Solvent Viscosity Dependence of the Folding Rate of a Small Protein: Distributed Computing Study

BOJAN ZAGROVIC, VIJAY PANDE
Biophysics Program and Department of Chemistry, Stanford University, Stanford, CA 94305

Received 16 December 2002; Accepted 12 March 2003

Abstract: By using distributed computing techniques and a supercluster of more than 20,000 processors we simulated folding of a 20-residue Trp Cage miniprotein in atomistic detail with implicit GB/SA solvent at a variety of solvent viscosities (γ). This allowed us to analyze the dependence of folding rates on viscosity. In particular, we focused on the low-viscosity regime (values below the viscosity of water). In accordance with Kramers' theory, we observe approximately linear dependence of the folding rate on $1/\gamma$ for values from $1–10^{-1}\times$ that of water viscosity. However, for the regime between $10^{-4}–10^{-1}\times$ that of water viscosity we observe power-law dependence of the form $k \sim \gamma^{-1/5}$. These results suggest that estimating folding rates from molecular simulations run at low viscosity under the assumption of linear dependence of rate on inverse viscosity may lead to erroneous results.

© 2003 Wiley Periodicals, Inc. J Comput Chem 24: 1432–1436, 2003

Key words: protein folding; Langevin dynamics; distributed computing; solvent viscosity; folding rate

Introduction

The rate of protein folding is assumed to be fundamentally limited by the rate of diffusion of the parts of the protein chain in the solvent.[1] Most theoretical treatments linking the rate of activated chemical processes in solution with diffusion are based on a theory originally developed by Kramers.[2] He proposed that the reaction rate of a diffusion-limited process in the high friction limit should be proportional to $1/\gamma$, inverse of the viscosity of the solvent. This proposition has been tested experimentally in the case of protein folding by several groups[3–7] and has in general proved to be correct. However, all of these studies have explored viscosities equal to or higher than the viscosity of water. The reason for this is simple: in a test tube, viscosity can be increased relatively easily by using viscogenic agents, while going the other way is difficult, if not impossible.

Simulating the process of protein folding in atomistic detail *in silico* has been one of the premier challenges of modern computational biology. Because the gap between the computationally accessible regime (nanoseconds) and what can be measured experimentally (microseconds and higher) is still large, different ingenious methods have been devised to speed the sampling while still attempting to retain physical accuracy. One of the commonly employed methods is to artificially decrease or completely eliminate the solvent viscosity in Langevin or Brownian dynamics-type simulations.[8–11] The rationale behind this is clear: if the rate of protein folding is inversely proportional to solvent viscosity, as proposed by Kramers, decreasing the viscosity should increase the rate of folding in the same proportion. Qualitatively, the lower the viscosity, the faster the protein chain will diffuse, and the faster it will reach the native topology. Moreover, if the inverse dependence on viscosity can be trusted quantitatively one could in principle even estimate the folding rate in water from the rate obtained in low-viscosity simulations. However, as mentioned above, the validity of Kramers' theory has been tested experimentally only for a limited range of viscosities higher than that of water. Further, because until recently simulating complete folding events at water viscosities was impossible, it is completely unclear to what degree Kramers' relation is applicable to the low-viscosity regime and to what degree it is justified to estimate folding rates by linear extrapolation from the low-viscosity results.

Recently, we developed a way of utilizing distributed computing techniques and large clusters of loosely coupled processors to

Correspondence to: V. Pande; e-mail: pande@stanford.edu

Contract/grant sponsor: ACS PRF; contract/grant number: 36028-AC4

Contract/grant sponsor: NIH BBCA; contract/grant number: R01GM62868-01A2)

Contract/grant sponsor: NSF MRSEC CPIMA; contract/grant number: DMR-9808677)

