlations from proteins simulations.

When $\lambda = 1$, all of the previous tensors are considered relevant and are used to estimate the variance matrix. Finally, DTA diagonalizes the variance matrix, resulting in an updated set of eigenvalues and eigenvectors that capture the dynamical behavior observed in the simulations observed thus far.

Identification of Conformational Substates Using Dynamical Deviations (η). Tensors provide a convenient means for encoding the collective dynamics observed in *windows* over the data. For example, in our experiments, we constructed tensors encoding the dynamics of the pairwise distances between the C^α atoms over windows of w sequential snapshots. Here, w is a parameter set by the user which can be adjusted to analyze different timescales. The resulting tensors thus had dimensions $n \times n \times w$ where n is the number of residues.

DTA tracks the evolution of the collective dynamics by updating the various covariance matrices $\mathbf{C}_d$ according to eq 4. These covariance matrices are then used to update the estimates of the orthogonal matrices $\mathbf{U}_i$, which reveal underlying patterns within the data. We have demonstrated previously[30,31] that it is possible to gain insights into dynamically coupled regions by clustering the $\mathbf{U}_i$ matrices.

The magnitude of any changes in the collective dynamics can be quantified by calculating the dynamical deviations η according to eq 1. Intuitively, any significant increase in η indicates that the collective motions have changed substantially. Such changes may be due to a transition between two different conformational substates. To detect such transitions, we monitor the empirical mean and standard deviation of η as the simulation is running. The instantaneous dynamical deviation threshold, η_t, is defined as follows:

$$\eta_t \geq \text{mean}(e_i|_{i=1}^t) + \alpha \times \text{std}(e_i|_{i=1}^t) \quad (5)$$

where $\eta_i|_{i=1}^t$ refers to the dynamical deviation up to time t, and α is an arbitrary positive constant. In our experiments, we set α to 2 (i.e., the second standard deviation). Thus, eq 5 can be used to automatically segment the MD trajectory into *dynamical segments*. As will be shown in the Results section, these segments correspond to different conformational substates characterized by different collective fluctuations. That is, spikes in the dynamical deviation are associated with the transition between conformational substates.

Principal Component Analysis (PCA). *Comparing Offline PCA with Online DTA.* We performed PCA on the C^α distance covariance matrix,[41] $\mathbf{D}$, defined as

$$D_{ij} = \langle (d_i - \langle d_i \rangle)(d_j - \langle d_j \rangle) \rangle = \mathbf{V}\Lambda_D\mathbf{V}^T \quad (6)$$

where d_i and d_j represent the pairwise distances of C^α atoms. The quantities within $\langle ... \rangle$ are average distances. The distance covariance matrix $\mathbf{D}$ was then diagonalized to obtain a set of eigenvectors $\mathbf{V}$ and eigenvalues λ_D. Modes were sorted according to their amplitudes in λ_D.

Note that $\mathbf{D}$ is a $m \times m$ matrix, where $m = n(n-1)/2$ and n is the number of C^α atoms in the protein. DTA, in contrast, models distance fluctuations using $n \times n$ matrices. In order to compare the results between PCA and DTA, therefore, it is necessary to construct a reduced representation of each eigenvector $\mathbf{v}_i$. To do this, we used the procedure described in Abseher and Nilges.[41] Here, the eigenvectors $\mathbf{v}_i$ are first mapped to a symmetric rank 2 matrix and then reduced to an n dimensional vector by summing the squares of the entries along a row:

$$v_l^{\text{red}} = \sum_{k=1}^{N} (v_{kl})^2 \quad (7)$$

This procedure allows one to accumulate the eigenvector components corresponding to distances from a common C^α atom. The resulting vectors are normalized and then used to compare the distance fluctuations measured by DTA and PCA.

Comparing Collective Fluctuations between Substates. PCA was also used to compare the dynamic segments obtained via DTA and RMSD. To do this, we performed PCA on the C^α atoms of the ensemble of structures in each segment. After removing the translational and rotational motions, the covariance matrix

$\mathbf{C}^{\mathrm{PCA}}$ was built using

$$C_{ij}^{\mathrm{PCA}} = (\mathbf{x}_i - \langle \mathbf{x}_i \rangle)(\mathbf{x}_j - \langle \mathbf{x}_j \rangle) = \mathbf{U}_{\mathrm{PCA}} \Lambda \mathbf{U}_{\mathrm{PCA}}^{T} \quad (8)$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ represent the positions of C^{α} atoms of residues i and j, respectively, and the $\langle ... \rangle$ is the average positional devaition. $\mathbf{U}_{\mathrm{PCA}}$ represents the eigenvectors, and Λ represents the eigenvalues (amplitudes of fluctuations) determined via PCA.

We measured the difference in the subspaces spanned by the segments by computing the normalized overlap[42] between the top 10 eigenvectors. The normalized overlap s between two substates A and B is defined as

$$s(A, B) = \frac{\sqrt{tr[(A^{1/2} - B^{1/2})^2]}}{\sqrt{tr(A) + tr(B)}} \quad (9)$$

The value of s can vary from a minimum of zero (no overlap) to a maximum of 1 (identical subspace). We note that the overlap was only used to compare subspaces, and not to test for convergence because short timescale windows are not expected to have converged.

We also compared the substates by computing the inner products of the respective eigenvectors. Here, we examined the top 10 eigenvectors (as in previous work[43]), which account for more than 70% of the overall variance. The inner product between eigenvectors measures the similarity between the direction of the large-scale fluctuations.

RESULTS

In previous work, we demonstrated that MD simulations of ubiquitin can accurately capture its behavior on microsecond timescales and that the motions revealed using quasi-harmonic analysis are functionally relevant.[43] In this section, we will summarize how DTA can be applied to characterize conformational substates in a protein simulation as it is progressing. All experiments were performed on tensors tracking the pairwise distances between C^{α} atoms over varying window sizes, as described below.

Comparison of Ubiquitin Dynamics. We first demonstrate that the Desmond simulation used in this paper sufficiently captures the inherent dynamics of ubiquitin by comparing it to previous experimental and computational results. We note that we carried out the simulation on the entire protein (residues 1−76). However, for the purposes of analysis, we have used residues 2−70. It is known that residues 1 and 71−76 are quite flexible, undergoing large scale fluctuations, and this may affect the interpretation of our results.

Figure 2 shows the root-mean squared fluctuations (RMSF) determined for C^{α} atoms of ubiquitin from residues 2 to 70. For comparison, these RMSFs are compared to the same quantities obtained from (a) a previously reported 0.5 μs simulation of ubiquitin obtained from multiple initial structures using AMBER as described in ref 43, (b) an NMR ensemble determined on the microsecond timescale (PDB code: 2K39),[44] and (c) 44 crystal structures obtained from the PDB. Casual inspection reveals that the four curves share significant overlap in terms of the flexible/constrained regions in ubiquitin. The overall correlation between the Desmond simulation and NMR ensemble, for example, is 0.8, whereas the correlation between the AMBER simulation and the NMR ensemble is 0.7. This observation leads us to believe that the Desmond simulations sample the overall conformation space of ubiquitin quite well and can be used for analyzing possible conformational substates.

