THE JOURNAL OF PHYSICAL CHEMISTRY B

Article

pubs.acs.org/JPCB

WExplore: Hierarchical Exploration of High-Dimensional Spaces Using the Weighted Ensemble Algorithm

Alex Dickson[†] and Charles L. Brooks, III*[,†,‡]

[†]Department of Chemistry, University of Michigan, Ann Arbor, Michigan, United States

[‡]Department of Biophysics, University of Michigan, Ann Arbor, Michigan, United States

ABSTRACT: As most relevant motions in biomolecular systems are inaccessible to conventional molecular dynamics simulations, algorithms that enhance sampling of rare events are indispensable. Increasing interest in intrinsically disordered systems and the desire to target ensembles of protein conformations (rather than single structures) in drug development motivate the need for enhanced sampling algorithms that are not limited to "two-basin" problems, and can efficiently determine structural ensembles. For systems that are not well-studied, this must often be done with little or no information about the dynamics of interest. Here we present a novel strategy to determine structural ensembles that uses dynamically defined sampling regions that are organized in a hierarchical framework. It is based on the weighted ensemble algorithm, where an ensemble of copies of the system ("replicas") is directed to new regions of configuration space through merging and cloning operations. The sampling hierarchy allows for a large number of regions to be defined, while using only a small number of replicas that can be balanced over multiple length scales. We demonstrate this algorithm on two model systems that are analytically solvable and examine the 10-residue peptide chignolin in explicit solvent. The latter system is analyzed using a configuration space network, and novel hydrogen bonds are found that facilitate folding.

INTRODUCTION

Molecular motions are complex. They are the aggregate of an extremely large number of interactions between pairs of atoms in a molecule. For a protein that is 50 amino acids long, there are roughly 1 000 000 pairwise atomic interactions to consider, not taking into account interactions with solvent. In many cases, a simpler dynamics emerges from this complexity, which allows for a meaningful projection of the dynamics onto one or two degrees of freedom. This is essential, since it is from these projections that we derive an understanding of the molecule itself. Molecular dynamics can be a useful tool to derive this understanding, but computational limitations prohibit direct study of the time scales over which much of the interesting dynamics takes place (microseconds to milliseconds). This motivates the development of computational methods that enhance sampling in biomolecular systems.

A wide array of enhanced sampling methods have been developed over the past decades, each of which has its own advantages and disadvantages. One class of methods works by dividing a low-dimensional order parameter space into regions, and encouraging or enforcing even sampling in each region.[1−5] This enables thorough sampling of low probability regions, such as transition states, which allows for the elucidation of transition mechanisms, transition rates, and free energy barriers. Although these approaches are useful for studying transitions between known states, their application requires much of the characterization of the dynamics to be done beforehand, as one needs to define the order parameters that describe the transition before sampling begins. String-based algorithms circumvent this problem by defining a string that connects initial and final states that winds its way through a (possibly) high-dimensional order parameter space;[6,7] however, they are restricted to a specific type of problem, in which two deep minima are separated by a low-probability transition state. Other methods are needed to characterize more complicated dynamics, such as the dynamics of an unfolded protein, or intrinsically disordered proteins.

Another class of methods enhance sampling in an undirected fashion and have the ability to describe transitions between multiple basins along a myriad of transition pathways. Replica exchange[8,9] allows for the crossing of free energy barriers by coupling the system to simulations that are run at higher temperatures. This is a powerful technique, but it scales poorly with increasing system size. Other approaches, including metadynamics,[10] temperature-accelerated molecular dynamics,[11] and adiabatic molecular dynamics,[12] use biasing forces to accelerate sampling in a limited sampling space of collective variables. This approach has proven useful but is limited to cases where the collective variables are differentiable; more complicated order parameters such as TM scores,[13] or $\alpha-\beta-\gamma$ interhelical orientations in nucleic acids,[14] cannot currently be used in this framework. Other sampling approaches combine many independent trajectory segments into a Markov model that describes the entire free energy surface of interest (such as Markov state models[15] and current implementations of mile-

Received: November 22, 2013
Revised: January 15, 2014
Published: February 3, 2014

   dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

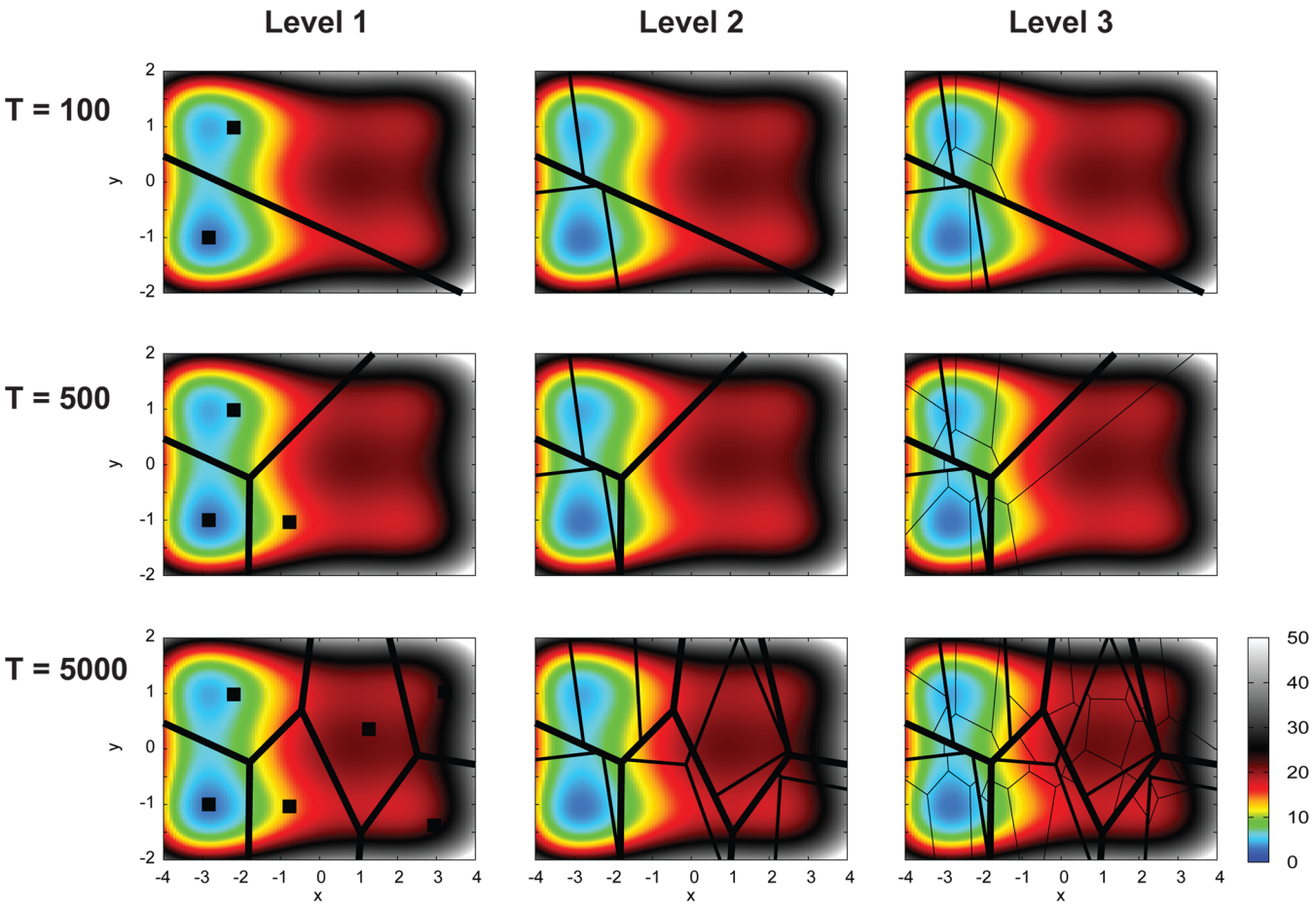

Figure 1. Region creation in WExplore. The left column shows the biggest regions (level 1), which are Voronoi polyhedra (VP) defined by a set of images, shown as black squares. Level-1 VP are defined such that every image is at least $d^2 = 4$ apart from every other level-1 image. These regions are subdivided by level-2 VP (middle column), which are defined such that they are at least $d^2 = 2$ apart from every other level-2 VP within the same level-1 region. Likewise, level-3 VP (right column) further subdivide each level-2 region. The middle and bottom rows show how sampling proceeds from left (high probability) to right (low probability) over the course of the simulation, climbing a barrier of about 20 kT. The color maps show the free energy of the two-dimensional surface, and the scale in the bottom right is in units of kT and is the same for all panels.

stoning[16−18]). Here, the assumption that state-to-state transitions are Markovian (i.e., lose memory of their initial conditions) requires small state sizes and large lag times, which can be difficult to achieve in practice.