Contract/grant sponsor: NIH BISTI; contract/grant number: IP20 GM64782-01

Contract/grant sponsor: ARO; contract/grant number: 41778-LS-RIP

Contract/grant sponsor: Stanford University; contract/grant number: Internet 2

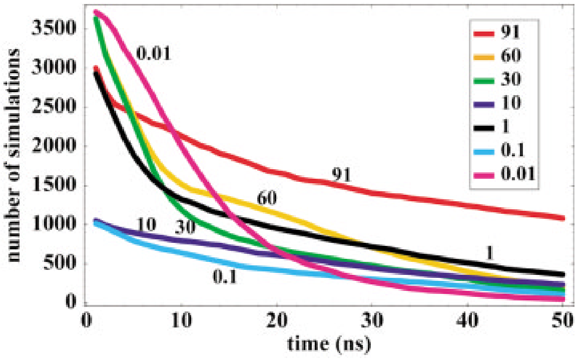

Figure 1. Total number of simulated trajectories that have reached a given timepoint at all values of viscosity used. Our distributed computing methodology results in a number of independent trajectories of different length: for the analysis in this article we included all the trajectories that have reached at most 50 ns. The numbers in the figure refer to the solvent viscosity used (ps^{-1}).

simulate protein folding on the tens of microsecond timescale.[12–18] This allowed us to generate for the first time complete folding trajectories of small proteins in atomistic detail. In particular, recently we reported a study of folding of a 20-residue Tryptophan Cage[16] and estimated the folding rate in water. In the present study, we simulated multiple folding events of the Tryptophan Cage starting from a fully extended conformation at different viscosities ranging from very low (10^4 times less viscous than water) to the full viscosity of water. We generated thousands of folding trajectories for each value of viscosity, each tens of nanoseconds long. Because this molecule is known to fold with single-exponential kinetics, we could estimate the folding rate at each value of viscosity by looking at what fraction of our simulations have folded in the tens of nanoseconds of simulated time (see Methods). The main question we ask is what is the dependence of the folding rate on solvent viscosity in our model. Finally, we discuss the implications of using low-viscosity simulation to study the nature of the mechanism of protein folding.

Methods

Using a large, heterogeneous computer cluster (over 20,000 processors), we generated thousands of relatively short (tens of nanoseconds) independent trajectories for 20-residue Tryptophan Cage polypeptide (sequence: NLYIQWLKDG GPSSGRPPPS) at a range of solvent viscosities (Fig. 1). The structure of the folded molecule was determined by NMR:[19,20] it contains a short α-helix from residues 2 to 8, a 3_{10}-helix from residues 11 to 14, and a C-terminal polyproline II helix that packs against the central tryptophan. Hagen and coworkers recently reported experimental evidence for cooperative two-state folding with a time constant for folding of 4 μs at room temperature.[21]

Our folding simulations were started from fully extended conformations ($\phi = -135°$, $\psi = 135°$). In addition, we also performed more than 100 μs of simulation starting from the experimental structure of the molecule to confirm the stability of our model and explore the variations of the native basin.[16] Even though the simulations are all started from the same structure (fully extended or native), they quickly diverge from each other due to the stochastic aspects of Langevin dynamics (see below). The simulations run using our modified version of the Tinker biomolecular simulation package (http://dasher.wustl.edu/tinker/) involved Langevin dynamics in implicit GB/SA[22] solvent with a 2-fs integration step at 300 K. Bond lengths were constrained using RATTLE.[23] No cutoffs were used for electrostatics. The protein was modeled using the OPLSua force field.[24] The solvent viscosity of water in our simulations was set to $\gamma = 91$ ps^{-1}. We ran thousands of independent trajectories at different solvent viscosities ranging from $10^{-4}\times$ to $1\times$ that of water. Coordinates were output every 1 ns. The simulations were carried out on more than 20,000 processors as a part of our ongoing Folding@Home distributed computing project (http://folding.stanford.edu), and involved a total of about a quarter of a trillion ($2.4 \cdot 10^{11}$) integration steps for a total of more than 450 μs of simulated time (Fig. 1). This corresponds to approximately 1 million single CPU (500 MHz) days of computation. Table 1 details the particulars of different folding simulations run at varying values of solvent viscosity, including the total amount of simulated time, as well as the total number of folding events observed (columns 2 and 3). In Figure 1, we show the total number of individual trajectories that have reached a given timepoint for all values of solvent viscosity used. The longest trajectories used for the analysis in this report were 50 ns long.

