JCTC Journal of Chemical Theory and Computation

ARTICLE

pubs.acs.org/JCTC

Flow-Dependent Unfolding and Refolding of an RNA by Nonequilibrium Umbrella Sampling

Alex Dickson, Mark Maienschein-Cline, and Allison Tovo-Dwyer

James Franck Institute, The University of Chicago, Chicago, Illinois 60637, United States

Jeff R. Hammond

Leadership Computing Facility, Argonne National Laboratory, Argonne, Illinois 60439, United States

Aaron R. Dinner*

James Franck Institute, The University of Chicago, Chicago, Illinois 60637, United States

ABSTRACT: Nonequilibrium experiments of single biomolecules such as force-induced unfolding reveal details about a few degrees of freedom of a complex system. Molecular dynamics simulations can provide complementary information, but exploration of the space of possible configurations is often hindered by large barriers in phase space that separate metastable regions. To solve this problem, enhanced sampling methods have been developed that divide a phase space into regions and integrate trajectory segments in each region. These methods boost the probability of passage over barriers and facilitate parallelization since integration of the trajectory segments does not require communication, aside from their initialization and termination. Here, we present a parallel version of an enhanced sampling method suitable for systems driven far from equilibrium: nonequilibrium umbrella sampling (NEUS). We apply this method to a coarse-grained model of a 262-nucleotide RNA molecule that unfolds and refolds in an explicit flow field modeled with stochastic rotation dynamics. Using NEUS, we are able to observe extremely rare unfolding events that have mean first passage times as long as 45 s (1.1×10^{15} dynamics steps). We examine the unfolding process for a range of flow rates of the medium, and we describe two competing pathways in which different intramolecular contacts are broken.

I. INTRODUCTION

Nonequilibrium measurements on biological macromolecules, such as mechanical force-induced unfolding[1] and flow-based analogs,[2] have emerged as a powerful complement to equilibrium studies. Indeed, it is now possible to follow the evolution of distances through fluorescence resonance energy transfer (FRET) simultaneously with forces through optical traps.[3] While these measurements provide unprecedented experimental data on the stochastic dynamics of individual molecules, they still only probe at most a few degrees of freedom among many. Molecular dynamics simulations, which provide complete information about the positions of all participating particles subject to the assumptions of the model, have proven to be a valuable tool for interpreting these data.[4] However, the time scales for conformational change are often long compared with elementary fluctuations, which makes waiting for the events of interest to occur spontaneously under conditions representative of experimental ones prohibitively computationally costly. To accelerate convergence, many simulation studies employ unrealistically extreme nonequilibrium conditions (see discussion in Hu et al.[5]).

Alternatively, enhanced sampling methods can be used to improve exploration of phase space and focus computational effort on low probability regions of mechanistic importance, such as transition states. The most widespread such methods[6−8] rely on the fact that the statistics of equilibrium systems are known *a priori*, which prevents the applicability of such methods to nonequilibrium situations. However, there now exist methods that can enhance the sampling of low probability regions without relying on equilibrium properties of the system.[9−20] Although these methods differ in detail, the essential idea in all of them is to harvest segments of unbiased dynamics trajectories such as to achieve relatively uniform sampling of different regions of a space of physically relevant degrees of freedom (order parameters). The acceleration of convergence follows from the fact that each trajectory segment need only traverse a small portion of the space of order parameters, across which the probability is relatively uniform.

We have been developing one such method: nonequilibrium umbrella sampling (NEUS).[13−16] In this paper, we present a streamlined version of the algorithm with improved convergence properties. The most significant change is the explicit association of a weight with each saved copy of the system, motivated by the weighted ensemble method.[17−20] The fact that many trajectory segments are integrated independently makes the method highly parallelizable, and we detail and implement a strategy that can provide excellent scaling to large numbers of processors.

We use the method to simulate partial unfolding and refolding of a coarse-grained model of a 262-nucleotide RNA molecule in

Received: June 2, 2011
Published: July 29, 2011

the presence of a flow field. Our interest in this system comes from single-molecule studies of FRET between probes on the L18 loop and 3′ terminus of the catalytic domain of the RNase P RNA from *Bacillus stearothermophilus*.[22−24] In these studies, the molecule was tethered in a microfluidic channel to enable relatively rapid changes in magnesium ion concentration, and this led to the question of whether flow contributed to the dynamics observed.[24] Specifically, we wondered whether there were dynamics like the quasi-periodic folding and unfolding observed in previous simulations of a homopolymer in a laminar flow field.[25] Here, we show that in the RNA under flow system there are two competing unfolding pathways, the likelihoods of which depend on the rate of flow of the solution. We compare these results with reversible unfolding simulations (without a net flow).

II. METHODS

II.A. Algorithm. As we show, the slowest degrees of freedom in the system examined here have relaxation times on the order of milliseconds to seconds, while straightforward simulations of the coarse-grained model are limited to tens of microseconds. Thus, enhanced sampling is needed. Here, we describe the version of nonequilibrium umbrella sampling (NEUS)[13−16] used in the present study. To this end, we summarize the overall strategy, and then we describe the phases of the simulation and parallelization; differences from earlier versions of the algorithm and competing methods are noted.

II.A.1. Overall Strategy. The sampling is guided by a set of physically relevant variables ("order parameters"). Ideally, these order parameters describe the slow dynamics in the system, and the remaining degrees of freedom relax relatively fast. In this work, we employ a single order-parameter that quantifies the total number of intramolecular contacts (section II.B). However, we explicitly separate the "forward" (unfolding) and "backward" (refolding) transition path ensembles as in Dickson et al.[15] This allows the sampling of the orthogonal degrees of freedom to differ between the two ensembles (i.e., allows for nonoverlapping unfolding and refolding transition path ensembles), and it enables the calculation of transition rates between basins.

For the simulations, we divide the space of order parameters into regions, which need not be uniform in size. Each region contains one or more copies of the system (walkers) that evolve independently according to the natural dynamics of the system, and we associate with that walker a weight for contributing to averages. When a walker of the system attempts to leave its region, the configuration is saved to a list of entry points for the neighboring region, along with the weight of the walker. When a neighboring list is full, the oldest saved configuration is overwritten, and its weight is distributed over the remaining points in the list in a manner that does not affect their relative probabilities of being chosen. The walker is then restarted from a saved configuration, i, which is chosen from one of its region's lists with a likelihood proportional to its weight (w_i). The weight of this point is then partitioned between the walker and the saved entry point: γw_i ($\gamma \in (0,1]$) is given to the walker, and the rest, $(1-\gamma)w_i$, remains associated with the saved entry point. Note that $\gamma = 1$ results in straightforward dynamics, or a single, continuous trajectory. Here, we use $\gamma = 0.9$. The incorporation of this feature in the NEUS algorithm is motivated by the (equal) partitioning of the probability when a trajectory branches in the weighted ensemble (WE) method;[17] it ensures conservation of the starting probability and suppresses artificial amplification of the probability of particular trajectories. As a result, we are able to obtain converged results with only one set (lattice) of regions in the extended space as opposed to two, as in previous work.[13−15]

II.A.2. Initialization. A common situation is that one is interested in studying a transition between two or more states, but one knows the configuration of the system in only one of the stable states. This situation applies here to the RNA-under-flow system, since we know the folded configuration but not the most likely unfolded configurations. Although in principle one could start the simulation in each region using any configuration consistent with the allowed order parameter values, in practice, it is best to start with a distribution of structures that is as consistent as possible with the physically weighted dynamics to avoid introducing unnecessary errors that take time to be corrected. To this end, we progressively activate the regions in a manner similar to forward flux sampling (FFS)[12] as follows.

We start by running an unconstrained simulation that is initialized in the known stable configuration. During this simulation, we record the configuration each time the system crosses a boundary of a region but do not reset the configuration. These configurations serve as the initial entry (i.e., resetting) points for the regions visited, and all such configurations are assigned equal weight. Following the unconstrained simulation, we begin the umbrella sampling simulation starting from saved entry points in each region that has at least one such point, employing and updating the copy weights as described above. Regions that were not visited previously are activated once entry points for them are obtained. As the simulations proceed, regions of lower and lower probability are activated by their neighbors, and trajectories emerge from the original stable state. Once all of the regions are activated, we are able to concurrently sample the entire order parameter space of interest, using only points that resulted directly from the starting distribution.