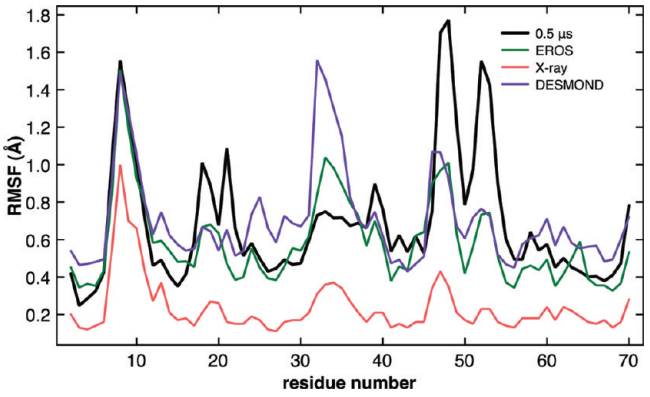

Figure 2. Comparison of Desmond to other ensembles, indicating largely similar fluctuation profiles across different forcefields and experimental ensembles. Our Desmond simulations of 1UBQ (150 ns) are most similar to the EROS (NMR) ensemble. Notice that the peaks in the RMSF curve coincide in Desmond, EROS, and a 0.5 μs ensemble determined by molecular dynamics simulations using the AMBER suite of tools.[43] The X-ray ensemble, as determined from 44 crystal structures, also indicates a similar fluctuation profile, albeit with a smaller magnitude of fluctuations. See correlation plots determined from AMBER, EROS, and Desmond in the Supporting Information.

Using RMSD to Identify Conformational Substates. A simple and straightforward approach to monitoring MD simulations is to compute root-mean squared deviations (RMSD) from an initial structure. RMSD measures the average distance between the backbones of two superimposed structures. We will show that DTA reveals different conformational substates than RMSD. The time-evolution of RMSD over the entire simulation is illustrated in Figure 3A. The average RMSD was computed with a window size of 10 snapshots ($w = 10$), spanning an interval of 0.1 ns. The average RMSD over the course of the 150 ns simulation was 2.2 Å, with a standard deviation of 0.56 Å. A visual inspection of the plot reveals that there are two points along the trajectory where significant structural changes occur. The first change occurs at approximately 42.0 ns and the second at 87.5 ns.

As shown in Table 1, the segments identified using RMSD are diverse in terms of their geometric properties. Closer inspection reveals that ubquitin remains fairly stable during the first 42 ns of the simulation. However, from 42 ns to about 87.5 ns, the simulation exhibits some large-scale fluctuations involving the α_1 helix and $\beta_1-\beta_2$ loops (Figure 3B). These motions are important in the context of ubiquitin binding.[43,44] In the last segment (87.5−150 ns), ubiquitin returns to a more native-like conformation with conformational changes in the $\beta_3-\beta_4$ loop and α_2 helix (Figure 3B).

RMSD is a measure of average structural deviations and does not necessarily provide insights into collective motions (i.e., whether the motions of different regions are coupled or independent). To compare the collective fluctuations across the three different substates identified using RMSD, we used PCA (described in the Methods section) to compare the subspaces spanned by each of these segments. As shown in Table 2, the pairwise overlap between segments is quite high. Moreover, the average overlap between each segment and the entire trajectory is 0.78, indicating that there is little difference in the collective fluctuations within each segment. This is further illustrated by the high overlap in the inner products of the top 10 eigenvectors (Supporting Information Figure S1).

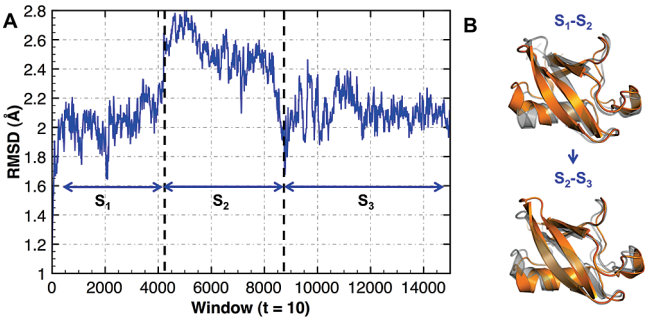

Figure 3. Tracking root-mean squared deviations (RMSD) indicates the presence of (at least) three segments. The RMSD profile from the MD simulation can be segmented into three parts: S_1 (0−42.0 ns), S_2 (42.0 ns +45.5 ns), and S_3 (87.5 ns +62.5 ns). Representative structures from each of the segments are compared on the right. In S_1 and S_2, we observe large-scale changes involving $\beta_1-\beta_2$ and $\beta_3-\beta_4$ loops as well as the C-terminal end of the α_1 helix. Minor conformational changes are observed in $\beta_2-\alpha_1$ and $\beta_4-\alpha_2$ loops. In S_2 and S_3, large-scale conformational changes are again observed in $\beta_1-\beta_2$ and $\beta_3-\beta_4$ loops as well as the C-terminal end of α_1. Note that with this transition, the protein comes back to a conformation that is more or less similar (in the RMSD sense) to the conformations in S_1.

Table 1. Summary of Segments Determined by Tracking RMSD[a]

no.	RMSD (Å)	scaled energy	time duration (ns)
S_1	2.0 ± 0.2	-0.128 ± 0.95	42.0
S_2	2.6 ± 0.4	0.123 ± 1.14	45.5
S_3	2.3 ± 0.3	-0.015 ± 0.91	62.5
total			150.0

[a] Each column shows the macroscopic geometric and energetic properties of segments. Scaled energy is defined as the sum of pairwise electrostatic and van der Waals interactions that have been normalized to have unit variance.[45,46] The time durations represent the length of the respective segments.

Table 2. Summary of Overlaps in Subspaces Spanned by RMSD[a]

no.	S_2	S_3	all
S_1	0.79	0.80	0.75
S_2		0.63	0.78
S_3			0.91

[a] Each column compares the subspace overlap between the the segments identified (S_1, S_2, and S_3). Normalized overlaps (eq 9) are computed as outlined in the Methods section. The final column represents the entire 150 ns trajectory. A summary of the inner products determined from the RMSD-based segmentation is provided in Figure S1 in the Supporting Information.

Next, we examined the scaled internal energy of the protein[45,46] in each segment. Internal energy, in this context, is defined as the sum of all nonbonded electrostatic and van der Waals interactions.[45] The internal energy values are normalized to have unit variance for ease of interpretation. The correlation coefficient between the RMSD profile (Figure 3) and the total energy is low (R = 0.26), as are the correlations between the RMSD profile and electrostatics (R = 0.24) and van der Waals (R = 0.06). Additionally, as illustrated in Table 1, the average scaled energies between the segments are fairly similar and have high standard deviations. We conclude that using RMSD as a metric for segmenting the trajectory into conformational substates results in a suboptimal energetic and dynamical description of the conformational landscape. As we will show in the subsequent subsections, DTA allows one to effectively overcome these limitations and identify conformational substates that also correspond to jumps in internal energy of the protein as simulations are progressing.

Comparing DTA with PCA. Figure 4 and Table 3 present the results of comparing normalized fluctuation encoded in the top five eigenvectors identified by DTA and PCA. The colored boxes in Figure 4 enclose regions with large distance fluctuations, including flexible loop regions $\beta_1-\beta_2$ (orange), $\beta_3-\beta_4$ (green), α_1 (magenta), and $\beta_2-\alpha_1$ (light blue). Notice that PCA and DTA show similar fluctuations. A quantitative comparison of the Spearman's rank correlation coefficients (Table 3) indicates that the similarity between some modes is statistically significant ($p < 0.05$). However, it also reveals some subtle differences between DTA and PCA. For example, a comparison of the top two modes from DTA indicates the greatest variance is associated with the entirety of α_1, but PCA detects motions only along the C-terminal end of α_1 (see modes A and B in Figure 4). The windowing aspect of DTA allows one to detect such hidden correlations which are not directly evident from PCA techniques. Thus, while PCA can pursue only the extent of spatial fluctuations (because the temporal dimension is averaged out), DTA can reveal correlations that include the temporal dimension. As we will show in the next section, the inclusion of the temporal dimension can also affect how the simulation can be segmented into different conformational substates.