The folding rate and the associated error were estimated from our simulations in the following way. We assume (as seen experimentally[21]) that the folding of the Tryptophan Cage exhibits

Table 1. Different Folding Simulations.

Viscosity (ps^{-1})	Total simulated time (μs)	Number of folding events	Folding time $\tau_{\text{MLE}} = 1/K_{\text{MLE}}$ (ns)
91	82.8	38	2228 ± 361
60	53.7	38	1435 ± 232
30	42.5	60	726 ± 94
10	27.6	90	323 ± 34
1	48.8	265	201 ± 12
0.1	20.5	176	129 ± 10
0.01	47.5	711	75 ± 3

single exponential behavior. In other words, the probability that a given molecule has folded between time t and $t + dt$ equals

$$P_{\text{folded}}(t, t + dt) = Ke^{-Kt}dt, \tag{1}$$

where K corresponds to the folding rate. Consequently, the probability that a molecule has not folded by time t equals

$$P_{\text{folded}}(T > t) = 1 - \int_0^t Ke^{-Kt}dt = e^{-Kt}. \tag{2}$$

In the case of an ensemble of N mutually independent folding processes each of different length (as is the case in our simulations, see Fig. 1), the likelihood of observing a total of n folding events, $L(n)$, each at time between t_i and $t_i + dt$, is therefore equal to

$$L(n) = \prod_{i=1}^{n} Ke^{-Kt_i}dt \prod_{j=1}^{N-n} e^{-Kt_j}, \tag{3}$$

where $\Pi_{i=1}^{n} Ke^{-Kt_i}dt$ is the probability that n have folded each between time t_i and $t_i + dt$ and $\Pi_{j=1}^{N-n} e^{-Kt_j}$ is the probability that the remaining $N - n$ trajectories have not folded by time t_j (length of each trajectory) each.

The log-likelihood of observing a total of n folding events each between time t_i and $t_i + dt$, $LL(n)$, then equals (by taking the log)

$$LL(n) = n \ln K - K \sum_{i=1}^{n} t_i - K \sum_{j=1}^{N-n} t_j. \tag{4}$$

By the principle of maximum likelihood,[25] the best estimator for the rate K given N, n, and each of t_i and t_j is the value that maximizes the log-likelihood. This can be determined by finding the first derivative of $LL(n)$ with respect to K and equating it with 0.

$$\frac{\partial}{\partial K} LL(n) = n\frac{1}{K} - \sum_{i=1}^{n} t_i - \sum_{j=1}^{N-n} t_j = 0 \tag{5}$$

or

$$K_{\text{MLE}} = \frac{n}{\sum_{i=1}^{n} t_i + \sum_{j=1}^{N-n} t_j}, \tag{6}$$

where t_i are the folding times for the n folding events and t_j are the lengths of the $N - n$ trajectories that have not resulted in folding. This gives the maximum likelihood estimate (MLE) for the folding rate.

The lower bound on the error of the above maximum likelihood estimate is given by the Cramer–Rao variance[25] as

$$\text{var} = \frac{-1}{E\left(\dfrac{\partial^2 LL(n, T)}{\partial K \partial K'}\right)} = \frac{n}{(\sum_{i=1}^{n} t_i + \sum_{j=1}^{N-n} t_j)^2} \tag{7}$$

or

$$\text{std} = \frac{\sqrt{n}}{\sum_{i=1}^{n} t_i + \sum_{j=1}^{N-n} t_j}. \tag{8}$$

The folding time, $\tau = 1/K$, is given by the following (the error is obtained by propagation of error from the above expression for the error in K_{MLE}):

$$\tau_{\text{MLE}} = \frac{\sum_{i=1}^{n} t_i + \sum_{j=1}^{N-n} t_j}{n} \pm \frac{\sum_{i=1}^{n} t_i + \sum_{j=1}^{N-n} t_j}{n^{3/2}}. \tag{9}$$

Note that in the above derivations we assumed that a molecule that has folded does not unfold after that (within the simulated time), i.e., we assumed that a folding event can be thought of as an absorbing boundary. Because in our simulations we simulate trajectories on a short timescale compared with the overall folding and unfolding times (nanoseconds vs. microseconds), this assumption holds well.