In the present study, the progressive initialization of regions accounts for about 2% of the total simulation time. The sampling procedure employed here further differs from FFS in that it does not explicitly require a notion of forward progress and thus can be used with sampling regions that are defined by an arbitrary number of order parameters. By the same token, trajectories are terminated when they cross any boundary, not only a forward one. This distinction is of practical importance when the dynamics do not lead rapidly back to the starting basin (see Dickson et al.[16] for further discussion).

II.A.3. Weight Redistribution. The algorithm as described is in principle complete. Indeed, it is very similar to the WE method except that (i) it permits strict control of the number of copies in a region (including only limiting it to one) and (ii) differs in the details of weight partitioning when resetting (branching) and redistributing when overwriting (pruning). However, the transfer of weight between regions of high probability can be very slow when the weight must pass through a bottleneck region of low probability. This is because a very large number of low probability walkers are required to add up to a significant change of weight in a high probability region. This convergence issue arises despite the fact that the time for initial exploration of the space decreases with an increasing number of regions, as in any umbrella sampling procedure.[8,26]

To accelerate convergence after the initialization phase, we periodically use the interface-to-interface crossing statistics to predict statistical weights for each region ($\{W_i\}$) and scale the weights of the entry points in each region, i, such that their sum is

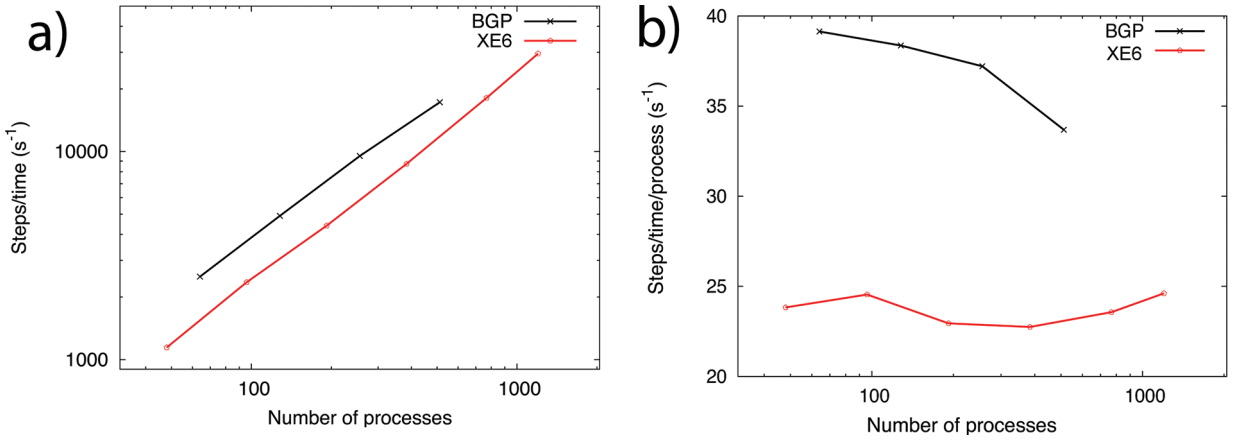

Figure 1. Scaling results on two parallel computing architectures: the Intrepid Blue Gene/P and the "Beagle" Cray XE6 supercomputers. Note that the code was not specifically optimized for performance on either machine. (a) Steps per unit time as a function of the number of processes on both machines. Perfect scaling on this plot is shown by a straight line with slope equal to 1. (b) Steps per unit time per process. Perfect scaling on this plot would be shown by a flat line. Note that here a process on the Blue Gene/P is composed of four cores, and a process on the Cray XE6 is composed of a single core.

equal to W_i. Here, the weights are obtained from a modified version of the scheme in Vanden-Eijnden and Venturoli,[27] where the total flux into a region is set equal to the total flux out of a region. To this end, we accumulate a transition matrix, **T**: each off-diagonal element t_{ij} is the number of transitions observed from region j to region i in the last weight update period, and each diagonal element $t_{ii} = -\sum_{j \neq i} t_{ji}$. We then solve the equation $\mathbf{TW} = 0$ for the weight vector **W** by using singular value decomposition to compute the nullspace of **T**, which is the single nontrivial solution **W**. Here, we perform this operation periodically throughout the simulation, as in previous NEUS studies;[15,27] this contrasts with the study by Bhatt et al.[19] in which a single such step is used to precondition the simulation and then flux balance is used to check convergence.

II.A.4. Parallelization. The simulations of the copies of the system require only limited communication. As such, NEUS and methods like it lend themselves well to parallelization. However, we find that they benefit from careful implementation on high performance computers. All simulations for the present study are run on parallel architectures using the Global Arrays toolkit,[28] which implements a global address space programming model in which processes can access remote data using one-sided communication. One-sided communication is particularly useful in this case, since the timing of boundary crossing events is not predictable. The global address space also enables one to distribute the storage of a large set of region entry points across the memory of many compute nodes. The entry points for each region, the region weights, boundary crossing statistics, and sampling histogram data are all stored as global arrays. These arrays can be modified by any process using "put" functions and "get" functions, where locks are used to enable atomic updates of global data (modifications of the entry point lists, for instance) that prevent processes from concurrently accessing the same region of a global array.

Although the dynamics of the copies are simulated essentially without communication once they are initialized, NEUS still periodically requires some collective operations, such as weight updates, and the computation of rates and probability distributions. To allow for such operations, we break down the simulation into "cycles" of computation, at the end of which all processes are synchronized. Within the cycles, the work is distributed among the processes as follows. When a process is finished running a trajectory segment, it queries how many steps have been run in each region k so far this cycle (N_k), and it uses the results to decide in which region to run the next trajectory segment. Specifically, it chooses to start a trajectory in region j with probability

$$P_j = \frac{N_{\text{steps}} - N_j}{\sum_k (N_{\text{steps}} - N_k)} \tag{1}$$

where N_{steps} is the number of steps to be run in each region per cycle. A trajectory is terminated if the counter in its region reaches N_{steps} (upon which the current configuration of the system is saved to the entry point list as a simple means of maintaining it), and a computational cycle ends when all counters reach N_{steps}. Here, N_{steps} is set to 2000.

Figure 1 shows preliminary scaling results obtained on two parallel architectures: Intrepid, a Blue Gene/P supercomputer, and Beagle, a Cray XE6 supercomputer. Each scaling test consisted of running 10 cycles (as defined above), and the wall-clock time elapsed between the beginning of the first cycle and the end of the last cycle was used to compute the number of dynamics steps per unit time. In each test, we use the RNA under flow system presented below, but we check for boundary crossings every 50 dynamics steps, as opposed to every five in the rest of the work presented here. Longer periods between boundary crossing checks results in better scaling since there is more time between communication events. On the Blue Gene/P, scaling tests are run on groups of 64, 128, 256, and 512 processes. In this case, each process is composed of four cores but acted as a single process in the NEUS algorithm, where three of the cores are used as OpenMP threads and one is used exclusively for communication. Figure 1a shows reasonable scaling up to 512 processes, but Figure 1b clearly reveals a loss in efficiency as the number of processes is increased. On Beagle, scaling tests were run on groups of 48, 96, 192, 384, 768, and 1200 processes. In this case, each process is composed of a single core; no OpenMP theading was used. Excellent scaling is observed for these numbers of processes, as seen in Figure 1, where the number of dynamics steps per unit time per process is roughly flat. The slight variation in Figure 1b reflects the specific compute nodes that are selected to run the job: nodes that are closer together in the machine

result in faster communication, and better scaling overall, which is expected.