The distance space PCA eigenvectors can be used to partition the landscape into conformational substates.[21−23,41] For this purpose, we use the projections from the distance space PCA to identify regions of the trajectory that show large deviations from the average behavior observed. The projections from PCA are computed *offline* using

$$q_i(t) = (\mathbf{d}(t) - \langle \mathbf{d} \rangle)\mathbf{v}_i \qquad (10)$$

where $q_i(t)$ represents the projection of conformation at time t, the first quantity on the right-hand side of the equation relates to the deviations in the pairwise distances, and the second term represents the eigenvector determined from eq 6. As shown in Figure 5, the top two eigenvectors partition the ubiquitin simulation into three substates (identified by the ellipses shown in the figure). The top two eigenvectors contribute to over 55% of the overall variance in the simulation, and hence the large-scale distance fluctuations observed can be considered a consequence of these two eigenvectors. The top two eigenvectors describe the fluctuations of the α_1 helix and the flexible loops $\beta_1-\beta_2$ and $\beta_3-\beta_4$.

The temporal evolution of the projections from the top two eigenvectors (see Supporting Information Figure S2) shows a large change around 42.0 ns of the simulation, followed by a gradual relaxation that begins at around 87.5 ns. The temporal evolution of the projections therefore closely follows the overall conformational changes depicted by the RMSD plots shown in Figure 3A. Further, as shown in Figure 5, the segments identified via PCA largely follow the RMSD partitioning of the conformational landscape. This is to be expected, since PCA pursues the variance (or extent of fluctuations) and therefore blindly chases the largest deviations observed in the simulation. However, as we have

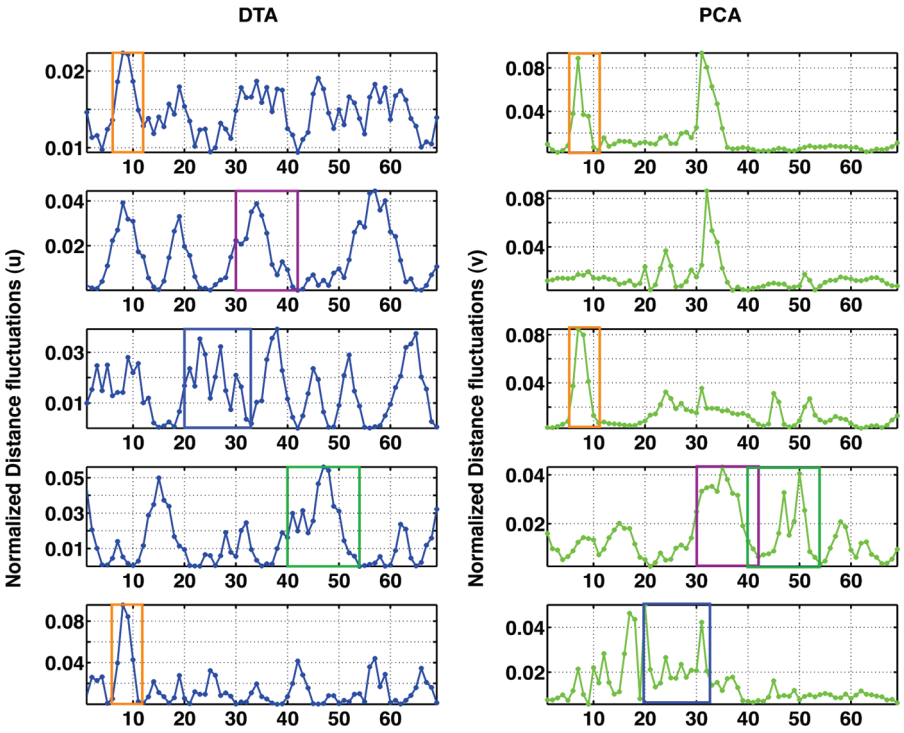

Figure 4. Comparing the online DTA with offline distance space PCA indicates similar fluctuations. Shown here are the normalized distance fluctuations from the top five eigenvectors from the online DTA performed over the entire trajectory and the offline distance space PCA. Note that while individual amplitudes across the top five modes might differ, the same regions in ubiquitin (highlighted in similar colored boxes across the plots) undergo similar fluctuations in both methods. A quantitative comparison between the modes also indicates the same (see Table 3).

Table 3. Similarity in Normalized Distance Fluctuations Determined from DTA (rows) and Distance Space PCA (columns) for the Top Five Eigenvectors from Each Method[a]

DTA/PCA	A	B	C	D	E
A	0.247 (4.06e-02)		0.303 (1.17e-02)	0.525 (5.03e-06)	
B	0.420 (3.65e-04)	0.242 (4.49e-02)	0.343 (4.04e-03)	0.255 (3.45e-02)	
C			0.249 (3.96e-02)		
D				0.323 (6.95e-03)	
E				0.247 (4.09e-02)	0.214 (7.71e-02)

[a] Only correlation values with p values (within brackets) that are significant (<0.05) are shown here.

recently pointed out,[31] the changes in these conformations need not necessarily correlate with energetic changes. We conclude that, for our simulations, PCA- and RMSD-based segmentations result in similar descriptions of the landscape.

DTA Segments the Conformational Landscape into Conformational Substates. In this section, we describe how dynamical deviations (η; described in the Methods section and eq 1) can be used to identify conformational substates as the simulations are running. We will first demonstrate how dynamical deviations η can be used to segment the trajectory by applying a threshold. We will then examine how our interpretation of the landscape changes as we examine different timescales (by increasing w).

The dynamical deviation η quantifies how much the previous window differs from the current window, in terms of its dynamical behavior. Spikes in η can therefore be used to segment the trajectory. For example, there are four obvious segments in Figure 6A, which uses a window size of 0.1 ns (w = 10). Each segment can be further partitioned according to η, as illustrated in Figure 6B. This is consistent with the view of a hierarchical conformational landscape.[2]

The process of segmenting the trajectory can be automated by applying a threshold η_t, as defined in eq 5. The time-evolution of η_t is shown as a red continuous line in Figure 6A. Note that η_t tends to rise for short periods of time and then stabilize as the simulation progresses. The rise and stabilization in η_t can be attributed to two aspects in the collective dynamics of the protein: (a) a period of fast changes in the fluctuations, indicated by the gray shaded regions in Figure 6A, followed by (b) a stable dynamical regime in which fluctuations are much less pronounced.

To compare the collective dynamics in the identified segments, we computed both the normalized overlaps (eq 9) and inner products between the top 10 eigenvectors determined from each of these segments. As shown in Figure 6C, the eigenvalue spectrum shows considerable difference between the segments identified. Table 4, which shows the overlaps between the subspace spanned by the top 10 eigenvectors in each segment, also illustrates that the maximal agreement between the subspaces is only about 0.64, which confirms that the motions between these dynamical segments are different. A second and more direct line

of evidence comes by examining the inner product matrices of each segment (see Supporting Information Figure S3), which further confirms that the collective motions are quite different in each of these segments.

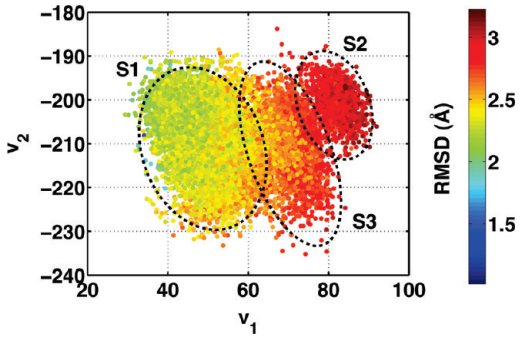

Figure 5. Projections from the top two eigenvectors determined from distance space PCA reveal three conformational substates closely following the RMSD based segmentation. Shown here are the projections of the top two eigenvectors determined from distance space PCA. Each conformation from the simulation (a total of 15 000 conformations) is colored with the individual RMSD values determined with respect to a single reference structure. Note that the segments marked (S_1, S_2, and S_3) correspond to the segments identified from Figure 3.