The weighted ensemble (WE) algorithm[2] can be a desirable alternative, as it works by a different mechanism and does not suffer from the limitations described above. WE uses an ensemble of copies of the system (or "replicas"), each of which carries a weight. These replicas are periodically cloned (where their weights are split among the clones) or merged (where weights are added) but otherwise evolve under the natural dynamics of the system. The combination of these trajectories is governed by the replica weights, and does not rely on a Markovian assumption. However, applying the WE algorithm in a high-dimensional order parameter space would require a large number of regions, and a large number of replicas, which is computationally unfeasible.

Here we present a strategy to circumvent this problem, by defining regions in a hierarchical fashion, which allows for a multiscale approach: regions are small enough to enhance small fluctuations that encourage escape from deep local minima but also have a wide reach to cover distant areas of the free energy landscape separated by large configurational changes. We first describe the resulting algorithm ("WExplore") and emphasize its differences from existing versions of the weighted ensemble algorithm. We then demonstrate the accuracy and efficiency of the method by applying it to systems of increasing dimensionality: Brownian motion on a two-dimensional potential surface, an N-dimensional biased random walk, and the 10-residue peptide chignolin in explicit solvent.

METHODS

Weighted Ensemble Algorithm. In a typical molecular dynamics simulation, the system spends the vast majority of the time in a high-probability basin of attraction and only rarely exits that basin to visit low-probability regions such as transition states. The weighted ensemble algorithm[2] enhances the sampling of these low-probability states by defining a set of regions that span the space, and enforcing that all regions are sampled evenly. Multiple copies of the system ("replicas") are run in parallel, and each is given a weight with which it contributes to statistical averages. Periodically throughout the simulation, the number of replicas in each region is kept constant by merging replicas in over-represented regions and cloning replicas in under-represented regions. When two replicas A and B, with weights w_A and w_B, are merged, the resulting replica has weight $w_A + w_B$ and takes on the configuration of replica A with probability $w_A/(w_A + w_B)$, and otherwise takes on the configuration of replica B.

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

When a replica is cloned, half of its weight is given to the new replica. High-weight replicas are also broken up by cloning, and low-weight replicas are merged, to encourage replicas in the same region to have approximately equal weights. In previous works, both the number of sampling regions and the number of replicas in each active region are kept constant.[19−25] In the algorithm presented here, the number of replicas is instead allowed to vary in each region (it is usually either 0, 1, or 2), and regions are added dynamically as sampling progresses. Below, we first describe a method to achieve this using Voronoi polyhedra, and then describe the modification of this method that is necessary for efficient application using large numbers of regions: sampling within a hierarchy.

Using Dynamically Defined Voronoi Polyhedra as Sampling Regions. Voronoi polyhedra are convenient ways to tessellate spaces of arbitrary dimension, and have been used previously to define sampling regions for enhanced sampling methods.[20,24−28] A sampling space is divided into a set of Voronoi polyhedra using a set of "images" ($\mathbf{\Phi}$), which are specific values of the sampling coordinates, obtained from specific configurations of the system. A given position $\mathbf{x}$ is in the Voronoi polyhedron i if $\mathbf{x}$ is closer to $\mathbf{\Phi}_i$ than to any other image, i.e., $|\mathbf{\Phi}_i - \mathbf{x}| < |\mathbf{\Phi}_j - \mathbf{x}|$ for all $j \neq i$. In this way, a space is divided into n regions, simply by defining n points in the space to use as images. In previous applications, the number of images remains constant throughout the simulation, although these images can be free to move. Here, the number of images is increasing throughout the simulation, and each image, once defined, remains fixed in space.

Voronoi polyhedra can be defined dynamically and used for sampling in a weighted ensemble simulation as follows. An initial point is used to seed the simulation, and an image ($\mathbf{\Phi}_0$) is placed at that point. A group of replicas, all with equal weight, are initialized at the same point and run forward in time until a replica is further than a critical distance (r) from $\mathbf{\Phi}_0$ (checking in increments of τ). Once a replica (say, i) reaches this distance, another image ($\mathbf{\Phi}_1$) is placed at $\mathbf{x}_i$. Now there are two regions: a replica is in region 0 if it is closest to $\mathbf{\Phi}_0$ and in region 1 if it is closest to $\mathbf{\Phi}_1$. This is the first step where balancing can be applied; cloning and merging steps proceed until the numbers of replicas in all regions are as even as possible. Successive regions are then defined when $\min_i|\mathbf{\Phi}_i - \mathbf{x}| > r$, in other words, when a replica is at least a distance r away from all images defined so far. Practically, a limiting number of regions (N_{reg}) is set such that a new image will not be defined if N_{reg} regions already exist. In this way, a set of N_{reg} regions can be defined that are all a distance r away from each other. One does not need to know the location of the boundaries between the regions, since in order to assign an image index to a replica one only needs to determine the distances from that replica to each image in the sampling space.

Although this approach can work well for small numbers of regions, it does not scale well with increasing N_{reg} for two reasons. First, in order to make a region assignment, one needs to perform N_{reg} distance calculations. For N_{reg} = 10 000, for example, when the distance calculation is the RMSD between two structures, this can dominate the computational cost of the simulation. Second, in order to effectively balance computational effort between regions, the number of replicas needs to be at least twice the number of regions. This is also unfeasible, due to sampling limitations. We show in the next section how to overcome these two pitfalls, by defining regions within a hierarchy.

Arranging Voronoi Polyhedra in a Hierarchy. Voronoi polyhedra (VP) can be arranged in a hierarchy as follows. The VP on the first level tile the full space but in a very coarse way, such that the entire space (or at least the part of it of interest to the researcher) is tiled using a small number of images, ι_1 (see Figure 1 for an example). These VP are defined using a large critical distance (r_1), meaning that the images that define them are at least a distance r_1 away from each other. Each of these VP are themselves divided by another set of images (the second level), numbering up to a maximum of ι_2, defined with a critical distance, $r_2 < r_1$. The level-2 VP are again subdivided by level-3 images, and so on, until the bottom level of the hierarchy is reached. The region index of a replica is now determined by first finding the closest level-1 image, then finding (within that level-1 image) the closest level-2 image, and so on. The total number of sampling regions is equal to $N_{reg} = \prod_i^L \iota_i$, which for four levels of hierarchy, each with 10 images, numbers 10 000. Considering these parameters, it is easy to see how the first pitfall mentioned above is avoided. To determine a region assignment for a replica, a maximum of 40 distance comparisons are required instead of 10 000, and this benefit increases as more levels are added to the hierarchy.

This arrangement also allows for a smaller number of replicas, since balancing is also performed in a hierarchical fashion (Figure 2). Balancing is first performed among the level-1 regions, which prioritizes even sampling among configurations that are the most distant. Within each level-1 image, the populations of the level-2 images are balanced, and so on, until the bottom level of the hierarchy is reached. This procedure is implemented computationally using a self-referencing subroutine. In our implementation, merges can only occur if two replicas have the same region index at all levels of the hierarchy. This is done to only merge together replicas with similar structures, and minimize the loss of structural heterogeneity in the ensemble. In concert with this balancing procedure, we also attempt to break up high-weight replicas and merge low-weight replicas as in the original WE algorithm[2] (not shown in Figure 2). In practice, balancing is only

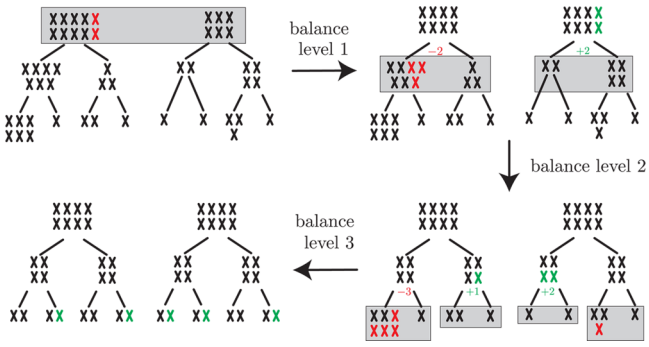

Figure 2. An example of the hierarchical balancing procedure. Here there are three levels to the sampling hierarchy, each of which has $\iota_i = 2$. Each replica is denoted by an "X". The X's at the upper levels of the hierarchy show the sum of all replicas below them in the tree. Replicas shown in red are lost (by merging) in the subsequent balancing step, and replicas shown in green are gained (by cloning) in the previous balancing step. Originally, there are a total of 10 replicas in the first level-1 region and six in the second level-1 region (top left). The first balancing step is then to remove two replicas on the left and clone two replicas on the right. This deficit (or surplus) is passed down the tree, and denoted by "−2" ("+2") in the top right panel. Balancing is then done in the second level on each branch, after first subtracting the deficit (on the left branch), or adding the surplus (on the right branch) that was determined above. The balancing then continues on each individual branch on the third level.