II.B. Model. The system is a model of the catalytic domain of RNase P RNA from *Bacillus stearothermophilus*. To make the simulations tractable, we use a coarse-grained representation that averages over the atomic structure and dynamics, while taking into account the secondary and tertiary interactions that stabilize the native state: the self-organized polymer (SOP) model.[29] In the SOP model, each nucleotide of the RNA is treated as a bead, and the beads interact through potentials that depend on the known native structure. The potential defining the model is the sum of a finitely extensible nonlinear elastic (FENE) potential that connects adjacent beads[30] (V_{FENE}); a Lennard-Jones attraction between beads that has a minimum at the native structure distance ($V_{\mathrm{nb}}^{\mathrm{att}}$); pairwise nonbonded repulsions scaling as r^{-6}, which locally straighten the chain and mimic steric repulsions between nucleotides ($V_{\mathrm{nb}}^{\mathrm{rep}}$); and a Weeks–Chandler–Andersen[31] (WCA) repulsion between each bead and the wall at $y = 0$ (V^{wall}). The total potential function is

$$V_{\mathrm{T}} = V_{\mathrm{FENE}} + V_{\mathrm{nb}}^{\mathrm{att}} + V_{\mathrm{nb}}^{\mathrm{rep}} + V_{\mathrm{wall}}$$

with

$$V_{\mathrm{FENE}} = -\sum_{i=1}^{N-1} \frac{k}{2} R_0^2 \log\left(1 - \frac{(r_{i,i+1} - r_{i,i+1}^0)^2}{R_0^2}\right)$$

$$V_{\mathrm{nb}}^{\mathrm{att}} = \sum_{i=1}^{N-3} \sum_{j=i+3}^{N} \varepsilon_h \left[\left(\frac{r_{ij}^0}{r_{ij}}\right)^{12} - 2\left(\frac{r_{ij}^0}{r_{ij}}\right)^{6}\right] \Delta_{ij}$$

$$V_{\mathrm{nb}}^{\mathrm{rep}} = \sum_{i=1}^{N-2} \varepsilon_l \left(\frac{\sigma^*}{r_{i,i+2}}\right)^{6} + \sum_{i=1}^{N-3} \sum_{j=i+3}^{N} \varepsilon_l \left(\frac{\sigma}{r_{ij}}\right)^{6} (1 - \Delta_{ij})$$

$$V_{\mathrm{wall}} = \sum_{i=1}^{N} H(2^{1/6}\sigma_{\mathrm{WCA}} - y_i) \times 4\varepsilon_l \left[\left(\frac{\sigma_{\mathrm{WCA}}}{y_i}\right)^{12} - \left(\frac{\sigma_{\mathrm{WCA}}}{y_i}\right)^{6}\right] \quad (2)$$

where r_{ij} is the distance between residues i and j, and r_{ij}^0 is their distance in the native structure. We set the parameters in eq 2 to those in Hyeon and Thirumalai,[29] namely, $k = 20$ kcal/(mol × Å^2), $R_0 = 2$ Å, $\varepsilon_h = 0.7$ kcal/mol, and $\varepsilon_l = 1.0$ kcal/mol. We set $\sigma = 7$ Å to ensure noncrossing of the chain, and we set $\sigma^* = 3.5$ Å to prevent the flattening of helical structures. In V_{wall}, $\sigma_{\mathrm{WCA}} = 2$ Å, and $H(x)$ is a Heaviside function equal to 0 for $x < 0$ and 1 for $x > 0$. The size of the box was chosen such that the residues would interact only with the $y = 0$ surface. Consequently, repulsive potentials were not needed for the other walls of the box, and no collisions of the residues with those walls were observed in our simulations. The equation of motion for the polymer is integrated with the Velocity–Verlet algorithm with time step $\delta t = 40$ fs.

The native, folded structure was constructed from the crystal structure for the full RNase P RNA.[32] The coordinates of the catalytic domain (262 residues) were isolated from the full structure (417 residues), and coarse-graining into beads was carried out by replacing the coordinates of each residue with its center of mass. Unstructured residues, which did not have crystal structure coordinates (in Figure 2, residues 161–181 in P1, 15–20 in P15, 64–73 in P18, and 106–125 in P19), were added by introducing the appropriate number of beads into the sequence, separated by the average bead–bead distance (about 5 Å); these unstructured residues have no contacts. The structure was allowed to relax to its minimum energy by integrating without solvent so that the added unstructured residues form

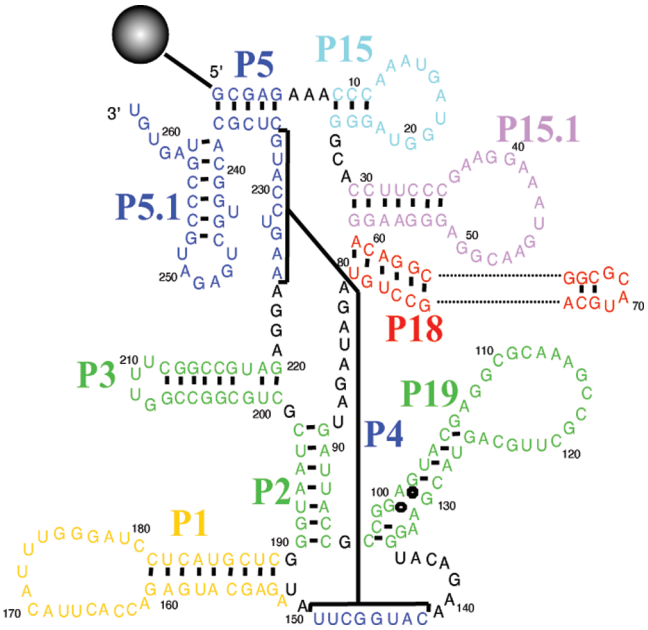

Figure 2. Secondary structure of the RNA molecule. In the simulation, the 5′ end of the molecule is attached to a tether (black sphere) that prevents the molecule from moving along with the flow. The index of every tenth residue is shown.

simple loops. Using this structure, we consider a native contact to exist ($\Delta_{ij} = 1$) between all pairs of residues i and j with $|i - j| > 2$ and a distance less than $R_{\mathrm{C}} = 14$ Å in the native structure; for all other pairs, $\Delta_{ij} = 0$.

The solvent in the simulation is modeled using the stochastic rotation dynamics method,[33−36] in which the solvent is represented by a large number (here, 503 200) of infinitesimal particles that are grouped into cubic "interaction cells". Each step of the algorithm comprises two parts: (1) free streaming, in which the position of particle i ($\mathbf{r}_i$) is updated according to $\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t$, where $\mathbf{v}_i$ is the velocity at time t and $\Delta t = 150\delta t$ is the solvent time step, and (2) "collision", in which $\mathbf{v}_i(t + \Delta t) = \mathbf{v}_{\mathrm{cell}}(t) + \Omega[\mathbf{v}_i(t) - \mathbf{v}_{\mathrm{cell}}(t)]$, where $\mathbf{v}_{\mathrm{cell}}$ is the average velocity of particles in the cell containing i and Ω is a stochastic rotation matrix which rotates vectors around a random axis by $\pm\alpha$, a fixed angle, with equal likelihood. Here, we use $\alpha = 0.243\pi$, which in combination with the other parameters used here for the solvent gives a viscosity of 0.8 g/m/s, which is approximately the viscosity of liquid water at our simulation temperature (300 K). The viscosity was calculated using eqs 10 and 14 of Kikuchi et al.[37]

We allow the solvent to influence the RNA by including the polymer beads in the collisions, as in Webster and Yeomans.[38] This is done using

$$\mathbf{v}_{\mathrm{cell}}(t) = \frac{\sum_{\mathrm{solv} \in \mathrm{cell}} m\mathbf{v}_i(t) + \sum_{\mathrm{res} \in \mathrm{cell}} M\mathbf{V}_i(t)}{N_{\mathrm{cell}}^{\mathrm{solv}}(t)m + N_{\mathrm{cell}}^{\mathrm{poly}}(t)M} \quad (3)$$

where $m = 32$ amu is the mass of the solvent particles (chosen to make a solvent mass density of 1 g/mL) and $M = 300$ amu is the mass of the residues, compared with a range in mass for RNA nucleotides of 320–360 amu. $\mathbf{V}_i(t)$ is the velocity vector for residue i, and the sums on the left and right are over all the solvent particles in the cell and all the residues in the cell, respectively.

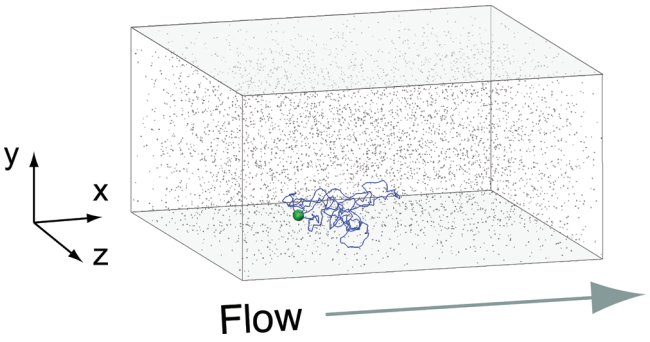

Figure 3. The simulation cell. The boundaries at $y = 0$ and $y = L_y$ have reflective boundary conditions, while the others are periodic. The tether point is shown as a green sphere, and the RNA molecule is in blue. A total of 5000 of the 503 200 solvent molecules are shown here. A flow is induced in the positive x direction by applying a constant acceleration to the solvent particles, which in turn causes extension of the RNA molecule in that direction.