Now that we have quantified the extent of the dissimilarity in the collective motions between each dynamical segment, we examine whether these segments share conformations that show similarity in their internal energy distributions. Within each dynamical segment identified by DTA (Figure 6), we measured the mean internal energy and standard deviations and summarize the same in Table 5. DTA-defined dynamical segments are better separated in terms of average energy with relatively smaller standard deviations in overall energy. To better illustrate how DTA performs with respect to identifying isoenergetic substates, we plot the internal energy as shown in Figure 7. Note that the transitions (i.e., the peaks in Figure 6) correspond well with changes in the internal energy. Further, the transition between CS_3 and CS_4 shows a significant change in the internal energy values, which is not true of the transitions between the segments identified using RMSD or PCA (the segments identified by RMSD and PCA are labeled as S_1-S_3 in Figure 7). Thus, in our experiments, DTA segmented the trajectory into conformational substates that exhibit more energetic homogeneity than those identified via RMSD or PCA. Moreover, the DTA substates exhibit significant differences in terms of their collective motions.

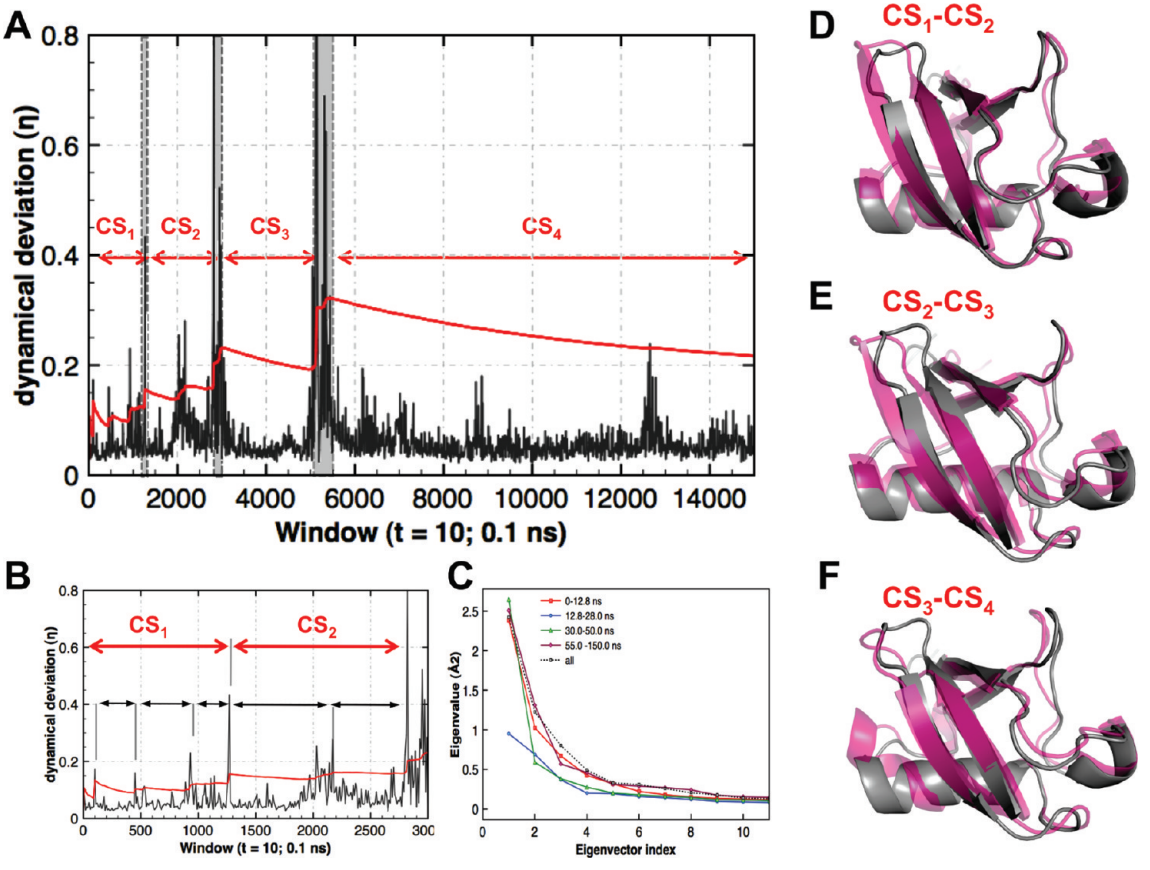

Figure 6. Tracking dynamical deviations (η) indicates the presence of four conformational substates. The η profile indicates the presence of sharp peaks along the simulation. The red line shows the second standard deviation intervals for η. (A) Peaks with significant changes in η indicate substantial changes in collective conformational dynamics, indicating that the protein has jumped into a new substate. In between transitions, there are phases where the collective behavior shows significant changes. This may be indicative of transition states involved in altering the collective behavior of the protein. (B) Each substate is formed by additional substates leading to the hierarchical organization of the landscape. (C) Comparison of the eigenspectrum from each of the dynamical segments. (D−F) Structural changes along the trajectory are summarized, highlighting the significant structural changes involved in the protein.

dx.doi.org/10.1021/ct100531j |J. Chem. Theory Comput. 2011, 7, 778–789

Table 4. Summary of Overlaps in Subspaces Spanned by DTA at $w = 10$ ($t = 0.1$ ns)[a]

no.	$CS_2^{0.1}$	$CS_3^{0.1}$	$CS_4^{0.1}$	all
$CS_1^{0.1}$	0.64	0.50	0.65	0.63
$CS_2^{0.1}$		0.54	0.58	0.60
$CS_3^{0.1}$			0.55	0.63
$CS_4^{0.1}$				0.87

[a] Each column compares the subspace overlap between the the segments identified (CS_1, CS_2, CS_3 ,and CS_4). Normalized overlaps (eq 9) are computed as outlined in the Methods section. The final column represents the entire 150 ns trajectory. A comparison of the inner products determined via PCA for each of the segments is shown in Figure S2 in the Supporting Information.

Table 5. Summary of CSs Determined by DTA[a]

no.	scaled energy	time duration (ns)
$CS_1^{0.1}$	-0.082 ± 1.06	$12.8 + 0.1$
$CS_2^{0.1}$	-0.254 ± 0.98	$15.1 + 0.9$
$CS_3^{0.1}$	-0.436 ± 0.99	$22.0 + 4.5$
$CS_4^{0.1}$	0.072 ± 0.94	94.6
total		150.0

[a] Each column shows the macroscopic geometric and energetic properties of segments. Scaled energy is computed as outlined in the Methods section.[45,46] The time durations represent the length of the respective segments, followed by the interval of time indicated by gray shaded regions highlighted in Figure 6.

The distinct substates identified by DTA are related to ubiquitin's function. We considered the structural changes along each of the segments (CS_1-CS_4). Between CS_1 and CS_2, the dominant conformational change involved is the rearrangement of the binding loop $\beta_1-\beta_2$ and a slight conformational change involving both α_1 and α_2 regions of the protein. The rest of the protein in this substate does not show any significant conformational change. In the transition between CS_2 and CS_3, one can observe the significant structural changes involved in $\beta_1-\beta_2$ as well as changes in the orientation of the β_5 strand and $\beta_2-\alpha_1$ loop regions of ubiquitin. In the transition between CS_3 and CS_4, the largest structural change involves the bending of the C-terminal end of α_1 as well as significant rearrangements in $\beta_1-\beta_2$ and $\beta_3-\beta_4$. These conformational changes have a direct implication in binding. As described in previous experimental[44] and computational studies,[43] the conformational changes in each of the conformational substates is unique and related to the movements of $\beta_1-\beta_2$ and $\beta_3-\beta_4$ loops, both of which form important interactions with ubiquitin's natural substrates. These changes occur throughout the simulation. However, there is only one segment where significant changes occur in α_1, implying that this motion may be much slower than the fluctuations associated with $\beta_1-\beta_2$ and $\beta_3-\beta_4$.