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

applied where possible, as many of the regions are occupied by one, or zero, replicas. The algorithm will thus not pass on a deficit to a branch of the hierarchy that cannot be accommodated. This is accomplished by keeping track of the number of "reducible" replicas (n_{red}) under each point in the tree. For instance, at the bottom level, a region that has n replicas has $n_{red} = n - 1$. At higher levels, n_{red} for a given region is equal to the sum of the n_{red} of its children in the hierarchy.

A cycle is composed of the propagation of the group of replicas forward in time by τ followed by region assignment and a balancing step, and the simulation is run for the desired number of cycles (N_{cycles}). In practice, a simulation typically is initialized using many copies of a single point. Thus, at the early stages of a simulation, only a few bottom-level images have been defined, and balancing occurs only between these images. This amplifies the effect of small fluctuations, as those replicas are then cloned, giving the system more chances to build on that fluctuation. Each time a new level is reached (such as the first level-2 image to be defined), that replica is cloned repeatedly, and up to half of the replicas are now located at that point. This is fitting, as that point is the farthest point from the initial configuration sampled so far, and by cloning it, we are giving the simulation more chances to explore this newly discovered territory. The critical distances associated with each level should be defined carefully, as region definition should be difficult (as to not use up the regions too quickly), but still attainable.

Chignolin Simulation. The trajectory segments are run using CHARMM,[29] with the CHARMM22 force field with CMAP corrections. An explicit solvent of 2801 TIP3P waters is used, along with two sodium ions to neutralize the system. Periodic boundary conditions in a cubic box with sides of length 43.9 Å are used, which ensures that the chignolin molecule does not interact with its image, even when it is fully unfolded. SHAKE is used to constrain the hydrogens along their covalent bonds, with a tolerance of 10^{-8}. Particle-mesh Ewald summation is used for nonbonded interactions with a switching function that scales the nonbonded interactions to zero at 11 Å, starting from 8 Å. Langevin dynamics and a leapfrog integrator are used, with a time step of 2 fs at the temperature 300 K. The first model of PDB structure 1UAO[30] is used as a starting structure. After solvation, the solvent and ions are minimized using 500 steps of steepest descent and 500 steps of the adopted basis Newton−Raphson method. The entire system is then minimized in the same way. The system is then heated, with harmonic restraints on the protein, from 40 to 300 K over 50 000 time steps. The constraints are then removed, and dynamics are run for 150 000 time steps for equilibration. The resulting structure is used as an initial condition for all 48 replicas.

Creating Network Graphs. We build the network graphs using the program Gephi 0.8.2, following the protocol in our previous work.[31,32] However, the states used here to build the graphs are the hierarchically organized set of images used in the WExplore algorithm during sampling, rather than a set of independently determined clusters. The size of the nodes is proportional to their statistical populations; however, for each graph, there is a minimum node size that is 30 times smaller than the size of the largest node. The orientation of the nodes is obtained using a force minimization algorithm built into Gephi (ForceAtlas), which introduces a repulsive force between all nodes but attracts nodes that are linked together with a force that is proportional to the weights of the links. The weights of the links are determined as follows. First, weights of directed links with values between 1 and 1000 are determined as $w_{ij} = 1000p_{ij}$, where p_{ij} is the transition probability from i to j. Weights of undirected links are then determined as the average of the two directed links. The graph is first allowed to minimize without adjusting for node sizes (i.e., with overlapping nodes), and then, a second minimization is subsequently performed with adjusting for node sizes to prevent overlap.

1.2 μs Conventional Simulation on GPU. For comparison purposes, we ran a single long simulation using conventional molecular dynamics using the OpenMM-CHARMM interface. OpenMM is a library for molecular simulation that can utilize a broad range of hardware architectures, including graphics processing units. Here we use OpenMM 4.1,[33] and the simulation and equilibration details are the same as those run in CHARMM, with the following exception: dynamics are run in the constant pressure ensemble with a barostat coupled to a reference pressure of 1 atm, and volume moves are attempted every 25 steps.

RESULTS AND DISCUSSION

Two-Dimensional Model Potential. To demonstrate the ability of WExplore to climb free energy barriers and to visualize the region creation process, we examine a two-dimensional model potential with one fast and one slow degree of freedom. The potential energy surface is given by

$$V(x, y) = \alpha_1(x^4 - \eta_1 x^2 + \gamma_1 x) + \alpha_2(y^4 - \eta_2 y^2 + \gamma_2 y) \quad (1)$$

such that there is no coupling between the x and y variables. There are two basins along each coordinate: the η parameters control the height of the barriers separating the basins, and the γ parameters control the relative stability of the basins. We employ a large barrier along the x coordinate to make it our slow degree of freedom, and make the $x < 0$ basin significantly more stable than the $x > 0$ one: $\alpha_1 = 0.15$, $\eta_1 = 12.5$, and $\gamma_1 = 20$, such that the $x > 0$ basin has a probability of $\sim 1 \times 10^{-6}$. For the fast degree of freedom (y), we use $\alpha_2 = 2.5$, $\eta_2 = 2$, and $\gamma_2 = 0.25$. The system evolves according to the equation of motion:

$$\mathbf{r}(t + \delta t) = \mathbf{r}(t) - D\beta\delta t\nabla_{\mathbf{r}}V + \delta\mathbf{r}^G \quad (2)$$

where δt is the numerical integration time step, $\delta\mathbf{r}^G$ is a random noise term with components chosen randomly from a Gaussian distribution with zero mean and variance $2D\delta t$, and $\mathbf{r} = (x, y)$. D is the diffusion coefficient, and β is the inverse temperature, which take the values 0.1 and 1, respectively.

The WExplore simulations use 48 replicas, with regions described using a hierarchy of three levels with region sizes of $d^2 = 4$, 2, and 1, where distances are calculated as $d^2 = (x - x_i)^2 + (y - y_i)^2$. Each cycle is composed of 1000 dynamics steps followed by replica balancing. A total of 10 simulations are run, each using a total of 50 000 cycles. For an efficiency comparison, we also run a set of 10 simulations with replica balancing turned off, where each simulation is equivalent to running 48 trajectories of 5×10^7 steps each. These are labeled "CONV" for conventional sampling using Brownian dynamics.

Figure 1 shows how the sampling regions at the three levels of the hierarchy change as the simulation progresses. All the trajectories begin from the same initial point (in this case, the lowest energy minimum at $(x, y) = (-2.84, -1)$), and an image is defined at that point on all sampling levels. After 100 cycles (top row), sampling is still confined to the two lowest energy basins, and there are two level-1 images defined (shown as squares). These separate the map into two Voronoi polyhedra (VP) which

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

are separated by a thick line. The level-1 VP are further subdivided by level-2 VP (medium-thick lines, center column), which are again subdivided by level-3 VP (thin lines, right column); the images that define these VP are not shown for clarity. As sampling progresses, more regions are defined by the sampling, until the bulk of the accessible space is covered (lower right panel). Sampling continues after regions are no longer being discovered, until the time-averaged weights of the replicas converge in each region.

This convergence can be observed using projections of the free energy onto the x direction (Figure 3). Figure 3A shows probability distributions along the slow degree of freedom as a function of sampling time. WExplore samples the right basin after 500 cycles; however, not enough weight has diffused into the $x = 2$ basin, resulting in an under-prediction of its statistical weight. Further sampling allows more weight to diffuse to the $x = 2$ basin, and after 50 000 cycles, the predicted probability distribution closely matches the analytically determined target. We quantify this convergence using an error metric, $E(t) = \langle |\log(P_a(x)) - \log(P_{obs}(x))| \rangle_x$, which averages the error in the logarithm of the probability over all points in the analytically determined histogram that have a weight greater than 1×10^{-9}. If the observed histogram (P_{obs}) equals zero at position x, we instead use the value of the minimum probability of allowed replicas in our system, which is 1×10^{-40}. Figure 3B shows this error as a function of time in comparison with the error from a conventional simulation (CONV), which performs poorly due to lack of resolution in low probability regions. The bottom panel of Figure 3 shows probability distributions in the x-direction when the simulation is initialized instead in the $x = 2$ basin. The population in the $x = 2$ basin is initially over-represented, and this over-representation decays over 25 000 cycles to the target distribution. This demonstrates that WExplore can be effective even if the starting conditions are not representative of the equilibrated ensemble.