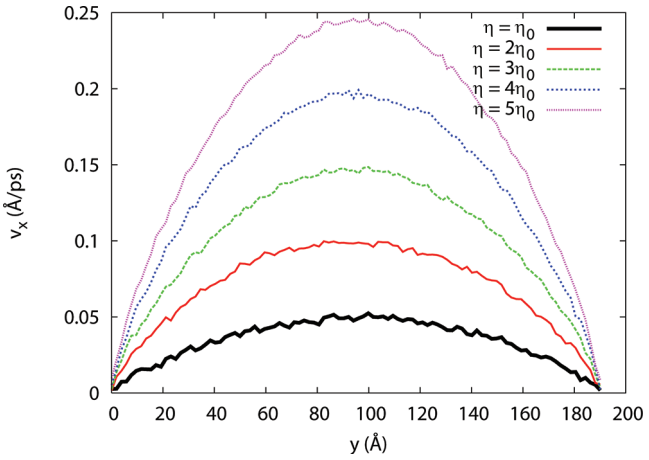

Figure 4. Flow velocity profiles. For each flow rate examined here, we plot the average velocity of solvent molecules in the x direction as a function of y. These were obtained without the polymer. The profiles are parabolic, due to the presence of reflective walls at $y = 0$ and $y = L_y$.

$N_{\text{cell}}^{\text{solv}}(t)$ and $N_{\text{cell}}^{\text{poly}}(t)$ are the number of solvent particles and residues in the cell, respectively, at time t. We then include the polymer beads in the collision step using $\mathbf{V}_i(t + \Delta t) = \mathbf{v}_{\text{cell}}(t) + \Omega[\mathbf{V}_i(t) - \mathbf{v}_{\text{cell}}(t)]$ where Ω is the same rotation matrix used for the solvent particles in the interaction cell.

We use periodic boundary conditions in the x and z directions and walls that reflect all components of the velocity of the solvent particles upon collision at $y = 0$ and $y = L_y$. We then drive the solvent to flow in the positive x direction (Figure 3). The dimensions of the box are $L_x = L_z = 384$ Å and $L_y = 192$ Å. The interaction cells are cubic with a side length of 8 Å, which was chosen to be comparable with the average distance traveled by a solvent particle in a time Δt. Following previous work, we shift the lattice periodically to avoid artifacts[34] and employ the generalized bounce back rule for partially filled cells along the $y = 0$ and $y = L_y$ edges.[35] An extra FENE interaction is added between the 5′ terminus and the tether point, located at (120, 25, 192 Å) to prevent the molecule from moving along with the flow.

The flow is introduced by accelerating each solvent particle that is not in the $y = 0$ or $y = L_y$ interaction cells in the x direction after every rotation step according to $v_x^i \rightarrow v_x^i + \eta\Delta t$, where η is an acceleration parameter. The η values used here range from η_0 to $5\eta_0$, where $\eta_0 = 625$ Å/fs^2. Figure 4 shows average flow profiles, obtained without the polymer. The Péclet number is the ratio of advective motion to thermal diffusive motion, given by

$$Pe = \frac{L\bar{v}_x}{D} \tag{4}$$

where $L = 8$ Å is the characteristic length, $\bar{v}_x$ is the average velocity of the solvent in the x direction, and D is the self-diffusion constant of a single residue calculated in zero flow. Here, Pe ranges from 7.5×10^{-3} to 3.9×10^{-2}, indicating that at all values of η we examined, thermal motion was much stronger than advective motion (i.e., $Pe \ll 1$). Prior to the start of the umbrella sampling simulation, the solvent was equilibrated without the polymer until the flow profiles converged; this required 40 ns, which corresponds to roughly 6700 streaming steps.

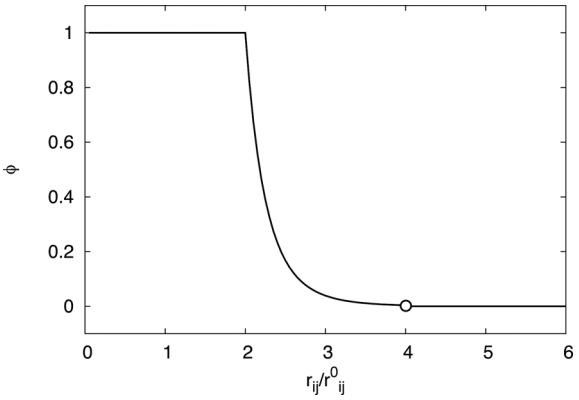

Figure 5. The function $\phi(r_{ij})$ that is used to calculate the order parameter N_c.

II.C. Order Parameter. The order parameter that we use here to distinguish between the folded and unfolded states is an estimate of the number of native contacts that are made in a given configuration:

$$N_c(t) = \sum_{i=1}^{N}\sum_{j\neq i}^{N} \Delta_{ij}\,\phi(r_{ij}(t)) \tag{5}$$

where $r_{ij}(t)$ is the distance between the two residues at time t, $\phi(r_{ij})$ is a function that is equal to 1 when the contact is satisfied ($r_{ij} < a_f r_{ij}^0$), is 0 when the contact is not satisfied ($r_{ij} > 2a_f r_{ij}^0$), and varies between 0 and 1 for intermediate values according to $(a_f r_{ij}^0/r_{ij})^8$, where the exponent was chosen to make the jump at $r_{ij} = 2a_f r_{ij}^0$ small, while being efficient to compute. The constant $a_f = 2.0$ was used here; we found that it provided a good balance between limiting sensitivity to fluctuations within stable states (large a_f) and detecting early unfolding activity (small a_f). A plot of $\phi(r_{ij})$ is shown in Figure 5.

We use this order parameter to define "folded" and "unfolded" basins as $N_c \geq N_{\text{fold}}$ and $N_c \leq N_{\text{unfold}}$, respectively. The choice of N_{fold} and N_{unfold} is discussed in section III. We separate the transition path ensemble into two subensembles: the unfolding ensemble and the refolding ensemble. The unfolding ensemble is composed of all trajectories that originate in the folded basin (regardless of whether they reach the unfolded basin or return to the folded one), and the refolding ensemble is composed of all trajectories that originate in the unfolded basin (regardless of whether they reach the folded basin or return to the unfolded one).