The fact that DTA tracks the temporal evolution of the covariance matrices (eq 4) produces qualitatively different segments than those identified via RMSD and PCA. In particular, DTA segments the trajectory based on changes in collective fluctuations, whereas PCA and RMSD segment according to large structural changes. Tables 2 and 4 support this distinction by showing that the average overlap between the various segments is low. We conclude that the inclusion of the temporal dimension provides additional information that is not accessible to PCA- and RMSD based methods. Further, as we will show in the next subsection, DTA can be used to examine the landscape on multiple timescales by varying the timescale parameter (w).

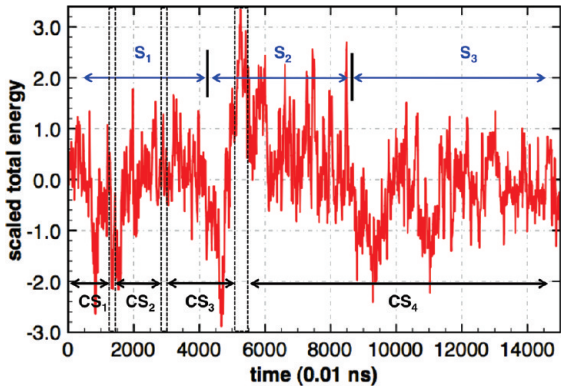

Figure 7. Tracking changes in total internal energy of the protein reveals conformational substates. The total internal energy of the protein computed from Desmond is shown by tracking windows of size $t = 10$. The segments (S_1-S_3) identified from Figure 3 are shown in blue. Observe that although there is some correspondence with the overall trends in RMSD values (Figure 3), the overall correlation between RMSD and energy values is quite low (0.26). The substates partitioned via DTA (Figure 6) are shown in black below. Note that the DTA-based segmentation captures the transition between CS_3 and CS_4, whereas the RMSD-based segmentation does not.

Effect of Increasing Time Scales on Identifying Conformational Substates. Note that in our analysis so far, we set the timescale parameter to $w = 10$ snapshots (0.1 ns). In this section, we demonstrate how DTA can be used to detect conformational transitions on longer timescales. We note that the timescale parameter is set by the user and that it is possible, in principle, to perform DTA on multiple timescales simultaneously (e.g., using multiple processors). For the remainder of this section, superscripts will be used to identify the timescale associated with each substate according to the timescale. For example, CS_2^1 denotes the second conformational substate on a 1.0 ns timescale. This will facilitate the comparison between substates on different timescales.

We also considered time windows of $w = 100$ (1.0 ns) and $w = 500$ (5.0 ns). The use of three different time windows ($w = 10$, $w = 100$, and $w = 500$) allows us to resolve the landscape on different timescales. The analysis on the 1 ns timescale has three dynamical segments ($CS_1^1-CS_3^1$). These segments are diverse, as evidenced by the eigenspectrum (Figure 8) and the low overlaps between the subspaces (Table 6). The analysis on the 5 ns timescale has two major dynamical segments ($CS_1^5-CS_2^5$; see Figure 9B). Pairwise comparisons of the η values across timescales are shown in Figure 9. Notice that the locations of peaks in the η values occasionally coincide (see shaded rectangles), but there are peaks that occur on only one timescale (see black arrows).

Spatially, the changes in the collective fluctuations in $CS_1^1-CS_3^1$ tend to be localized near $\beta_1-\beta_2$. On the 1 ns timescale, however, the collective fluctuations in $CS_1^1-CS_3^1$ are concentrated in the $\beta_1-\beta_2$ turn, α_1, and the $\beta_2-\alpha_1$ loop. On the 5 ns timescale, the collective fluctuations in $CS_1^5-CS_2^5$ involve α_1

We also examined the changes in collective motions by comparing the covariance matrices (see Figure 10). Here, the covariances on the three timescales were normalized to have unit variance. Note that between 0.1 and 1.0 ns timescales, the differences in the covariance matrices are more global (left panel

dx.doi.org/10.1021/ct100531j |J. Chem. Theory Comput. 2011, 7, 778–789

in Figure 10A), spanning multiple regions of the protein. A similar observation can also be made for the comparison between 0.1 and 5.0 ns timescales (middle panel of Figure 10A). These changes are concentrated along the functionally relevant regions of the protein. In particular, changes in collective fluctuations on 1.0 ns timescales are reduced along $\beta_1-\beta_2$ with respect to α_1 and $\beta_4-\alpha_2$ regions (Figure 10B). However, collective changes are enhanced across several regions in the protein including $\beta_2-\alpha_1$, $\beta_3-\beta_4$, and $\alpha_2-\beta_5$, indicating that, on longer timescales, these correlations become more pronounced (Figure 10C). Note that between 0.1 and 5.0 ns timescales, only the correlations become more pronounced—the regions identified to be flexible within ubiquitin are still the same. This emphasizes the inherent flexibility in ubiquitin that is present even on smaller timescales, which is observed on longer timescales, albeit with higher amplitudes. It is also clear from the two plots that one can identify individual residues that undergo changes in their distance fluctuations with respect to the rest of the protein. In this case, we observe Gln40 and Asp22 to undergo changes in their motions.

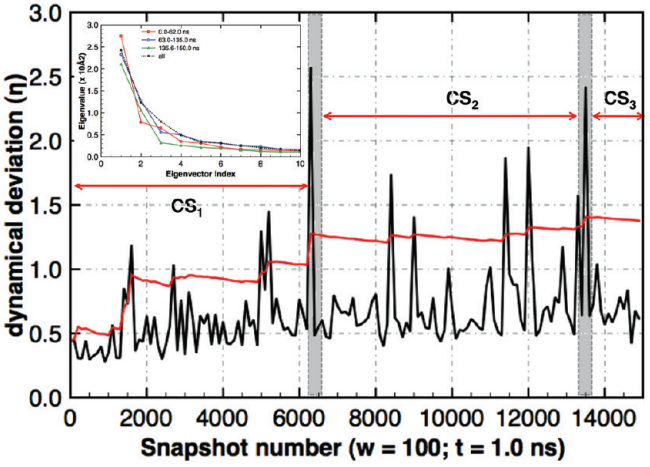

Figure 8. CSs on longer timescales reveal unique dynamical fluctuations in ubiquitin. The plot shows the η as a function of the time window (w = 100; 1.0 ns). On longer timescales, we observe larger η and only three segments ($CS_1^1-CS_3^1$). Within each segment, we observe smaller spikes, indicating the presence of smaller substates. Gray shaded regions indicate segments where large changes are observed in η, indicating dynamical transition points. The inset shows the comparison of the eigenspectrum determined from the segments here. A comparison of the subspaces is shown in Table 6.

Table 6. Summary of Overlaps in Subspaces Spanned by DTA at w = 100; (t = 1.0 ns)[a]

no.	$CS_2^{1.0}$	$CS_3^{1.0}$	all
$CS_1^{1.0}$	0.64	0.52	0.76
$CS_2^{1.0}$		0.65	0.85
$CS_3^{1.0}$			0.62

[a] Each column compares the subspace overlap between the the segments identified ($CS_1^{1.0ns}$, $CS_2^{1.0ns}$n and $CS_3^{1.0ns}$). Normalized overlaps (eq 9) are computed as outlined in the Methods section. The final column represents the entire 150 ns trajectory.

A comparison of the covariance matrices at 1.0 and 5.0 ns does not yield significant changes. As seen from the right-hand panel of Figure 10A, the color bar shows relatively smaller localized changes in the covariances. This is to be expected since the timescales are roughly on the same order. However, we do observe several localized changes in the protein's motions, notably along the flexible $\beta_1-\beta_2$ region of the protein. This region tends to undergo fast fluctuations (O(ps)) and, hence, is clearly visible in the difference plots. A second localized fluctuation occurs in the $\alpha_1-\beta_3$ region; however, it is of much smaller amplitude. Thus, depending on the timescale at which distance fluctuations are monitored, DTA can provide a succinct and unique resolution of the conformational landscape.