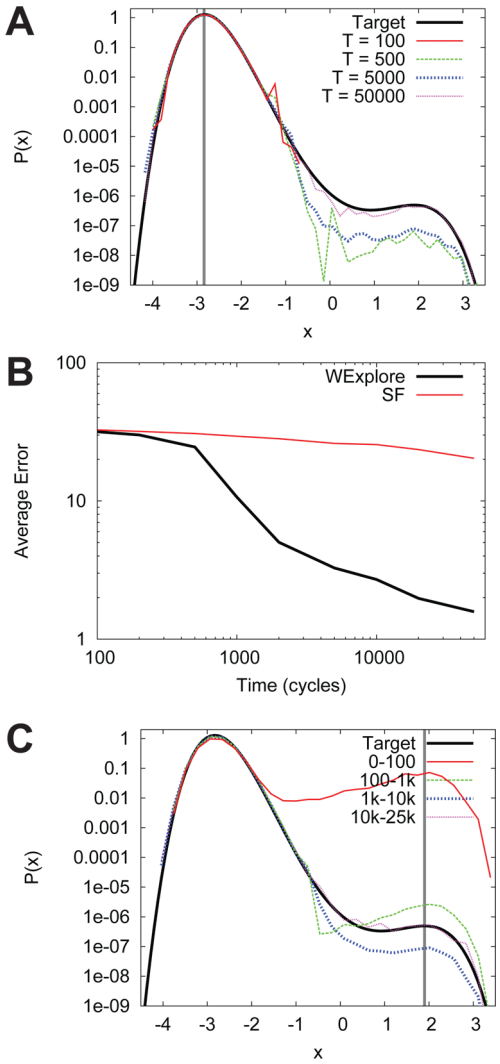

Figure 3. Convergence toward the analytical result for the two-dimensional model system. (A) Histograms of the x coordinate at different time points in the WExplore simulation. The time is in units of sampling cycles, which are 1000 steps each. The vertical line shows the starting position. (B) Error as a function of sampling time for both WExplore and conventional sampling (CONV). CONV is implemented using the same code as WExplore but with replica balancing turned off. Error is defined as in the main text. (C) Same as in part A but starting instead around $x = 2$, with the starting point shown by the vertical line. The histograms are averaged over the intervals shown in the legend, in order to exclude effects of the initial conditions.

***N*-Dimensional Biased Random Walk.** To study the performance of the algorithm in higher dimensional spaces, we use a system with a tunable dimensionality: a biased random walk in N dimensions. The position of the "walker" is described by an N-dimensional vector of non-negative integers ($\mathbf{X}$). In each dynamics step, for each dimension i, x_i increases with probability $P_u = 0.25$, and decreases with probability $1 - P_u = 0.75$ for all $x_i > 0$. The system is bound at 0 for all dimensions: moves to $x_i = -1$ are rejected.

Ten WExplore simulations are performed at each of $N = 2$, 5, 10, and 20 with the same parameters. The regions are described using a hierarchy of four levels with region sizes of $d = 0.25$, 1, 4, and 16, where distances are defined using the Manhattan norm: $d = (1/N)\sum_{i=1}^{N} |x - x_i|$. We use a maximum branch number of 10 for each level, which allows for a maximum of 10 000 regions. Balancing of the set of $N_{rep} = 200$ replicas is performed every 10 dynamics steps, and each simulation is run for 10 000 cycles. We again run a set of simulations with replica balancing turned off for comparison (CONV). For low values of N, we expect the regions to be able to densely tile the space, and allow for good sampling at large values of x. However, as N increases, the benefits of enhanced sampling should decrease, as more regions are clustered close to the origin.

Since all of the dimensions in this system are interchangable, we average over projections onto each degree of freedom in the system, and show them for each value of N in Figure 4. For instance, the $N = 20$ curve is the average of 200 histograms (10 simulations, each with 20 projections). As N increases from 2 to 20, the range of x values visited and the accuracy of the predicted probability both decrease. To quantify the accuracy, we use the measure $A = \sum_x a(x)$, where

$$a(x) = \begin{cases} 1 + \dfrac{|\log(P^t(x)) - \log(P(x))|}{\log(P^t(x))} & \text{if } \log(P(x)) > 2\log(P^t(x)) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where $P(x)$ is the predicted probability of position x and $P^t(x) = (2/3)(1/3)^x$ is the target probability. $a(x)$ is equal to a maximum of 1 when $P(x) = P^t(x)$ and a minimum of zero when either $P(x)$ is undefined or $P(x)$ is far from the target value. Figure 5 compares the measured A, as well as the range of x values visited,

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

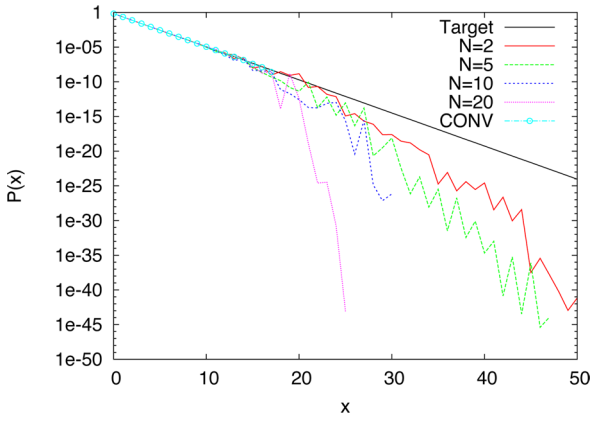

Figure 4. Histograms for the N-dimensional random walk as computed by WExplore and CONV. The curves are averages over all dimensions in the system, as well as all 10 simulations performed at each value of N. The black line is a target curve computed analytically.

between WExplore and CONV for the four values of N. As expected, as N increases, the benefits of WExplore over CONV decrease. For $N = 2$ and $N = 5$, we see significantly larger values of A, indicating accurate sampling over a wide range of x. However, for $N \geq 10$, the expected value of A is the same with and without enhanced sampling. Similar results are found for the range of sampled x values.

These results suggest that WExplore is not efficient for sampling spaces composed of more than $N = 10$ independent degrees of freedom. However, although small-scale motions can occur along every degree of freedom in the sampling space, usually large-scale motions can be projected onto a much smaller subspace of collective variables. In these cases, WExplore is capable of handling a larger number of dimensions in its sampling space. To illustrate this concept, we break the degrees of freedom in the N-dimensional random walk into "soft" and "hard" degrees of freedom. Hard degrees of freedom have $P_u = 0.95$, and soft degrees of freedom have $P_u = 0.75$, as above. This way, over the course of the simulation, large-scale motions will mostly occur in the "soft" subspace, and small-scale motions in the "hard" subspace will mostly act as noise for the distance calculation. This mimics the biomolecular scenario where there are many fast motions over small distances, such as vibrations and rotations, but relatively few soft degrees of freedom that cause large molecular distortions, such as unfolding, interdomain motions, and secondary structure rearrangements. We run a series of simulations with 5 slow degrees of freedom and vary the total dimensionality, N. The benefits of enhanced sampling here are now extended up to $N = 50$ (Figure 6).

Although these results do not have direct implications for biomolecular systems, they provide a starting point for discussion. First, the values of P_u for both the hard and soft degrees of freedom are important, because they determine the size of fluctuations along each dimension. The distance metric used to decide when new regions should be created is a sum of terms over each degree of freedom. In order for sampling regions to be placed along the soft degrees of freedom (thus enhancing sampling in that direction), the size of slow fluctuations needs to be comparable to the sum of the sizes of the hard fluctuations, which grows, in this case, with the total dimensionality. This motivates the need for a careful selection of variables to be included in the sampling subspace; it is best to use small sampling spaces, ideally composed only of soft degrees of freedom, that is, those associated with the large conformation changes one wishes to observe in the simulation.