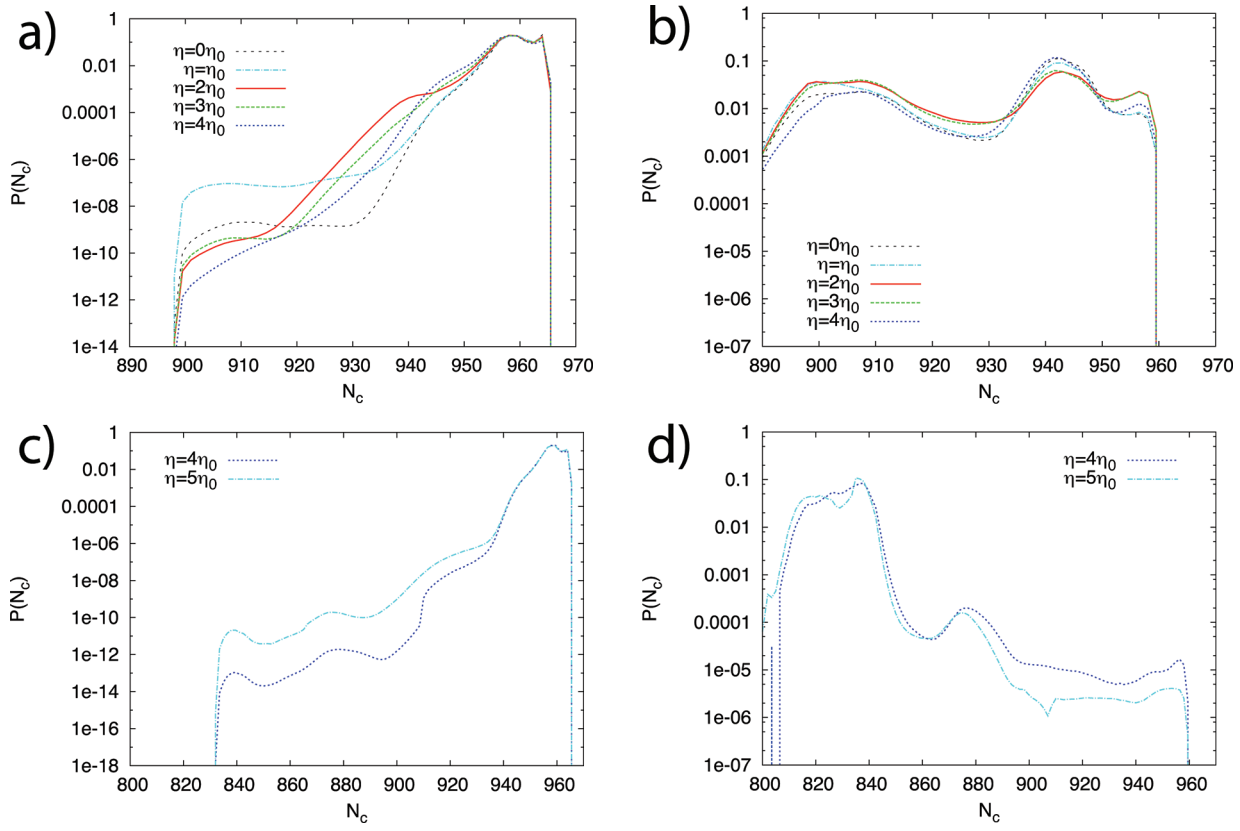

Figure 6. N_c histograms for both pathways. (a,b) The unfolding and refolding ensembles of path M, respectively. The histograms for all flow pressures are shown and share a similar shape. (c,d) The unfolding and refolding ensembles of path E, respectively. Flow pressures $\eta/\eta_0 = 4$ and 5 are shown. In all panels, the histograms are normalized such that the sum of the 200 points in each curve is equal to 1.

In other words, the ensembles are defined by the histories rather than futures of walkers. Each ensemble has its own set of regions that span the order parameter space. As shown in Dickson et al.,[15] the two sets of regions can be seen as a single set of nonoverlapping regions in an extended space, and transition rates between the basins can be obtained by calculating fluxes in this extended space.

II.D. Simulation Details. In the simulations presented here, the saved entry point lists for each region are divided into two lists of 250 points each. One list is dedicated to points coming from the right (higher N_c) and the other to points coming from the left (lower N_c). This helps ensure that the left and right ensembles are both well described. An element of a list consists of the positions and velocities of all the residues of the molecule, as well as forces from the previous step of the Velocity–Verlet algorithm. Along with these data, we store the weight of the trajectory and a time counter that is used to determine when to perform solvent streaming steps. We found it unnecessary to store the coordinates of the solvent along with the flux input point, since the solvent relaxes almost instantaneously to the presence of the polymer (data not shown), as there are no steric interactions between the polymer and solvent.

In the work below, 2000 RNA time steps in each active sampling region constitute a cycle. We allow 3000 cycles for progressive initialization (phase II) and another 3000 cycles with global weight updates (phase III). We perform a global weight update at the beginning of phase III and again every 600 cycles after that. As will be discussed below, the number of sampling regions used depends on the pathway observed and is either 40 or 84 in each direction, for a total of either 80 or 168 regions in the extended space. The total number of sampling steps depends on how fast regions are initialized in phase II, but it is less than 9.6×10^8 in the 40 region case and less than 2.02×10^9 in the 84 region case.

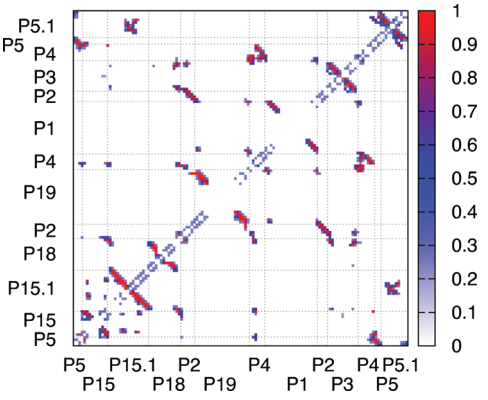

Figure 7. The contact map that is representative of the folded state for all flow rates examined here. This map was obtained using structures from entry points for the region in the unfolding ensemble with the highest value of N_c, at the end of a $\eta = 2\eta_0$ simulation.

III. RESULTS

The RNA-under-flow system was examined at five different flow accelerations: $\eta = \eta_0$, $2\eta_0$, $3\eta_0$, $4\eta_0$, and $5\eta_0$. These correspond to Péclet numbers of 7.5×10^{-3}, 1.5×10^{-2},

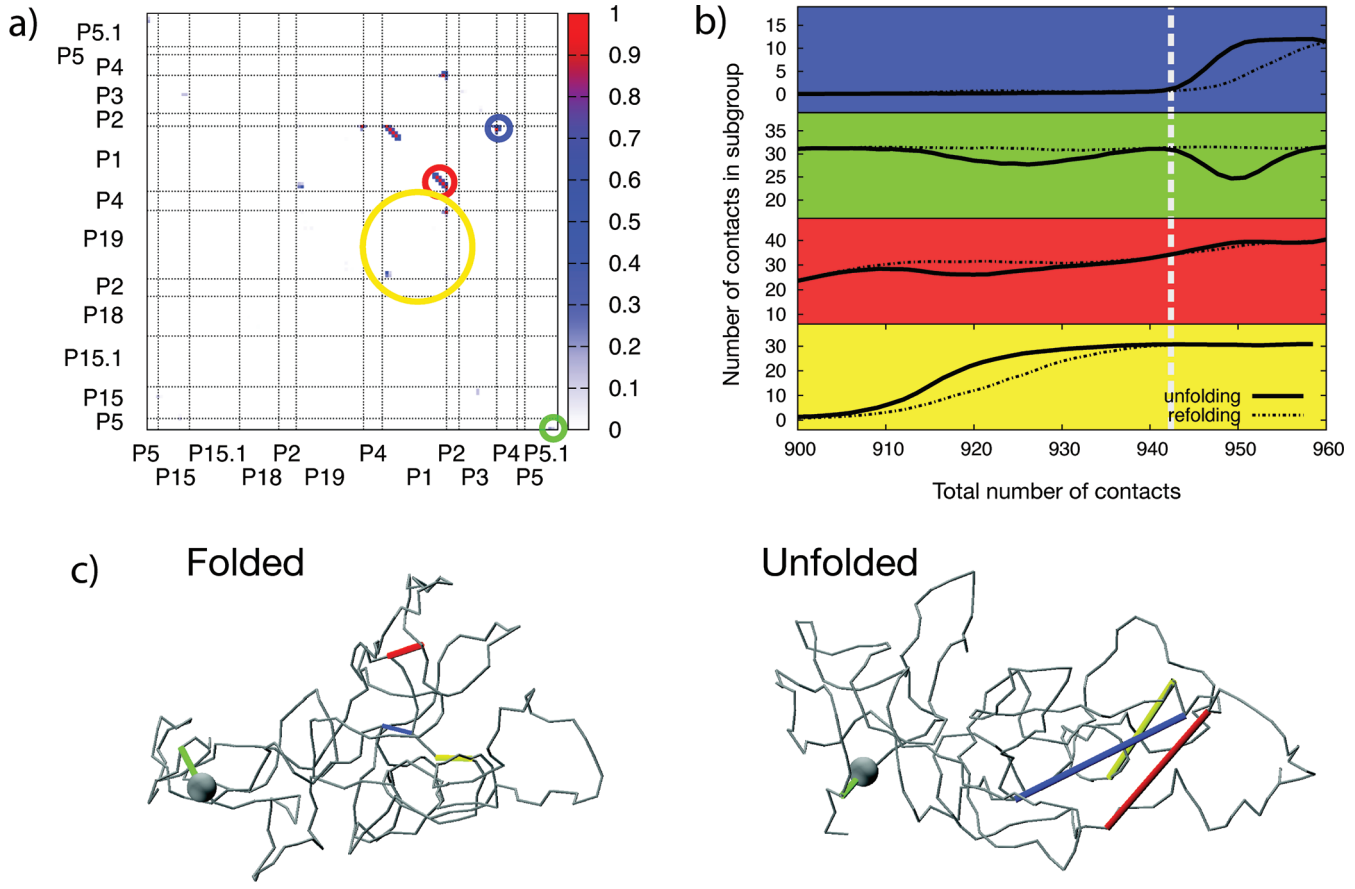

Figure 8. Analysis of path M. (a) Contact difference map obtained by subtracting the contact map of the unfolded state from the contact map of the folded state. This reveals the contacts which are broken along the pathway. The colored circles show the division of these contacts into subgroups. Note that although the contact map is symmetric, the colored circles are only shown in the lower right triangle for clarity. (b) The number of contacts in each subgroup is plotted as a function of the total number of contacts averaged over the $\eta = 2\eta_0$ ensemble of structures. These curves are computed using structures in the saved entry point lists for every region in both the unfolding and refolding ensembles, at many times throughout phase III of the simulation. The vertical line shows the metastable states along the refolding pathway. In the text, the subgroups are numbered 1 to 4, starting at the top. (c) Representative contacts from each group are shown on the RNA molecule for the folded and unfolded states.