To use the above equations for estimating folding rates, we need a precise definition of when a molecule has folded. In this study, we define a molecule to be folded if its Cα root mean square deviation (RMSD) from the experimental structure is 2.5 Å or less, and unfolded otherwise. Our simulations of the native basin of TrpCage[16] indicate that the native state contains structures that are on average 2.5 Å away from the experimental structure, so we decided to use this as a cutoff defining the folded state in our simulations started from the extended state. In Table 1, we show the MLEs for the folding times at different viscosities including the associated error.

Results and Discussion

Our primary result is shown in Figure 2. We plot the folding time ($\tau = 1/K$) vs. viscosity (γ) for several different values of γ. We see an approximately linear relationship between folding time and viscosity $\tau \sim \gamma$ (shown as a line with slope $\approx$0.9 in Fig. 1) for a regime near the water viscosity (10 ps^{-1} < γ < 100 ps^{-1}). At lower values of γ, we see a $\tau \sim \gamma^{1/5}$ power–law relationship (line in Fig. 1 with slope $\approx$0.2) between the folding time and viscosity parameter.

Thus, we see that while the Kramers relationship $\tau \sim \gamma$ is preserved for near water-like viscosities this does not continue for values of γ typically used in low-viscosity simulations (e.g., 1 ps^{-1} or smaller). What are the implications of this result? From a practical point of view, our results suggest that modest reduction in viscosity (to at most 1/10 that of water) may be used in stochastic simulations to speed folding rates of small proteins while still allowing one to predict rates in water-like viscosity using simple Kramers'-type extrapolation. However, it is also clear that very low-viscosity simulations (below 1/10 that of water) cannot be used to predict folding rates—at least not by simple extrapolation that $\tau \approx \tau_{\text{low}} (\gamma_{\text{water}} / \gamma_{\text{low}})$. If one were to do this for our Trp Cage simulations, the predicted rates would typically be off by one to two orders of magnitude.

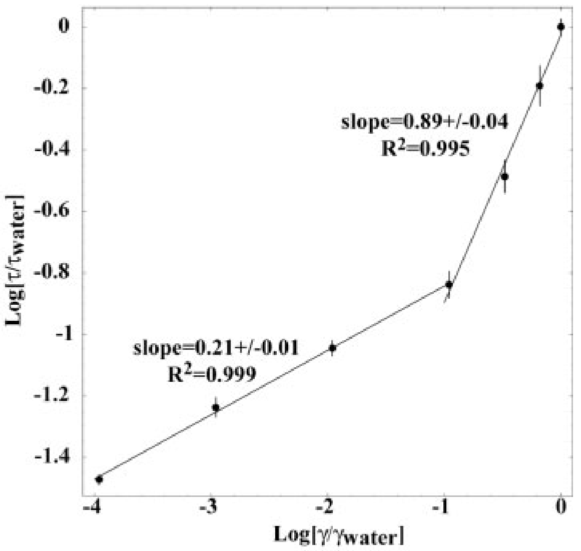

Figure 2. Viscosity dependence of the folding time of the Tryptophan Cage molecule. The folding times and associated errors were calculated based on our simulations using the maximum-likelihood approach (see Methods). In this figure, both the folding times and viscosities are given relative to the folding time in water and the viscosity of water, respectively. The error bars given are error propagated on the basis of the Cramer–Rao errors for the individual folding times. The logs are base 10.