Chignolin in Explicit Solvent. Chignolin is a 10-residue peptide that folds into a β-hairpin structure, which was designed to be maximally stable on the basis of statistics of known protein segments.[30] It has served as an excellent system for theoretical study,[34−37] since it is one of the smallest molecules that exhibits a stable secondary structure. The folded structure of the molecule consists of a central turn region (residues 4−7) flanked by three amino acids on each side, which form the stem of the hairpin, which is stabilized by hydrogen bonding interactions. Two main folding mechanisms for hairpins of this type have been proposed: the "zipper" model,[38] where the turn region forms first and stem residues come into contact successively, moving outward from the turn, and the "hydrophobic collapse" model,[39] where there is an initial, fast collapse to a compact structure, followed by slow rearrangements resulting in the hairpin structure. Enemark and Rajagopalan[34] observed the folding of chignolin in ten 1 μs trajectories, and found folding to occur predominantly by a "broken-zipper" mechanism, where stem residues come into contact out of order; residues 1 and 10 come into contact earliest. Suenaga et al.,[36] in contrast, observed that interactions between the aromatic rings of Tyr2 and Typ9 initiate the folding process. Using WExplore, we build a configuration space network for chignolin that connects the folded state to the unfolded state through an ensemble of folding pathways. We then characterize the intermediate states based on the distance between the

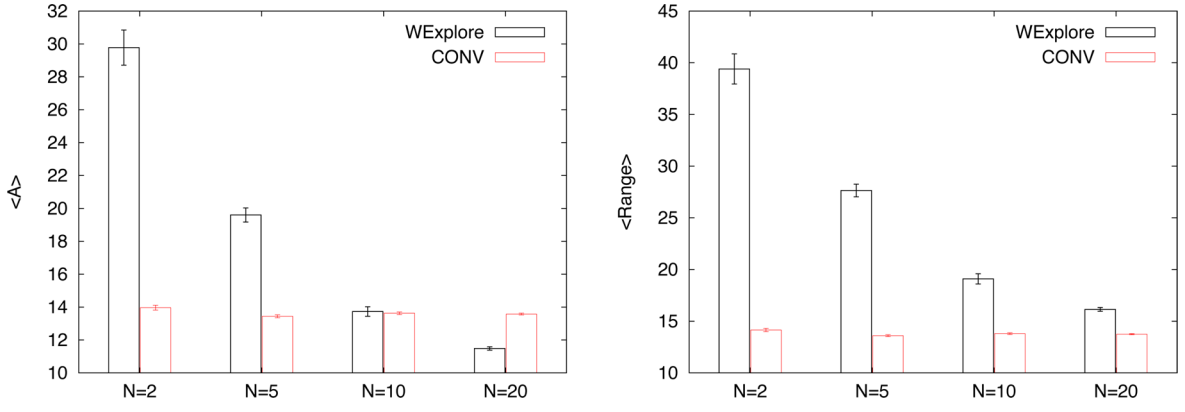

Figure 5. Left: Average values of the measure A, described in the text. Right: Average range of x values visited. The error bars are standard errors computed across all dimensions and all 10 simulations for each N.

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

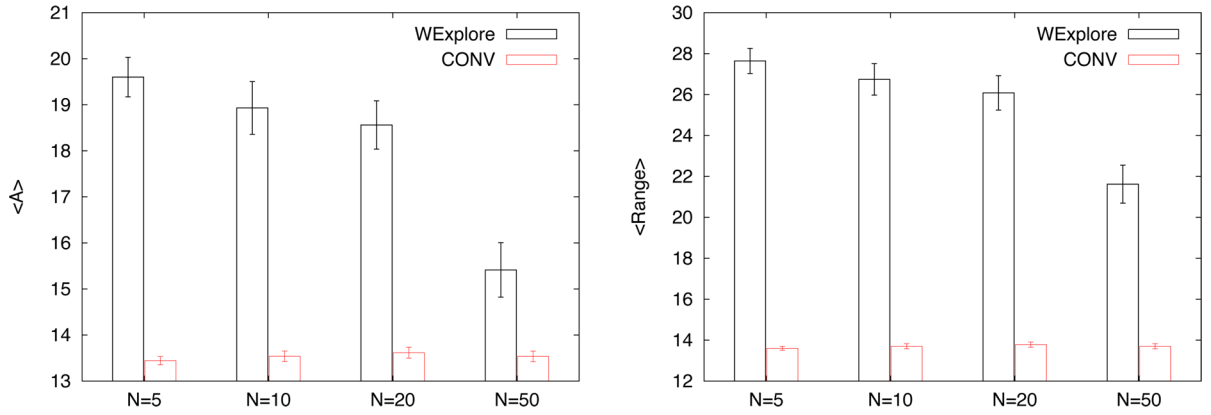

Figure 6. Average values of A and the range of x values for systems with 5 slow degrees of freedom with varying N. The error bars are standard errors computed across all dimensions and all 10 simulations for each N.

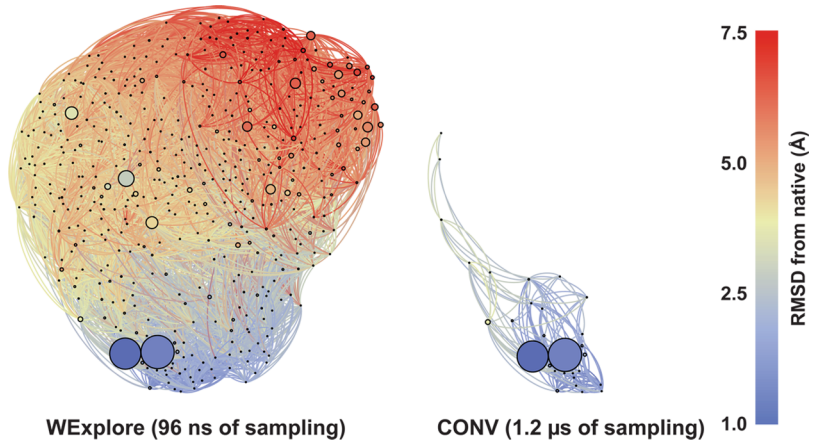

Figure 7. Comparison of sampling ranges from WExplore (phase I) and a conventional simulation. The sampling range for each simulation is shown using the full configuration space network for chignolin, where only the nodes visited in that simulation are visible. (left) Regions visited by WExplore in phase I, using a combined 96 ns of sampling. (right) Regions visited by a single long conventional MD trajectory, using 1.2 μs of sampling. In each network, the nodes are colored by the Cα RMSD to the native state, and the sizes of the nodes are the equilibrium statistical weight (as determined by the complete WExplore simulation). The edges shown in each network are determined by the complete WExplore simulation.

hydrophobic residues (Tyr2 and Trp9), the distances between many native and non-native hydrogen bonding pairs, and the distance between the Cα atoms of each residue pair in the stem.

The details of the molecular dynamics simulation are given in the Methods section. For WExplore, we use the Cα RMSD as a distance metric. Four levels of hierarchy are used with cutoff distances of 1, 2, 4, and 6 Å. At each level of hierarchy, the maximum number of images under each parent is 10, allowing for a total of up to 10^4 images. We use 48 replicas distributed over 48 processors. The simulation proceeds in three phases. First, we perform 1000 cycles of time length τ = 2 ps, with a replica balancing procedure performed at the end of each cycle (phase I); 522 regions are defined, covering native, fully unfolded, and intermediate regions of phase space. In phase II, we then perform a simulation without defining new regions to collect region-to-region crossing statistics, which is comprised of another 1000 cycles. We then build a transition matrix $\mathbb{T}(\tau)$, the element $t_{ij}(\tau)$ of which is the probability, given initialization in region i, of being in region j after a time τ. The largest eigenvalue of this matrix is 1, and it corresponds to a unique solution of the equilibrium weights. Note however that this procedure is akin to making an assumption of Markovian dynamics over the time period τ, which is not strictly valid, and thus the weights obtained are not exact. We thus use these weights to initialize a new simulation with one replica in each region with a nonzero weight, giving us 418 replicas. The simulation is continued for another ~1200 cycles of time length 10τ (phase III), until the relative weights of the folded and unfolded states converged. The total amount of sampling across the set of replicas is 96, 96, and 10 032 ns in phases I, II, and III, respectively, or 10.2 μs total.

A full configuration space network (CSN) of chignolin is shown in Figure 7 (see the Methods section for details on CSN construction). The states used in the network are the hierarchically organized images used in the WExplore algorithm, and we note that the algorithm provides all the information needed to generate a CSN after phase II. The size of the nodes in the network is proportional to the weight of each state, and the color here of each node is the Cα RMSD of each state to the native configuration; the color of an edge is the average color of the two nodes it connects. The two largest, blue nodes in the lower left are the native state, and the red states in the upper right are the farthest from native. Similar to a previously determined CSN of a chignolin mutant (CLN025),[32] the unfolded state is not significantly partitioned, and forms a homogeneous mass with no noticeable outcroppings. The 522 states in the network, from folded to fully unfolded, are found in 96 ns of sampling by WExplore, while a much smaller set, with lower RMSD to native, is found by a long 1.2 μs trajectory run on graphics processing

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

units (see the Methods section for GPU simulation details). Note that, although all the states are found in 96 ns, a much longer sampling time is required to obtain a set of equilibrated weights for each state.