DISCUSSION

DTA Overcomes Limitations of Readily Available Observables from MD Simulations. Popular measures for monitoring MD simulations include RMSD, radius of gyration, kinetic/potential/total energy, velocity, pressure, and temperature. These observables have been traditionally used to monitor the "health" of the simulation and identify events that affect its quality. While these observables are certainly valuable, they do not (and cannot) track how concerted changes to a group of atoms or residues within a protein affect its dynamical behavior as the simulations are progressing. Tracking concerted, collective changes in the dynamical behavior of a protein needs a suitable measure that is sensitive enough not only to capture large

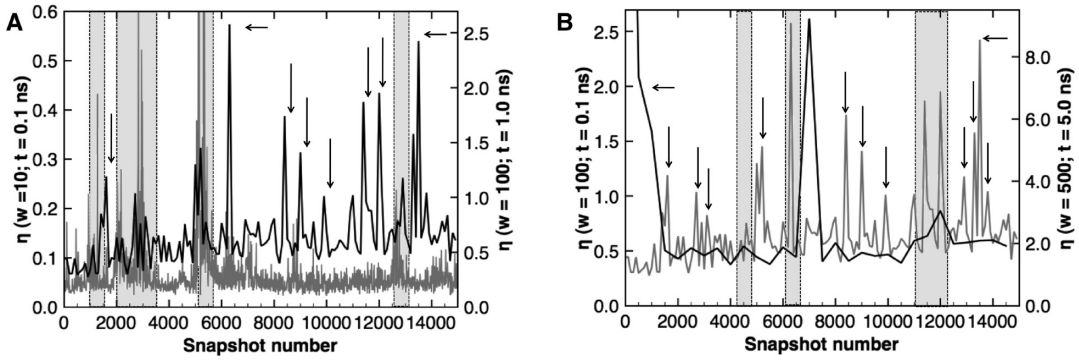

Figure 9. Comparing η on different timescales reveals differences in the resolution of the conformational landscape of ubiquitin. Panel A compares the landscape on timescales of 0.1 and 1.0 ns. Panel B compares the landscape at 1.0 and 5.0 ns resolutions. Note that there are two axes used here. The plots show whether there are overlaps in η on different timescales. Shaded rectangles are used to highlight regions that show close correspondence to changes in collective behavior on both timescales being compared. Black arrows show the time points that are present on the faster timescale but not on the longer timescale.

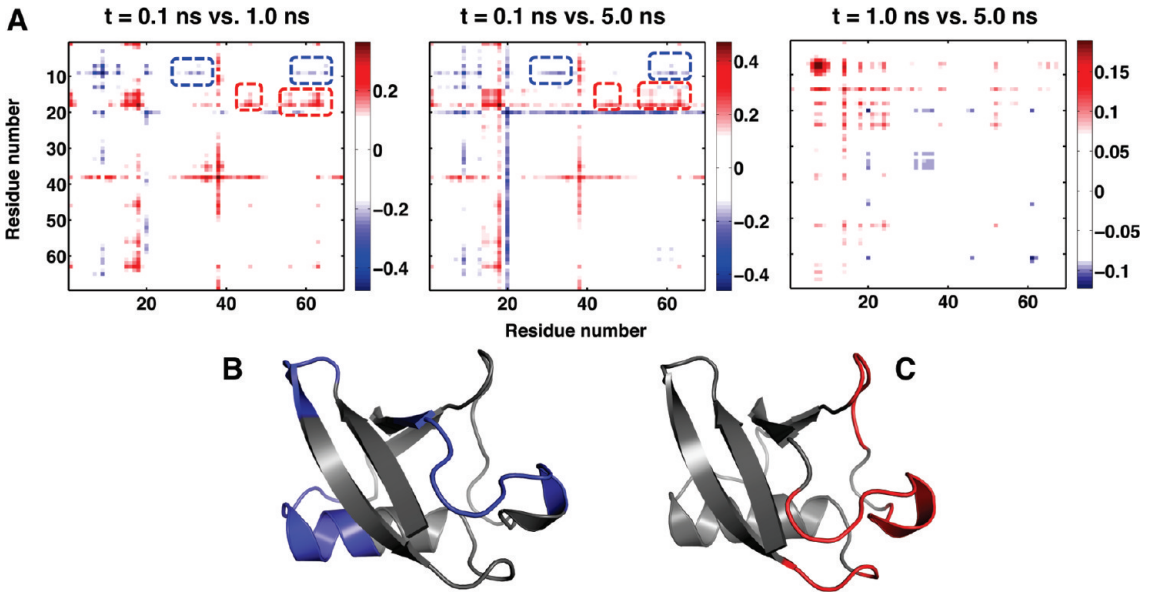

Figure 10. Comparison of covariance of distance fluctuations on longer timescales, indicating change along functionally relevant regions in ubiquitin. (A) The difference in the covariance determined at three different temporal resolutions, namely, $w = 10$ (0.1 ns), $w = 100$ (1.0 ns), and $w = 500$ (5.0 ns). Although the covariance on each timescale implicates similar flexible regions in ubiquitin (see the Supporting Information), the differences reveal subtle yet important dynamical changes. The regions showing significant differences are highlighted using a rounded rectangle in each plot. Notice that there are regions of both increased and decreased covariance. (B) Regions that show a significant decrease in covariances on longer timescales. (C) Regions that show a significant increase in covariances. The regions highlighted are important for ubiquitin's binding motions (see the Results section for more details). Also note that, between the 1.0 and 5.0 ns timescales, the differences in the covariance matrices are highly localized—indicating that the overall resolution of the landscape has not changed very much.

conformational changes but also to characterize subtle, yet functionally relevant changes. In this paper, we have used one such measure, namely, dynamical deviations, η (eq 1), to organize the conformational landscape in terms of substates that share similar collective behaviors and energies.

Recently, a technique called TimeScapes based on coarse-graining MD trajectories using side-chain contacts was implemented to track time-dependent conformational changes in protein simulations.[47] TimeScapes is able to model conformational shifts along the simulation representing secondary and tertiary structures corresponding to functionally important transitions. Furthermore, it was also able to identify basins and transitions based on an activity measure. The approach is similar in terms of analyzing simulations online. However, unlike DTA, TimeScapes does not allow for tracking collective behavioral changes over simulations. Our approach is complementary and allows one to track any feature (see below) from a MD simulation for changes in collective behavior. In previous work,[48] we have also demonstrated the ability of DTA to quantify changes in contact maps during MD simulations.

It must be noted that while DTA overcomes the limitations posed by simple observable measures from MD simulations, it is dependent on the representation used in describing collective behavior. For our representation, we have used C^{α}-distance matrices as a means to monitor changes in collective motions. It is entirely possible to use other distance/geometric measures such as hydrogen bond/hydrophobic interactions,[49] dihedral/torsion angles,[50] and energy measures such as electrostatics/van der Waals/potential/kinetic energy[45,46] to track collective changes in these measures. It is of interest to see how well changes in geometric measures are correlated with changes in energy measures and will be pursued in a future study using a fairly straightforward extension of DTA.[51]

The energetic description obtained via DTA, especially from the dynamical segments, is much better than the RMSD based segmentation of the landscape. Although one should not expect to find energetically homogeneous substates directly from DTA, it is at least encouraging to note that the there is some energetic similarities within a dynamical segment. Thus, DTA represents a step in the right direction toward understanding the complex spatiotemporal dependencies that might exist within the conformational and energetic landscape.

Spatiotemporal Insights into Conformational Landscape. A valuable utility of DTA in monitoring collective behavior is that it can capture both spatial and temporal changes in collective behavior over the course of a MD simulation. This has some bearing on our understanding of the complex conformational landscape. The conformational landscape that is sampled by the MD simulation can be divided into phases: a *stable* phase in which the η does not change very much and a *dynamic* phase in which the η shows significant changes in its behavior. The stable phases are indicated by relatively small changes in η, whereas the dynamic phases involve large changes in η. The dynamic phase where η shows an increase involves significant rearrangements in the overall conformation before stabilizing with minor conformational changes, dominated by localized motions in side chains and corresponding changes in C^{α} positions.