2.3×10^{-2}, 3.0×10^{-2}, and 3.9×10^{-2}, respectively. These numbers indicate that thermal motion is much more important than advective motion (i.e., $Pe \ll 1$), but as we show, there are significant flow effects. We also examine the equilibrium case: $\eta = 0$. For each flow rate, we obtained folding and unfolding rates, probability distributions for the numbers of native contacts, and sets of input structures to each umbrella sampling region, from which we can reconstruct folding and unfolding pathways. As detailed below, the folded and unfolded basins were defined by our measure of the number of native contacts, N_c (section II.C).

III.A. Competing Unfolding Pathways. Interestingly, we found two competing reaction pathways for the molecule. One pathway ("path M") occurred by breaking contacts in the middle of the molecule, in and around the P1 loop (residues 150−190, see Figure 2), while the other ("path E") occurred by breaking contacts in and around the P5 region (residues 1−5 and 234−238), which is near the tethered end. We obtained pathways in duplicate for each value of η and found a dependence of the pathway on the flow pressure. For $\eta \leq 3\eta_0$, we observed path M in both trials. For $\eta = 5\eta_0$, we observed path E in both trials. And, for $\eta = 4\eta_0$, we observed path M and path E each once, which suggests that path E is more probable for higher flow rates, and that $\eta = 4\eta_0$ is close to a transition point where the relative probabilities of the two pathways cross over.

The folded basin for both pathways was located at $N_c \geq 960$, and the unfolded basin was placed at the first metastable unfolded structure we encountered along each unfolding pathway. Although these structures could be intermediates to further unfolded states, we will call these structures "unfolded" and their corresponding basins "unfolded basins". For path M, we set the unfolded basin to $N_c \leq 900$, and for path E we set the unfolded basin to $N_c \leq 834$. In both pathways, we define the regions in N_c with an even spacing of $\Delta N_c = 1.5$, giving us 40 regions for the unfolding pathway in path M and 84 regions for the unfolding pathway in path E. There are an equal number of regions in the refolding pathways in both cases, giving us a total of 80 and 168 regions in paths M and E, respectively.

III.B. Pathway Analysis. Probability distribution functions of the order parameter N_c are shown in Figure 6, for both pathways, and for both the unfolding and refolding ensembles. Histograms were saved every 50 cycles, and each curve shown in Figure 6 is an average of the last 20 histograms. For path M, we show histograms for $\eta \leq 4\eta_0$. In the unfolding ensemble (Figure 6a), there is a strong peak at $N_c = 960$ for all flow rates, corresponding to the native state. In the refolding ensemble (Figure 6b), there is

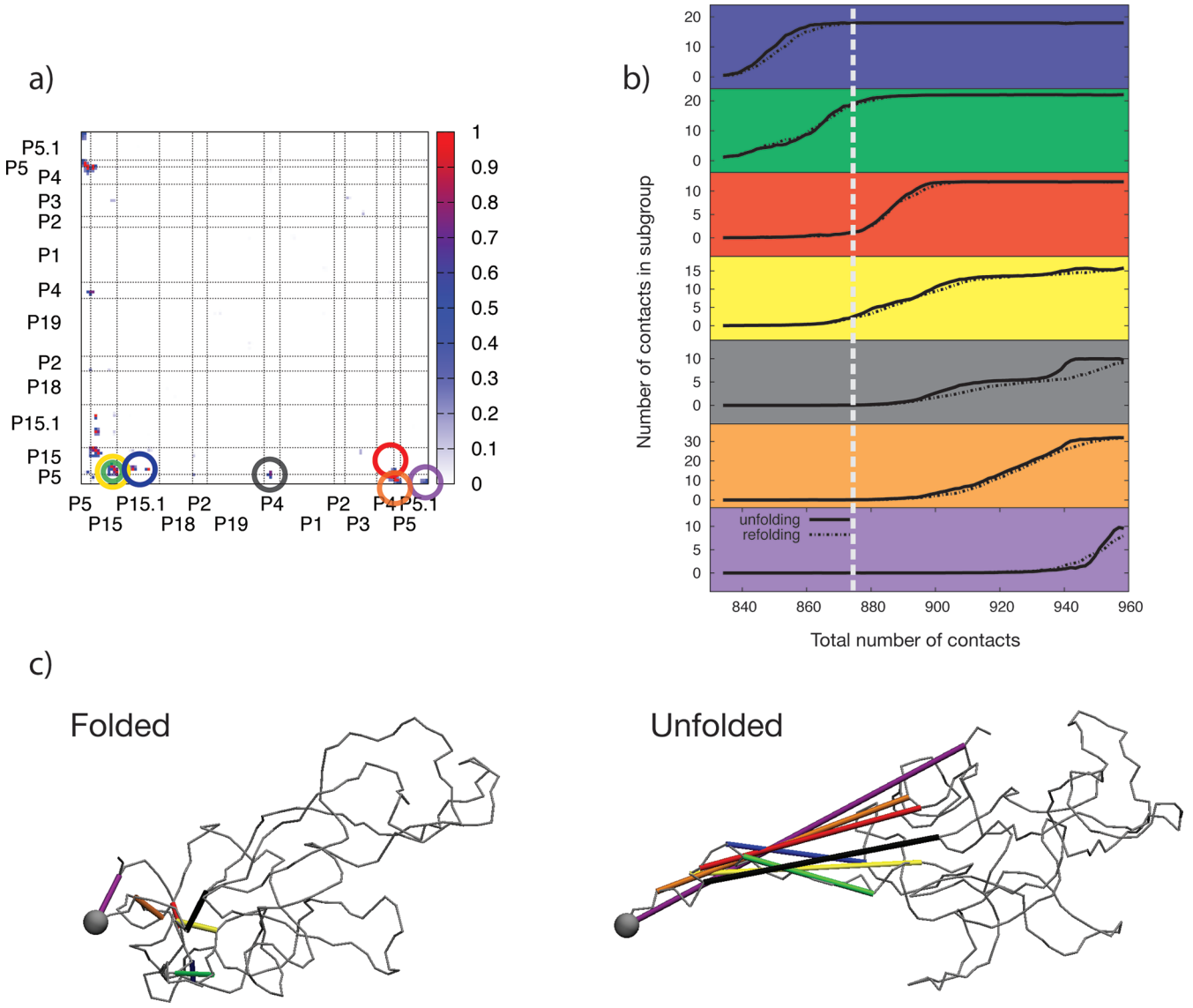

Figure 9. Analysis of path E. See descriptions of panels in Figure 8. (a) Contact difference map. The green and yellow circles define subgroups of secondary and tertiary contacts within the region, respectively. Although the contact map is symmetric, the colored circles are only shown in the lower right triangle for clarity. (b) Note in this pathway that there is a strong overlap between the unfolding and refolding ensembles.

a peak at $N_c = 907$ corresponding to the first metastable unfolded state and an intermediate unfolded state at $N_c = 942$. For path E, we show histograms for $\eta = 4\eta_0$ and $5\eta_0$. Here, the refolding ensemble (Figure 6d) shows that there are two metastable states near the unfolded basin with peaks at $N_c = 820$ and $N_c = 838$, as well as an intermediate at $N_c = 875$.