In Figure 8, we show convergence of the fraction of population in the native state. At the end of phase II, before the reweighting

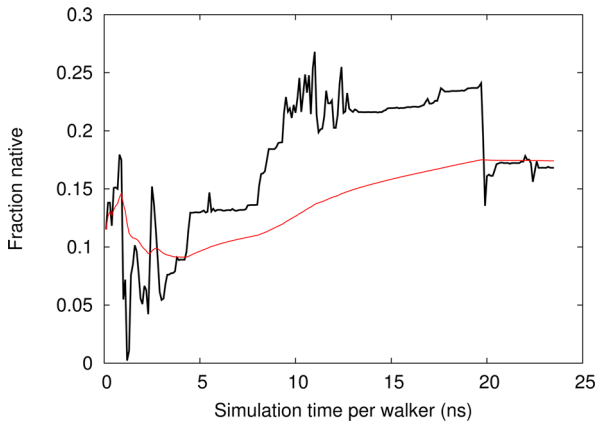

Figure 8. The fraction of population in the folded state as a function of simulation time. This is shown over phase III of the simulation: time 0 corresponds to the reweighting of the replicas, and the fraction native prior to reweighting is approximately 1. A rolling average of replica weights over five cycles is shown (black line), as well as a running average (red line). A replica is considered native if it is in a region with a Cα RMSD less than 2.5 Å to the native state.

procedure, the vast majority of the weight is in the folded state, so the native fraction approximately equals 1. The reweighting procedure initially overcompensates and assigns 12% of the weight to the native ensemble. This error is due to non-Markovian dynamics over the short lag time used here. Over the course of phase III, there is a shift in weight toward the folded state, which levels off at 17.4% at the end of the simulation. This is a factor of 3 lower than experimental results obtained from circular dichroism melting curves (60% native at 300 K).[30]

To investigate the folding mechanism, we project the Cα−Cα distances of the stem pairs (4−7, 3−8, 2−9, and 1−10) onto the network (Figure 9). In the bottom right panel, the 1−10 distance is shown. From this graph, we can see that only the two native-like states have low Cα−Cα distances for the 1−10 pair. This implies that these residues are the last to come together when folding and the first to come apart when unfolding, which is in direct conflict with the results of Enemark et al.[34] It is possible that the OPLS-AA force field used in their work demonstrates different behavior than the CHARMM22 force field used here. As we move higher up the stem, nodes change from red to blue in a monotonic fashion, indicating a zipper-like mechanism. To test this more directly, we use a broken-zipper metric, $BZ = b_{12} + b_{13} + b_{14} + b_{23} + b_{24} + b_{34}$, where

$$b_{ij} = \begin{cases} 0 & \text{if } d_i < d_j \\ d_i - d_j & \text{otherwise} \end{cases} \tag{4}$$

where d_1 is the 4−7 Cα−Cα distance, d_2 is 3−8, d_3 is 2−9, and d_4 is 1−10. BZ = 0 indicates no deviation from the zipper model, and BZ > 0 quantifies broken-zipper deviations in Å. BZ is projected onto the CSN in Figure 10. We do not find significant broken-zipper deviations: only 9 of 522 states have BZ > 1 Å. The largest weighted of such states is shown in the figure; the largest contribution to BZ in this case is b_{23} = 1.7 Å.

We further study the order of folding events by projecting distances of native and non-native hydrogen bonds onto the CSN (Figure 11). A blue color indicates that the atoms are close enough that a hydrogen bond (HB) could form between them. Some HBs are only formed in the native state, such as those between residues 1 and 10. In agreement with previous findings, the 3O−7N HB is formed in a wide range of non-native states, indicating that it is important to the folding mechanism.[35,40] 3N−8O, in contrast, is formed only close to the native state. Our

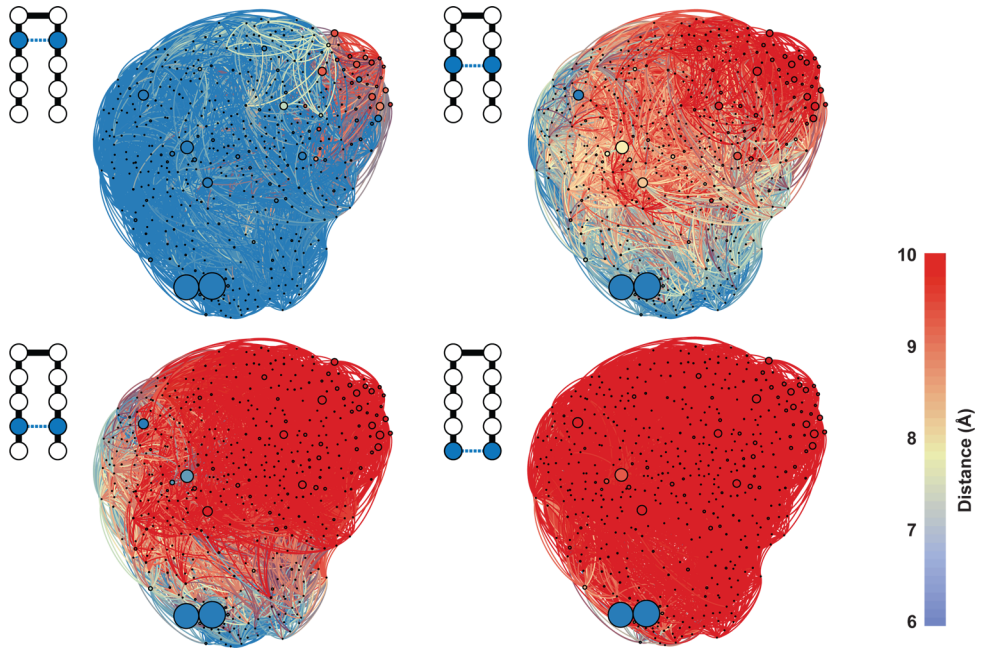

Figure 9. Cα−Cα distances projected onto the CSN. The schematic accompanying each network shows the residues involved: 4−7, 3−8, 2−9, and 1−10 for the top left, top right, bottom left, and bottom right, respectively. For comparison, the distances are only shown in the range 6−10 Å.

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

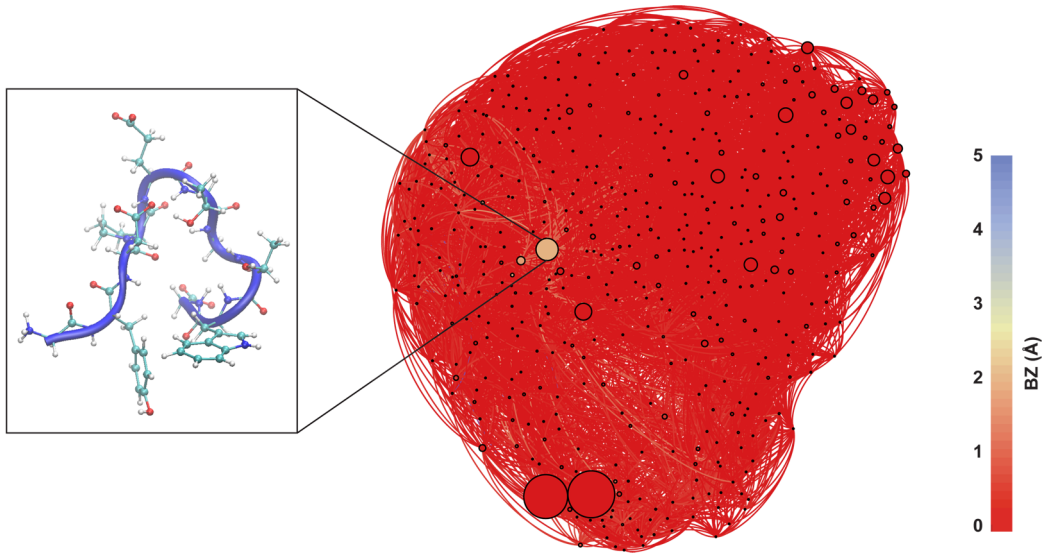

Figure 10. The broken-zipper metric (BZ) projected onto the CSN. There are only a handful of positive BZ states, the highest weighted of which is shown.