In our simulations of ubiquitin, we observed that at a temporal resolution of 0.1 ns, there are numerous changes that occur rather suddenly over the 150 ns. Though these changes involve the functionally relevant binding regions ($\beta_1-\beta_2$, $\beta_3-\beta_4$, and C-terminal tip of α_1), in these small time windows, we do not observe any significant correlations between the functionally important regions in ubiquitin. However, on longer timescales (1.0 and 5.0 ns respectively), we find there is a small correlation between $\beta_1-\beta_2$ and $\beta_3-\beta_4$ (Figure 10A; observe the correlation

between residues 7 and 9 with 46 and 48). The emergence of such correlations can have some consequence in interpreting functional relevance of these motions in binding. Both $\beta_1-\beta_2$ and $\beta_3-\beta_4$ may need to be precisely positioned in order to form the interactions needed to bind its substrate. While it remains to be seen if such coordinated motions are a prerequisite for interface formation,[52] we note that the correlations between the binding regions occur only along specific time points in our simulations. Such coordinated motions may also play a role in protein folding pathways where secondary structures might need to interact before sampling native state configurations.[53] The ability of DTA to pick up such correlated changes along folding pathways will be studied in the future.

Further, a comparison of DTA with offline PCA reveals the additional insights obtained by using online techniques. The fluctuations (large-scale motions) described by DTA and PCA are quite similar. However, they do differ in terms of the correlated motions that are depicted. On the basis of the time resolution used in DTA, these correlations between different parts of the protein can change. PCA-based techniques operate on time-averaged covariance matrices and, hence, cannot detect subtle changes that may occur over the course of a simulation. DTA, however, includes correlations that arise on different timescales. This provides a unique viewpoint for interpreting the conformational landscape: changes in the correlations between spatially separated parts of the protein on different timescales may provide insights into how these distal couplings arise.

CONCLUSIONS

In this paper, we have demonstrated the utility of dynamic tensor analysis (DTA) to identify and characterize dynamical segments along a MD trajectory that share similar geometric and energetic properties. That these dynamical segments may form a starting point to identify conformational substates has several interesting pointers for future studies. For one, the application of DTA to the study of protein folding pathways to distinguish successful (folded) versus unsuccessful (unfolded) states is already underway. It is also useful to investigate if one can correlate multiple tensor streams including geometric and energetic properties across simulations, which we plan to pursue using an existing approach.[51] We hope that the availability of such tools can be valuable in processing extant data sets, such as those in Dynameomics,[54] and those that will be produced by Anton.[13]

The ability to identify and relate protein motions at different temporal scales opens up opportunities for characterizing protein landscape in the spirit of previous work by Frauenfelder and coworkers.[55] Further, it will also provide an integrating platform for combining studies that simultaneously includes both spatial and temporal aspects of the complex conformational landscape, which is indeed thought to be a requirement to fully characterize a protein's conformational landscape. Furthermore, in the context of complex biological functions such as enzyme catalysis,[56] it is believed that such a holistic description of the landscape can be valuable.

Implementation and Availability. DTA is implemented in both Python[57] and Matlab.[58] The Matlab code is available on request from the authors. The DTA implementation in Python is part of a package called pyTensor[49] and is hosted at http://code.google.com/p/pytensor/. The source code and package are available for download for free. The package has been implemented such that it can be easily adapted to read a variety of inputs generated from MD simulations. Processing outputs from MD simulations using custom-written python scripts are also available upon request. Currently, our custom-written scripts can read distance and energy values from MD simulations (written for a variety of packages including AMBER and Desmond). The DTA package additionally implements a subset of the tensor toolbox,[59,60] which can be used for further development. The Python version of the code can be integrated into any number of available packages including HiMach[61] and Biskit.[62]

ASSOCIATED CONTENT

Supporting Information. Six additional figures and their descriptions are available. This material is available free of charge via the Internet at http://pubs.acs.org.

AUTHOR INFORMATION

Corresponding Author

*Phone: (412) 268 7571. Fax: (412) 268 5576. E-mail: cjl@cs.cmu.edu.

Present Addresses

[§]Computational Biology Institute, Computer Science Research, Computer Science and Mathematics

ACKNOWLEDGMENT

The authors thank Dr. Pratul K. Agarwal for providing access to the ubiquitin simulations and critically reading the manuscript and commenting on it.

REFERENCES

(1) Frauenfelder, H.; Petsko, G. A.; Tsernoglou, D. *Nature* **1979**, *280*, 558–563.

(2) Frauenfelder, H.; Parak, F.; Young, R. D. *Annu. Rev. Biophys. Biophys. Chem.* **1988**, *17*, 451–479.

(3) Henzler-Wildman, K.; Kern, D. *Nature* **2007**, *450*, 964–972.

(4) Boehr, D. D.; Nussinov, R.; Wright, P. E. *Nat. Chem. Biol.* **2009**, *5*, 789–796.

(5) Fraser, J.; Clarkson, M.; Degnan, S.; Erion, R.; Kern, D.; Alber, T. *Nature* **2009**, *462*, 669–673.

(6) Boehr, D. D.; Dyson, H. J.; Wright, P. E. *Science* **2006**, *313*, 1638–1642.

(7) Zaccai, G. *Science* **2000**, *288*, 1604–1607.

(8) Fitter, J. *Biophys. J.* **2003**, *84*, 3924–3930.

(9) Eisenmesser, E. Z.; Bosco, D. A.; Akke, M.; Kern, D. *Science* **2002**, *295*, 1520–1523.

(10) Eisenmesser, E. Z.; Millet, O.; Labeikovsky, W.; Korzhnev, D.; Bosco, D.; Skalicky, J.; Kay, L.; Kern, D. *Nature* **2005**, *438*, 117–121.

(11) Balbach, J.; Forge, V.; van Nuland, N. A. J.; Winder, S. L.; Hore, P. J.; Dobson, C. M. *Nat. Struct. Mol. Biol.* **1995**, *2*, 865–870.

(12) Karplus, M.; McCammon, J. A. *Nat. Struct. Biol.* **2002**, *9*, 646–652.

(13) Shaw, D. E.; et al. *SIGARCH Comput. Archit. News* **2007**, *35*, 1–12.

(14) Stone, J. E.; Phillips, J.; Freddolino, P. L.; Hardy, D. J.; Trabuco, L.; Schulten, K. *J. Comput. Chem.* **2007**, *28*, 2618–2640.

(15) Anderson, J. A.; Lorenz, C. D.; Travesset, A. *J. Comput. Phys.* **2008**, *227*, 5342–5359.

(16) Friedrichs, M. S.; Eastman, P.; Vaidyanathan, V.; Houston, M.; LeGrand, S.; Beberg, A. L.; Ensign, D. L.; Bruns, C. M.; Pande, V. S. *J. Comput. Chem.* **2009**, *30*, 864–872.

(17) Hampton, S.; Agarwal, P. K.; Alam, S. R.; Crozier, P. S. Towards In *Proceedings of the International Conference on High Performance Computing & Simulation*; Smari, W. A., McIntire, J. P., Eds.; HPCS' 10; IEEE: Piscataway, NJ, 2010; pp 98−107.