What could be the cause of this deviation in folding rate? Consider two limiting scenarios for how a protein might fold. On one extreme, proteins fold directly into the folded state; in this case, the protein collapse will be rate limiting. On the other extreme, proteins first fold to a random globular conformation, and then motions in the collapsed, globular state continue until the protein folds; in this case, the protein internal friction will be the rate-limiting aspect. Thus, as one decreases the viscosity from water-like viscosities to essentially no protein–solvent friction one would expect that the protein–protein ("internal") friction would become the rate-limiting aspect. This crossover would impact both the rate as well as the mechanism seen in low-viscosity folding simulations.

This conjecture is consistent with our results. As we decrease γ, we see a reduced dependence of the folding rate on γ. The Trp Cage is relatively small and even when globular has a significant fraction of residues with exposed surface area (and thus still some protein–solvent friction). For larger proteins, a considerably smaller fraction will be exposed and thus we expect that this difference between water-like and low viscosities should be even further exaggerated.

Several groups studied the viscosity dependence of conformational changes induced in heme proteins by the binding of CO or O_2 and noted similar effects to what we see.[26–29] Eaton and coworkers studied the effects of changing viscosity on the conformational changes in myoglobin and were able to fit their data to a simple model of the form[28,29]

$$k \approx \frac{C}{\sigma + \gamma}\, e^{-E/RT}, \tag{10}$$

where C is an adjustable parameter, E is the size of the energy barrier to conformational change, R is the gas constant, T is temperature, and γ is the viscosity of the solvent. The parameter σ was interpreted as the contribution of the protein friction to total friction. In their study, similar to our results, they differentiated between solvent–friction-dominated and protein–friction-dominated regimes. Note that our results could be easily fit to a model of the above type. However, in the case of conformational changes in myoglobin the transition between the two regimes described above occurred around 1 cP ($\approx$viscosity of water), an order of magnitude or so larger than what we see. It is possible that Trp Cage, being a much smaller molecule than myoglobin, is more susceptible to changes in solvent viscosity even at lower values of viscosity. After all, the fraction of solvent-exposed atoms in Trp Cage is much greater than in myoglobin. Further, protein folding is an extreme example of conformational change, involving large motions of the protein chain, and it is likely that its dependence on solvent viscosity is more pronounced.

Conclusions

It is natural to ask what the consequences of simulating with reduced viscosity are. For thermodynamic properties, the viscosity is irrelevant—all that matters is that one has sufficiently sampled the Boltzmann ensemble; indeed, reduced viscosity could likely help achieve greater sampling. However, for kinetic properties the implications of reduced viscosity are less clear. If we look at the kinetics of simple systems, such as the isomerization of around a bond,[30] we see a nonmonotonic relationship between viscosity and the rate. At high viscosities, there is a low rate because the system is slowed by the viscous nature of the solvent, as one would expect. As one reduces the viscosity, the rate should increase. However, if the viscosity is reduced too far the rate can begin to increase again; this is due to the well-known connection between the solvent viscosity and the random thermal fluctuations it creates. It is these fluctuations that allow the system to cross energy barriers, and thus if the system is too greatly disconnected from the solvent (i.e., the viscosity is too low) then the rate for barrier crossing can decrease.

Thus, we see that the relationship between rate and viscosity can be nontrivial for simple, energy barrier crossing problems. What about more complex free energy barrier crossing problems, such as protein folding or conformational change? Should one expect a similar nonmonotonic relationship? Some experimental evidence points in this direction.[26–29] Further, results from simplified models suggest this to be the case.[31] However, these models use the classic Langevin integrator with a coarse-grained representation for the protein. Would a similar relationship be seen with more complex, accurate atomistic models?

We find a similar, nonmonotonic relationship. Perhaps of more practical value, even in the monotonic regime of the relationship, we do not see a linear scaling between folding time and viscosity, and thus significantly reduced viscosity simulations cannot be used to predict folding rates. Moreover, we suggest that these deviations are due to differences in the very folding mechanism (because different solvent–protein vs. protein–protein friction values will drive the system to different pathways). Thus, it is also likely that reduced viscosity simulation may not yield predictive results for folding mechanisms as well.