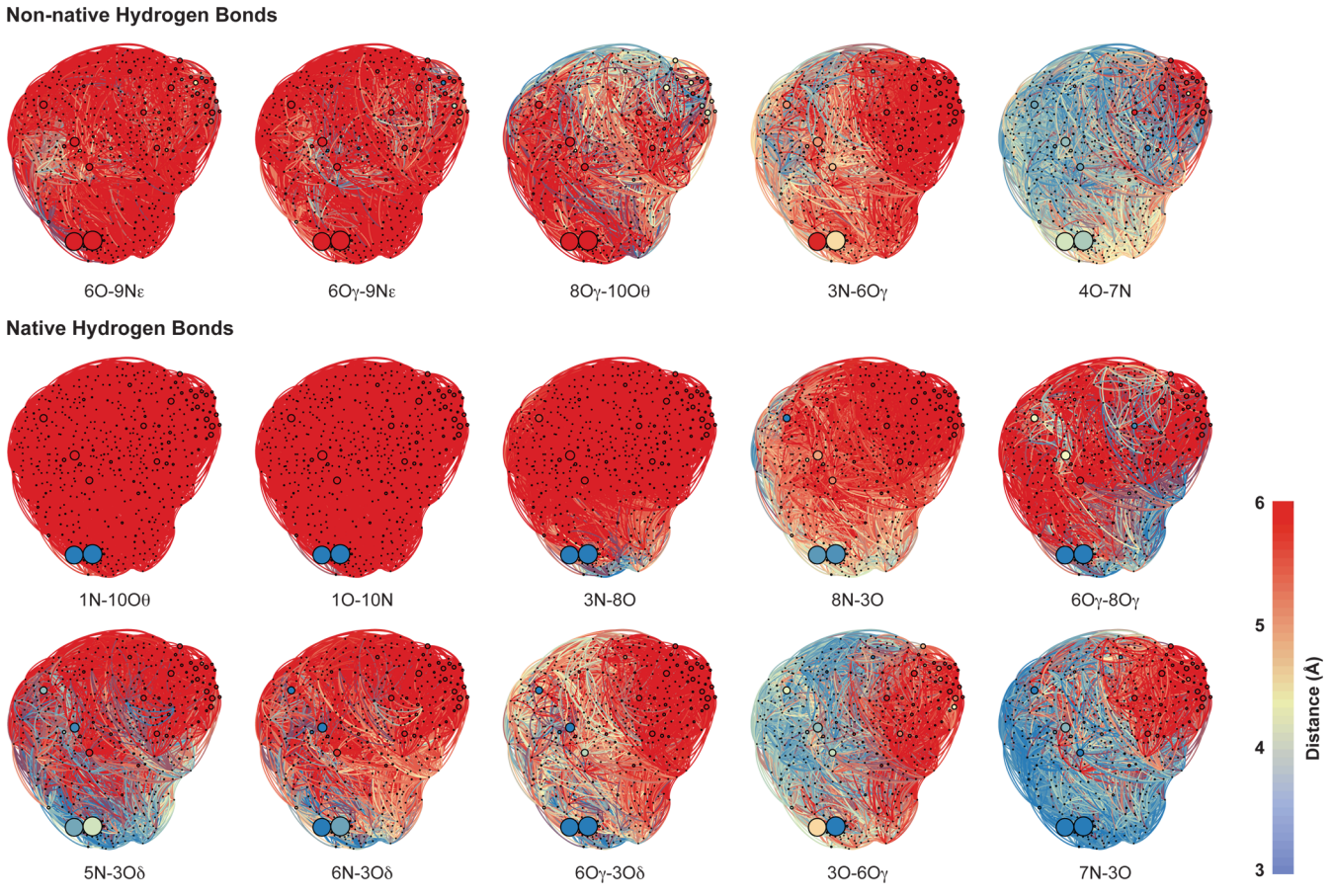

Figure 11. Native and non-native hydrogen bonding distances projected onto the CSN. The color map for each distance is restricted to the range 3−6 Å for easy comparison. Blue indicates that the two atoms are close enough to form a hydrogen bond in that state.

results suggest that 4O−7N, which is not present in the native state, might play an important role in stabilizing the turn, as it is formed in the majority of unfolded structures. 3O−6Oγ could play a similar role, as it is also strongly represented in the unfolded ensemble. Finally, we show a map of the loop−loop distances between Tyr2 and Trp9 (Figure 12). The distance is between the centers of mass of the heavy atoms of each side chain. Interestingly, there are two highly weighted states in the unfolded ensemble with low loop−loop distances (one of which is the broken-zipper state previously identified). These states have collapsed configurations but lack key native hydrogen bonds needed to stabilize the structure (such as 3N−8O).

dx.doi.org/10.1021/jp411479c | J. Phys. Chem. B 2014, 118, 3532−3542

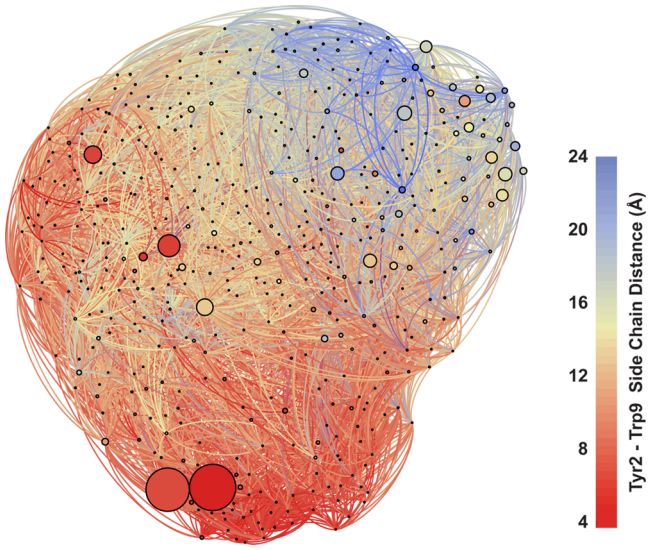

Figure 12. Loop−loop distance between Tyr2 and Tyr9 projected onto the CSN. The color bar here extends from the minimum to the maximum observed values (4 and 24 Å) of that distance.

CONCLUSION

The above results demonstrate the utility of WExplore to generate configurational ensembles that can span large free energy barriers. It requires only a starting configuration and the definition of a sampling space, and is hence useful when limited information is known about the system. We found that the weight equilibration step (phase III) is approximately 100 times more expensive than the initial exploration step for chignolin, and this suggests that WExplore would be especially powerful within a sample-and-select framework, where knowledge of the weights is not necessary.

The ensembles generated with this method can be used in a variety of contexts. For instance, in pK_a calculations, conformations with low statistical weights (such as those that expose titratable residues) can have outsized importance. They can be useful in drug development, to provide additional drug targets for ligands to bind via conformational selection. An ensemble-based description is necessary for proteins and complexes with intrinsic disorder, and WExplore would likely be more efficient than building ensembles via conventional simulation.

As demonstrated for chignolin, WExplore is easily combined with a configuration space network analysis, since it has states and rates built into the algorithm. If a more quantitative model is required (such as a Markov state model), snapshots can be saved over the course of the algorithm (at the beginning and end of each trajectory segment), and these can be clustered using more conventional approaches. However, the downside is that the lag time (τ) used in WExplore would be inflexible; it would be best to make sure τ is equal to or greater than the desired lag time in the resulting Markov model, and this could affect the efficiency of the WExplore simulation. Alternatively, the images discovered using the initial exploration step of WExplore could be used as a set of starting points for Markov state model generation. As these images are guaranteed to be kinetically accessible from the starting state, this strategy could be more efficient than seeding MSM simulations using high-temperature or denatured states.

AUTHOR INFORMATION

Corresponding Author

*E-mail: brookscl@umich.edu.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

We acknowledge support from the Center for Theoretical Biological Physics (CTBP, https://ctbp.ucsd.edu/) funded by the NSF (PHY0216576).