To characterize structures along the pathways, we construct contact difference maps by subtracting average contact maps for the unfolded states from that for the folded state shown in Figure 7. The contact maps for the unfolded states are computed using the structures from the saved entry point lists for the regions in the refolding ensembles with the lowest values of N_c, and similarly a contact map for the folded state is computed using structures from the saved entry point list for the region in the unfolding ensemble with the highest value of N_c. The contact difference maps are shown in Figures 8a and 9a along with characteristic structures of the folded and unfolded states (Figures 8c and 9c). On the basis of their kinetic behavior, we divide the contacts into groups and track the population of each group as a function of N_c (Figures 8b and 9b).

The vertical lines in Figures 8b and 9b show the metastable states along the refolding pathway. For path M, the local maximum at $N_c = 942$ is associated with the reformation of contacts in the P1 loop (subgroup 3, the third from the top in Figure 8b). For path E, the local maximum at $N_c = 875$ is associated with the reformation of contacts in the P15 loop (subgroups 2 and 4). For path M, we observe that the unfolding and refolding ensembles do not overlap. Specifically, contacts between the end points of the molecule (P5−P5.1 contacts) break and reform along the unfolding pathway but remain intact during the refolding pathway. In this regard, it is important to keep in mind that the unfolding ensemble, as defined in section II.C contains both folded-to-unfolded trajectories and folded-to-folded trajectories. The fact that the feature in question appears in analogous calculations for the reversible system ($\eta = 0$), where there can be no hysteresis, suggests that the P5−P5.1 contacts

dx.doi.org/10.1021/ct200371n |J. Chem. Theory Comput. 2011, 7, 2710–2720

Table 1. Unfolding and Refolding Mean First Passage Times for Path M, Obtained for $\eta = 0$, η_0, $2\eta_0$, $3\eta_0$, and $4\eta_0$[a]

η/η_0	unfolding (NEUS), in ms	refolding (NEUS), in ns	refolding (SF), in ns
0	29	1.4	1.7
1	0.60	1.4	0.8
2	140	0.40	1.4
3	170	0.48	0.6
4	3200	0.99	0.5

[a] For refolding pathways, the MFPTs from umbrella sampling (NEUS) and straightforward sampling (SF) are shown.

Table 2. Unfolding and Refolding Mean First Passage Times for Path E, Obtained for $\eta = 4\eta_0$ and $5\eta_0$[a]

η/η_0	unfolding (NEUS), in ms	refolding (NEUS), in μs	refolding (SF), in μs
4	45000	0.76	0.08
5	670	4.9	1.2

[a] For refolding pathways, the MFPTs from umbrella sampling (NEUS) and straightforward sampling (SF) are shown.

are broken along folded-to-folded trajectories and that this process is not a causal part of the path M unfolding mechanism.

There are variations in the contact subgroup projections between the $\eta = 2\eta_0$ ensemble (shown in Figure 8b) and the $\eta = 0$ and $\eta = \eta_0$ ensembles (not shown). Specifically, at N_c = 930, contacts in subgroup 4 are still mostly intact for $\eta = 2\eta_0$ but are mostly broken in the $\eta = 0$ and $\eta = \eta_0$ ensembles. This difference in the pathway coincides with the difference in shape of the unfolding N_c histograms shown in Figure 6a. For $\eta = 0$ and $\eta = \eta_0$, the major drop in probability has already occurred at N_c = 930 (going from right to left), where for the others, the probability continues to drop significantly for $N_c < 930$.

III.C. Transition Rates. The mean first passage times of the unfolding processes are given in Tables 1 and 2 for paths M and E, respectively. These range from 0.60 to 3200 ms for path M and 0.6 to 45 s for path E. As each dynamics step is 40 fs, these correspond to numbers of dynamics steps between 1.6×10^{10} and 1.1×10^{15}. The unfolding and refolding MFPTs are shown as functions of flow pressure in Figure 10.

Although we are only able to obtain unfolding rates for a small number of η values, the data suggests counterintuitive behavior for path M. For small flow rates, the MFPT decreases with increasing flow, which is intuitive, since one would expect the flow field to destabilize folded structures. However, for $\eta > \eta_0$, the MFPT increases with the flow rate; unfolding becomes more difficult as greater flow is applied to the system. Such behavior could be caused by larger flow gradients at the $y = 0$ boundary, causing nucleotides in the P1 loop to be pushed together rather than pulled apart. The difference in pathways between the $\eta \leq \eta_0$ flow rates and the $\eta \geq 2\eta_0$ flow rates described above could also explain the nonmonotonic rate behavior seen here, since the different pathways could involve different interactions with the flowing medium. Further sampling at intermediate flow rates, as well as isolated studies of the different intermediates, would be helpful to confirm this trend. For the two data points obtained for path E, the MFPT for unfolding decreases with increasing flow rate.

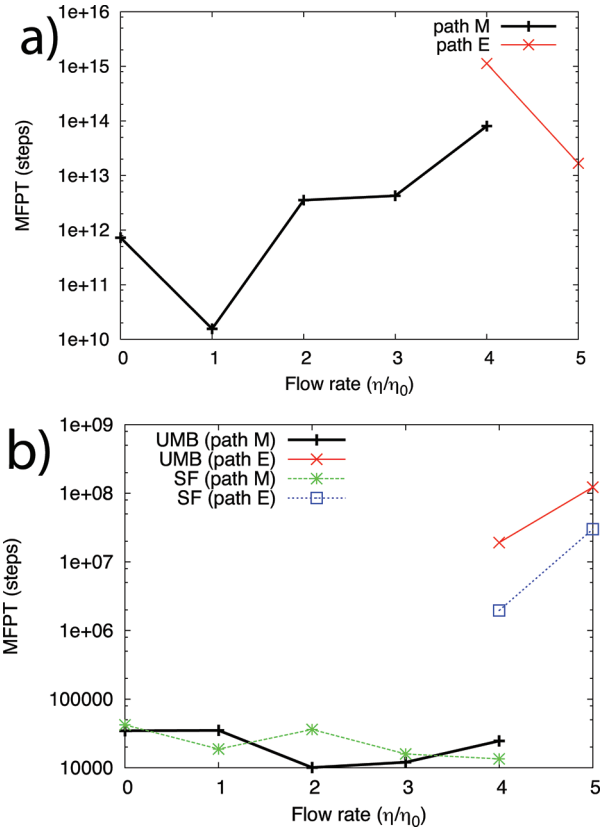

Figure 10. (a) Mean first passage times for unfolding events, as predicted by NEUS. For path M, this is the average number of steps required to go from N_c = 960 to N^c = 900, and for path E, this is the average number of steps to go from N_c = 960 to N_c = 834. (b) Mean first passage times for refolding events, comparing umbrella sampling (UMB) and straightforward trajectories (SF). These agree to within an order of magnitude.

We note that the MFPTs for path M and path E are not directly comparable, since the former measures the average amount of time to go from 960 to 900 contacts, and the latter measures the average amount of time to go from 960 to 834 contacts.

The rates of refolding are also given in Tables 1 and 2. They are much faster, which makes comparisons with straightforward trajectories possible. We use the umbrella sampling saved entry point lists to generate an initial unfolded ensemble for each flow rate, since umbrella sampling is our only access to physically weighted unfolded states. We compare refolding rates for both pathways and all flow pressures, which agree to within an order of magnitude. For path M, the refolding MFPT is short (~1 ns) and relatively constant with varying flow rate. For path E, the refolding MFPTs are longer, since the unfolded state is more stable, and increase with increasing flow rate: 0.76 μs for $\eta = 4\eta_0$ and 4.9 μs for $\eta = 5\eta_0$. The behavior suggests that, in this regime, higher flow fields stabilize the unfolded state.

To illustrate the importance of the enhanced sampling algorithm for the unfolding simulations, we computed 16 independent trajectories of 16 μs (4×10^8 dynamics steps) starting from structures taken from the folded basin. These trajectories were run using $\eta = 5\eta_0$, and "unfolding" was defined as reaching 900 contacts instead of the usual 834 for path E, in order to increase the probability of observing an unfolding event. Using NEUS, we found the MFPT for this process was 0.21 ms, making the length of

the straightforward trajectories 12.5% of the predicted MFPT, and no unfolding events were observed. These simulations required 30 days of computation on 16 2.5 GHz Intel Xeon processors. This also emphasizes the computational benefit of parallelization, as the $\sim 2 \times 10^9$ steps for the largest umbrella sampling simulations were completed in ~30 h of computation on 64 processors. However, even if a similar parallelization scheme using 64 processors was employed for straightforward trajectories, we predict that it would still take an average of ~1900 years to observe a single path E unfolding trajectory for $\eta = 4\eta_0$, and many times that to observe an ensemble of unfolding events.