(18) Bowers, K. J.; Dror, R. O.; Shaw, D. E. *J. Comput. Phys.* **2007**, *221*, 303–329.
(19) Hess, B.; Kutzner, C.; van der Spoel, D.; Lindahl, E. *J. Chem. Theory Comput.* **2008**, *4*, 435–447.
(20) Jolliffe, I. T. *Principal Component Analysis*, 2nd ed.; Springer-Verlag New York, Inc.: New York, 2002; Springer Series in Statistics.
(21) Karplus, M.; Kushick, J. N. *Macromolecules* **1981**, *14*, 325–332.
(22) Amadei, A.; Lissen, A. B. M.; Berendsen, H. J. C. *Proteins* **1993**, *17*, 412–425.
(23) Materese, C. K.; Goldmon, C. C.; Papoian, G. A. *Proc. Natl. Acad. Sci. U. S. A.* **2008**, *105*, 10659–10664.
(24) Okazaki, K.; Takada, S. *Proc. Natl. Acad. Sci. U. S. A.* **2008**, *105*, 11182–11187.
(25) Lange, O.; Grubmuller, H. *Proteins* **2007**, *70*, 1294–1312.
(26) Bowman, G. R.; Beauchamp, K. A.; Boxer, G.; Pande, V. S. *J. Chem. Phys.* **2009**, *131*, 124101.
(27) Shao, J.; Tanner, S.; Thompson, N.; Cheatham, T. *J. Chem. Theory Comput.* **2007**, *3*, 2312–2334.
(28) Frickenhaus, S.; Kannan, S.; Zacharias, M. *J. Comput. Chem.* **2009**, *30*, 479–492.
(29) Daura, X.; van Gunsteren, W. F.; Mark, A. E. *Proteins* **1999**, *34*, 269–280.
(30) Ramanathan, A.; Agarwal, P.; Kurnikova, M.; Langmead, C. In *Research in Computational Molecular Biology*; Batzoglou, S., Ed.; Springer: Berlin, 2009; Vol. 5541; Lecture Notes in Computer Science, pp 138−154.
(31) Ramanathan, A.; Agarwal, P. K.; Kurnikova, M.; Langmead, C. J. *J. Comput. Biol.* **2010**, *17*, 309–324.
(32) Sun, J.; Tao, D.; Faloutsos, C. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*; Eliassi-Rad, T., Ungar, L., Craven, M., Gunopulos, D., Eds.; KDD '06; ACM: New York, 2006; pp 374−383.
(33) Jorgensen, W.; Tirado-Rives, J. *J. Am. Chem. Soc.* **1988**, *110*, 1657–1666.
(34) Jorgensen, W.; Maxwell, D.; Tirado-Rives, J. *J. Am. Chem. Soc.* **1996**, *118*, 11225–11236.
(35) Berweger, C. D.; van Gunsteren, W. F.; Müller-Plathe, F. *Chem. Phys. Lett.* **1995**, *232*, 429–436.
(36) Bowers, K. J.; Chow, E.; Xu, H.; Dror, R. O.; Eastwood, M. P.; Gregersen, B. A.; Klepeis, J. L.; Kolossvary, I.; Moraes, M. A.; Sacerdoti, F. D.; Salmon, J. K.; Shan, Y.; Shaw, D. E. *SC 2006 Conference, Proceedings of the ACM/IEEE*; IEEE Computer Society: Los Alamitos, CA, 2006; p 43.
(37) Krautler, V.; van Gunsteren, W. F.; Hünenberger, P. *J. Comput. Chem.* **2001**, *22*, 501–508.
(38) Shan, Y.; Klepeis, J. L.; Eastwood, M. P.; Dror, R. O.; Shaw, D. E. *J. Chem. Phys.* **2005**, *122*, 054101.
(39) Papadimitriou, S.; Sun, J.; Faloutsos, C. In *Proceedings of the 31st International Conference on Very Large Data Bases, Trondheim, Norway*; Böhm, K., Jensen, C. S., Haas, L. M., Kersten, M. L., Larson, P.-Å., Ooi, B. C., Eds.; ACM: New York, 2005; Vol. 31, pp 697−708.
(40) Smilde, A.; Bro, R.; Geladi, P. *Multi-way Analysis: Applications in the Chemical Sciences*; J. Wiley and Sons, Ltd.: West Sussex, England, 2004.
(41) Abseher, R.; Nilges, M. *J. Mol. Biol.* **1998**, *279*, 911–920.
(42) Hess, B. *Phys. Rev. E* **2000**, *62*, 8438–8448.
(43) Ramanathan, A.; Agarwal, P. K. *J. Phys. Chem. B* **2009**, *113*, 11169–11180.
(44) Lange, O. F.; Lakomek, N.-A.; Fares, C.; Schroder, G. F.; Walter, K. F. A.; Becker, S.; Meiler, J.; Grubmuller, H.; Griesinger, C.; de Groot, B. L. *Science* **2008**, *320*, 1471–1475.
(45) Kong, Y.; Karplus, M. *Structure* **2007**, *15*, 611–623.
(46) Kong, Y.; Karplus, M. *Proteins* **2009**, *74*, 145–154.
(47) Wriggers, W.; Stafford, K. A.; Shan, Y.; Piana, S.; Maragakis, P.; Lindorff-Larsen, K.; Miller, P. J.; Gullingsrud, J.; Rendleman, C. A.; Eastwood, M. P.; Dror, R. O.; Shaw, D. E. *J. Chem. Theory Comput.* **2009**, *5*, 2595–2605.
(48) Ramanathan, A.; Agarwal, P. K.; Langmead, C. J. *Using tensor analysis to characterize contact-map dynamics in proteins*; Technical Report CMU-CS-08-10, Carnegie Mellon University: Pittsburgh, PA, 2008.
(49) Yoo, J. O.; Ramanathan, A.; Langmead, C. J. *PyTensor: A Python based Tensor Library*; Technical Report CMU-CS-10-102; Carnegie Mellon University: Pittsburgh, PA, 2010.
(50) Maisuradze, G. G.; Leitner, D. *Proteins* **2007**, *67*, 569–578.
(51) Sun, J.; Papadimitriou, S.; Yu, P. S. In *Learning from Data Streams: Processing Techniques in Sensor Networks*; Gama, J., Gaber, M. M., Eds.; Springer: New York, 2007; Chapter 11, pp 165−184.
(52) Yogurtcu, O. N.; Erdemli, S. B.; Nussinov, R.; Turkay, M.; Keskin, O. *Biophys. J.* **2008**, *94*, 3475–3485.
(53) Narzi, D.; Daidone, I.; Amadei, A.; Di Nola, A. *J. Chem. Theory Comput.* **2008**, *4*, 1940–1948.
(54) van der Kamp, M. W.; Schaeffer, R. D.; Jonsson, A. L.; Scouras, A. D.; Simms, A. M.; Toofanny, R. D.; Benson, N. C.; Anderson, P. C.; Merkley, E. D.; Rysavy, S.; Bromley, D.; Beck, D. A.; Daggett, V. *Structure* **2010**, *18*, 423–435.
(55) Frauenfelder, H.; Sligar, S.; Wolynes, P. *Science* **1991**, *254*, 1598–1603.
(56) Agarwal, P. K. *Microb. Cell Fact.* **2006**, *5*, 2.
(57) van Rossum, G. *Python Reference Manual*; Technical Report CS-R9526; Centrum voor Wiskunde en Informatica (CWI): Amsterdam, 1995.
(58) *MATLAB*, R2009a; Mathworks: Natick, MA, 2009.
(59) Bader, B.; Kolda, T. *ACM T. Math. Software* **2006**, *32*, 635–653.
(60) Bader, B.; Kolda, T. *SIAM J. Sci. Comput.* **2007**, *30*, 205–231.
(61) Tu, T.; Rendleman, C. A.; Borhani, D. W.; Dror, R. O.; Gullingsrud, J.; Jensen, M. O.; Klepeis, J. L.; Maragakis, P.; Miller, P.; Stafford, K. A.; Shaw, D. E. A scalable parallel framework for analyzing terascale molecular dynamics simulation trajectories; In *Proceedings of ACM/IEEE Conference on Supercomputing*; SC' 08; IEEE: Piscataway, NJ, 2008; pp 56:1−12.
(62) Grünberg, R.; Nilges, M.; Leckner, J. *Bioinformatics* **2007**, *23*, 769–770.