REFERENCES

(1) Torrie, G. M.; Valleau, J. P. Non-Physical Sampling Distributions in Monte-Carlo Free-energy Estimation - Umbrella Sampling. *J. Comput. Phys.* **1977**, *23*, 187−199.
(2) Huber, G. A.; Kim, S. Weighted-Ensemble Brownian Dynamics Simulations for Protein Association Reactions. *Biophys. J.* **1996**, *70*, 97−110.
(3) Faradjian, A. K.; Elber, R. Computing Time Scales from Reaction Coordinates by Milestoning. *J. Chem. Phys.* **2004**, *120*, 10880.
(4) Allen, R. J.; Warren, P. B.; ten Wolde, P. R. Sampling Rare Switching Events in Biochemical Networks. *Phys. Rev. Lett.* **2005**, *94*, 018104.
(5) Warmflash, A.; Bhimalapuram, P.; Dinner, A. R. Umbrella Sampling for Nonequilibrium Processes. *J. Chem. Phys.* **2007**, *127*, 154112.
(6) E, W.; Ren, W.; Vanden-Eijnden, E. String Method for the Study of Rare Events. *Phys. Rev. B* **2002**, *66*, 052301.
(7) E, W.; Vanden-Eijnden, E.; Ren, W. Finite Temperature String Method for the Study of Rare Events. *J. Phys. Chem. B* **2005**, *109*, 6688−6693.
(8) Hansmann, U. Parallel Tempering Algorithm for Conformational Studies of Biological Molecules. *Chem. Phys. Lett.* **1997**, *281*, 140−150.
(9) Sugita, Y.; Okamoto, Y. Replica-Exchange Molecular Dynamics Method for Protein Folding. *Chem. Phys. Lett.* **1999**, *314*, 141−151.
(10) Laio, A.; Parrinello, M. Escaping Free-Energy Minima. *Proc. Natl. Acad. Sci. U.S.A.* **2002**, *99*, 12562−12566.
(11) Maragliano, L.; Vanden-Eijnden, E. A Temperature Accelerated Method for Sampling Free Energy and Determining Reaction Pathways in Rare Events Simulations. *Chem. Phys. Lett.* **2006**, *426*, 168−175.
(12) Wu, X.; Brooks, B. R. Self-Guided Langevin Dynamics Simulation Method. *Chem. Phys. Lett.* **2003**, *381*, 512−518.
(13) Zhang, Y.; Skolnick, J. Scoring Function for Automated Assessment of Protein Structure Template Quality. *Proteins* **2004**, *57*, 702−710.
(14) Bailor, M. H.; Mustoe, A. M.; Brooks, C. L., III; Al-Hashimi, H. M. 3D Maps of RNA Interhelical Junctions. *Nat. Protoc.* **2011**, *6*, 1536−1545.
(15) Chodera, J. D.; Swope, W. C.; Pitera, J. W.; Dill, K. A. Long-Time Protein Folding Dynamics from Short-Time Molecular Dynamics Simulations. *Multiscale Model. Simul.* **2006**, *5*, 1214.
(16) Vanden-Eijnden, E.; Venturoli, M. Markovian Milestoning with Voronoi Tessellations. *J. Chem. Phys.* **2009**, *130*, 194101.
(17) Májek, P.; Elber, R. Milestoning Without a Reaction Coordinate. *J. Chem. Theory Comput.* **2010**, 1805−1817.
(18) Kreuzer, S. M.; Elber, R. Coiled-Coil Response to Mechanical Force: Global Stability and Local Cracking. *Biophys. J.* **2013**, *105*, 951−961.
(19) Zhang, B. W.; Jasnow, D.; Zuckerman, D. M. Efficient and Verified Simulation of a Path Ensemble for Conformational Change in a United-Residue Model of Calmodulin. *Proc. Natl. Acad. Sci. U.S.A.* **2007**, *104*, 18043−18048.
(20) Zhang, B. W.; Jasnow, D.; Zuckerman, D. M. The "Weighted Ensemble" Path Sampling Method is Statistically Exact for a Broad Class of Stochastic Processes and Binning Procedures. *J. Chem. Phys.* **2010**, *132*, 054107.

(21) Bhatt, D.; Zhang, B. W.; Zuckerman, D. M. Steady-State Simulations using Weighted Ensemble Path Sampling. *J. Chem. Phys.* **2010**, *133*, 014110.
(22) Adelman, J. L.; Dale, A. L.; Zwier, M. C.; Bhatt, D.; Chong, L. T.; Zuckerman, D. M.; Grabe, M. Simulations of the Alternative Access Mechanism of the Sodium Symporter Mhp1. *Biophys. J.* **2011**, *101*, 2399−2407.
(23) Zwier, M. C.; Kaus, J. W.; Chong, L. T. Efficient Explicit-Solvent Molecular Dynamics Simulations of Molecular Association Kinetics: Methane/Methane, Na+/Cl-, Methane/Benzene, and K+/18-Crown-6 Ether. *J. Chem. Theory Comput.* **2011**, *7*, 1189−1197.
(24) Bhatt, D.; Bahar, I. An Adaptive Weighted Ensemble Procedure for Efficient Computation of Free Energies and First Passage Rates. *J. Chem. Phys.* **2012**, *137*, 104101.
(25) Adelman, J. L.; Grabe, M. Simulating Rare Events using a Weighted Ensemble-Based String Method. *J. Chem. Phys.* **2013**, *138*, 044105.
(26) Vanden-Eijnden, E.; Venturoli, M. Revisiting the Finite Temperature String Method for the Calculation of Reaction Tubes and Free Energies. *J. Chem. Phys.* **2009**, *130*, 194103.
(27) Dickson, A.; Warmflash, A.; Dinner, A. R. Nonequilibrium Umbrella Sampling in Spaces of Many Order Parameters. *J. Chem. Phys.* **2009**, *130*, 074104.
(28) Dickson, A.; Warmflash, A.; Dinner, A. R. Separating Forward and Backward Pathways in Nonequilibrium Umbrella Sampling. *J. Chem. Phys.* **2009**, *131*, 154104.
(29) Brooks, B. R.; Brooks, C. L., III; Mackerell, A. D., Jr.; Nilsson, L.; Petrella, R. J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; et al. CHARMM: The Biomolecular Simulation Program. *J. Comput. Chem.* **2009**, *30*, 1545−1614.
(30) Honda, S.; Yamasaki, K.; Sawada, Y.; Morii, H. 10 Residue Folded Peptide Designed By Segment Statistics. *Structure* **2004**, *12*, 1507−1518.
(31) Dickson, A.; Brooks, C. L., III. Quantifying Hub-Like Behavior in Protein Folding Networks. *J. Chem. Theory Comput.* **2012**, *8*, 3044−3052.
(32) Dickson, A.; Brooks, C. L., III. Native States of Fast-Folding Proteins are Kinetic Traps. *J. Am. Chem. Soc.* **2013**, *135*, 4729−4734.
(33) Eastman, P.; Friedrichs, M. S.; Chodera, J. D.; Radmer, R. J.; Bruns, C. M.; Ku, J. P.; Beauchamp, K. A.; Lane, T. J.; Wang, L.-P.; Shukla, D.; et al. OpenMM 4: A Reusable, Extensible, Hardware Independent Library for High Performance Molecular Simulation. *J. Chem. Theory Comput.* **2013**, *9*, 461−469.
(34) Enemark, S.; Rajagopalan, R. Turn-Directed Folding Dynamics of β-Hairpin-Forming de novo Decapeptide Chignolin. *Phys. Chem. Chem. Phys.* **2012**, *14*, 12442−12450.
(35) Harada, R.; Kitao, A. Exploring the Folding Free Energy Landscape of a β-Hairpin Miniprotein, Chignolin, Using Multiscale Free Energy Landscape Calculation Method. *J. Phys. Chem. B* **2011**, *115*, 8806−8812.
(36) Suenaga, A.; Narumi, T.; Futatsugi, N.; Yanai, R.; Ohno, Y.; Okimoto, N.; Taiji, M. Folding Dynamics of 10-Residue Beta-Hairpin Peptide Chignolin. *Chem.—Asian J.* **2007**, *2*, 591−598.
(37) Seibert, M. M.; Patriksson, A.; Hess, B.; Van Der Spoel, D. Reproducible Polypeptide Folding and Structure Prediction using Molecular Dynamics Simulations. *J. Mol. Biol.* **2005**, *354*, 173−183.
(38) Muñoz, V.; Thompson, P. a.; Hofrichter, J.; Eaton, W. A. Folding Dynamics and Mechanism of Beta-Hairpin Formation. *Nature* **1997**, *390*, 196−199.
(39) Dinner, A.; Lazaridis, T.; Karplus, M. Understanding β-Hairpin Formation. *Proc. Natl. Acad. Sci. U.S.A.* **1999**, *96*, 9068−9073.
(40) Satoh, D.; Shimizu, K.; Nakamura, S.; Terada, T. Folding Free-Energy Landscape of a 10-Residue Mini-Protein, Chignolin. *FEBS Lett.* **2006**, *580*, 3422−3426.