IV. CONCLUSION

Here, we have presented a parallel version of NEUS and applied it to a coarse-grained macromolecular system driven far from equilibrium by flow. We obtained folding and unfolding rates and mechanisms for a range of flow speeds. This range was chosen to be physically reasonable yet result in significant flow effects. It is large compared to 1.6×10^{-5}, the Péclet number of the flow used to change the magnesium ion concentrations in the RNase P RNA single molecule experiments of Qu et al.,[23] and our simulations suggest that flow did not contribute to the dynamics discussed in refs 23 and 24, at least at moderately high magnesium ion concentrations, which strongly favor the folded state. A lack of knowledge of the structure of the RNA at low magnesium ion concentrations prevents us from assessing that situation.

Due to the stability of the native state, unfolding transitions were extremely slow, occurring as slowly as once in every 1.1×10^{15} dynamics steps, or every 45 s in real time. We observed two different unfolding pathways, one where secondary contacts were broken in the P1 loop, and another where contacts were broken in and around the P5 loop, which is near the tethered end point. We defined unfolded and folded states using an order parameter that measures the number of native contacts. If one were to use more than one order parameter, sampling could be enforced separately along these two pathways. This would allow for a more precise description of the competition between the two pathways for a given flow rate, and a description of the transition between the pathways of maximum probability as the flow rate changes. Work is currently underway to achieve this goal. The parallelization strategy presented here for piecewise sampling methods will enable treatment of increasingly complex order parameter spaces as large-scale computational architectures continue to grow in size.

AUTHOR INFORMATION

Corresponding Author

*E-mail: dinner@uchicago.edu.

ACKNOWLEDGMENT

We would like to thank Nicholas Guttenberg and Jonathan Weare for useful discussions on the algorithm and Glenna Smith and Norbert Scherer for help with the RNA model. We would also like to thank Lorenzo Pesce for help running NEUS on the Beagle Cray XE6 Supercomputer. This work was supported by National Science Foundation grant no. MCB-0547854, an Argonne−University of Chicago Strategic Collaborative Initiative Award, and the Natural Sciences and Engineering Research Council. Most of the calculations were run on "Fusion," a 320-node computing cluster operated by the Laboratory Computing Resource Center at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357. Scaling data were obtained for Intrepid, a Blue Gene/P supercomputer at the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357, and for Beagle, a Cray XE6 supercomputer, which is supported in part by NIH through resources provided by the Computation Institute, University of Chicago and Argonne National Laboratory, under grant S10 RR029030-01.

REFERENCES

(1) Liphardt, J.; Onoa, B.; Smith, S. B.; Tinoco, I.; Bustamante, C. *Science* **2001**, *292*, 733–737.
(2) Lin, Y.; Zhao, T.; Jian, X.; Farooqui, Z.; Qu, X.; He, C.; Dinner, A. R.; Scherer, N. F. *Biophys. J.* **2009**, *96*, 1911–1917.
(3) Comstock, M. J.; Ha, T.; Chemla, Y. R. *Nat. Methods* **2011**, *8*, 335–340.
(4) Sotomayor, M.; Schulten, K. *Science* **2007**, *316*, 1144–1148.
(5) Hu, J.; Ma, A.; Dinner, A. R. *J. Chem. Phys.* **2006**, *125*, 114101.
(6) Dellago, C.; Bolhuis, P. G. *Advanced Computer Simulation Approaches for Soft Matter Sciences III*; Springer-Verlag: Berlin, 2009; Vol. 221, pp 167−233.
(7) Vanden-Eijnden, E. *Annu. Rev. Phys. Chem.* **2010**, *61*, 391–420.
(8) Frenkel, D.; Smit, B. *Understanding Molecular Simulation: From Algorithms to Applications*; Academic Press: London, 2002; pp 192−196; 389−397; 431−462.
(9) Allen, R. J.; Warren, P. B.; ten Wolde, P. R. *Phys. Rev. Lett.* **2005**, *94*, 018104.
(10) Allen, R. J.; Frenkel, D.; ten Wolde, P. R. *J. Chem. Phys.* **2006**, *124*, 024102.
(11) Allen, R. J.; Frenkel, D.; ten Wolde, P. R. *J. Chem. Phys.* **2006**, *124*, 194111.
(12) Allen, R. J.; Valeriani, C.; ten Wolde, P. R. *J. Phys.: Condens. Matter* **2009**, *21*, 463102.
(13) Warmflash, A.; Bhimalapuram, P.; Dinner, A. R. *J. Chem. Phys.* **2007**, *127*, 154112.
(14) Dickson, A.; Warmflash, A.; Dinner, A. R. *J. Chem. Phys.* **2009**, *130*, 074104.
(15) Dickson, A.; Warmflash, A.; Dinner, A. R. *J. Chem. Phys.* **2009**, *131*, 154104.
(16) Dickson, A.; Dinner, A. R. *Annu. Rev. Phys. Chem.* **2010**, *61*, 441–59.
(17) Huber, G. A.; Kim, S. *Biophys. J.* **1996**, *70*, 97–110.
(18) Zhang, B. W.; Jasnow, D.; Zuckerman, D. M. *Proc. Natl. Acad. Sci. U.S.A.* **2007**, *104*, 18043–18048.
(19) Bhatt, D.; Zhang, B. W.; Zuckerman, D. M. *J. Chem. Phys.* **2010**, *133*, 014110.
(20) Bhatt, D.; Zuckerman, D. M. *J. Chem. Theory Comput.* **2010**, *6*, 3527–3539.
(21) Zhang, B. W.; Jasnow, D.; Zuckerman, D. M. *J. Chem. Phys.* **2010**, *132*, 054107.
(22) Smith, G. J.; Lee, K. T.; Qu, X.; Xie, Z.; Pesic, J.; Sosnick, T. R.; Pan, T.; Scherer, N. F. *J. Mol. Biol.* **2008**, *378*, 943–953.
(23) Qu, X.; Smith, G. J.; Lee, K. T.; Sosnick, T. R.; Pan, T.; Scherer, N. F. *Proc. Natl. Acad. Sci. U.S.A.* **2008**, *105*, 6602–6607.
(24) Li, Y.; Qu, X. H.; Ma, A.; Smith, G. J.; Scherer, N. F.; Dinner, A. R. *J. Phys. Chem. B* **2009**, *113*, 7579–7590.
(25) Delgado-Buscaglioni, R. *Phys. Rev. Lett.* **2006**, *96*, 088303.
(26) Chandler, D. *Introduction to Modern Statistical Mechanics*; Oxford University Press: New York, 1987; pp 168−175.
(27) Vanden-Eijnden, E.; Venturoli, M. *J. Chem. Phys.* **2009**, *131*, 044120.
(28) Nieplocha, J.; Harrison, R. J.; Littlefield, R. J. *J. Supercomput.* **1996**, *10*, 169–189.

(29) Hyeon, C.; Thirumalai, D. *Biophys. J.* **2007**, *92*, 731–743.
(30) Kremer, K.; Grest, G. S. *J. Chem. Phys.* **1990**, *92*, 5057–5086.
(31) Weeks, J. D.; Chandler, D.; Andersen, H. C. *J. Chem. Phys.* **1971**, *54*, 5237.
(32) Kazantsev, A. V.; Krivenko, A. A.; Harrington, D. J.; Holbrook, S. R.; Adams, P. D.; Pace, N. R. *Proc. Natl. Acad. Sci. U.S.A.* **2005**, *102*, 13392–13397.
(33) Malevanets, A.; Kapral, R. *J. Chem. Phys.* **1999**, *110*, 8605–8613.
(34) Ihle, T.; Kroll, D. M. *Phys. Rev. E* **2001**, *63*, 020201.
(35) Lamura, A.; Gompper, G.; Ihle, T.; Kroll, D. M. *Europhys. Lett.* **2001**, *56*, 319–325.
(36) Allahyarov, E.; Gompper, G. *Phys. Rev. E* **2002**, *66*, 036702.
(37) Kikuchi, N.; Pooley, C. M.; Ryder, J. F.; Yeomans, J. M. *J. Chem. Phys.* **2003**, *119*, 6388–6395.
(38) Webster, M. A.; Yeomans, J. M. *J. Chem. Phys.* **2005**, *122*, 164903.