Acknowledgments

The authors thank all the contributors to the Folding@Home project; a complete list can be found at http://folding.stanford.edu. They thank all of the members of the Pande group for help and discussion. B.Z. acknowledges support from the HHMI Predoctoral Fellowship program. This work was supported by grants from the as well as by gifts from the Intel and Google Corps.

References

1. Eaton, W. A.; Munoz, V.; Hagen, S. J.; Jas, G. S.; Lapidus, L. J.; Henry, E. R.; Hofrichter, J. Annu Rev Biophys Biomol Struct 2000, 29, 327.
2. Kramers, H. A. Physica 1940, 7, 284.
3. Haas, E.; Katchalski-Katzir, E.; Steinbert, I. Z. Biopolymers 1978, 17, 11.
4. Chrunyk, B. A.; Matthews, C. R. Biochemistry 1990, 29, 2149.
5. Waldburger, C. D.; Jonsson, T.; Sauer, R. T. Proc Natl Acad Sci USA 1996, 93, 2629.
6. Plaxco, K. W.; Baker, D. Proc Natl Acad Sci USA 1998, 95, 13591.
7. Bhattacharyya, R. P.; Sosnick, T. R. Biochemistry 1999, 38, 2601.
8. Ferrara, P.; Caflisch, A. Proc Natl Acad Sci USA 2000, 97, 10780.
9. Cavalli, A.; Ferrara, P.; Caflisch, A. Proteins 2002, 47, 305.
10. Ferrara, P.; Apostolakis, J.; Caflisch, A. Proteins 2002, 46, 24.
11. Simmerling, C.; Strockbine, B.; Roitberg, A. E. J Am Chem Soc 2002, 124, 11258.
12. Shirts, M. R.; Pande, V. S. Phys Rev Lett 2001, 86, 4983.
13. Shirts, M. R.; Pande, V. S. Science 2001, 290, 1903.
14. Zagrovic, B.; Sorin, E. J.; Pande, V. J Mol Biol 2001, 313, 151.
15. Snow, C. D.; Nguyen, H.; Pande, V. S.; Gruebele, M. Nature 2002, 420, 102.
16. Snow, C. D.; Zagrovic, B.; Pande, V. S. J Am Chem Soc 2002, 124, 14548.
17. Zagrovic, B.; Snow, C.; Khaliq, S.; Shirts, M.; Pande, V. J Mol Biol 2002, 323, 153.
18. Zagrovic, B.; Snow, C. D.; Shirts, M. R.; Pande, V. S. J Mol Biol 2002, 323, 927.
19. Gellman, S. H.; Woolfson, D. N. Nat Struct Biol 2002, 9, 408.
20. Neidigh, J. W.; Fesinmeyer, R. M.; Andersen, N. H. Nat Struct Biol 2002, 9, 425.
21. Qiu, L.; Pabit, S. A.; Roitberg, A. E.; Hagen, S. J. J Am Chem Soc 2002, 124, 12952.
22. Qiu, D.; Shenkin, P. S.; Hollinger, F. P.; Still, W. C. J Phys Chem 1997, 3005.
23. Andersen, H. C. J Comput Phys 1983, 52, 24.
24. Jorgensen, W. L.; Tirado-Rives, J. J Am Chem Soc 1988, 110, 1666.
25. Hogg, R. V.; Craig, A. T. Introduction to Mathematical Statistics; Prentice-Hall: Englewood Cliffs, NJ, 1994.
26. Hasinoff, B. B. Arch Biochem Biophys 1981, 211, 396.
27. Lavalette, D.; Tetreau, C. Eur J Biochem 1988, 177, 97.
28. Ansari, A.; Jones, C. M.; Henry, E. R.; Hofrichter, J.; Eaton, W. A. Science 1992, 256, 1796.
29. Ansari, A.; Jones, C. M.; Henry, E. R.; Hofrichter, J.; Eaton, W. A. Biochemistry 1994, 33, 5128.
30. Chandler, D. J Chem Phys 1978, 68, 2959.
31. Klimov, D. K.; Thirumalai, D. Phys Rev Lett 1997, 79, 317.