

Brownian dynamics simulations of protein folding: Access to milliseconds time scale and beyond

ATIPAT ROJNUCKARIN[†], SANGTAE KIM^{†‡}, AND SHANKAR SUBRAMANIAM[§]

[†]Department of Chemical Engineering, University of Wisconsin–Madison, Madison, WI 53706; and [§]Departments of Molecular and Integrative Physiology and Chemical Engineering, Center for Biophysics and Computational Biology, Beckman Institute for Advanced Science and Technology, and National Center for Supercomputing Applications, University of Illinois at Urbana–Champaign, Champaign, IL 61801

Communicated by Edwin N. Lightfoot, Jr., University of Wisconsin–Madison, Madison, WI, January 23, 1998 (received for review July 21, 1997)

ABSTRACT Protein folding occurs on a time scale ranging from milliseconds to minutes for a majority of proteins. Computer simulation of protein folding, from a random configuration to the native structure, is nontrivial owing to the large disparity between the simulation and folding time scales. As an effort to overcome this limitation, simple models with idealized protein subdomains, e.g., the diffusion–collision model of Karplus and Weaver, have gained some popularity. We present here new results for the folding of a four-helix bundle within the framework of the diffusion–collision model. Even with such simplifying assumptions, a direct application of standard Brownian dynamics methods would consume 10,000 processor-years on current supercomputers. We circumvent this difficulty by invoking a special Brownian dynamics simulation. The method features the calculation of the mean passage time of an event from the flux overpopulation method and the sampling of events that lead to productive collisions even if their probability is extremely small (because of large free-energy barriers that separate them from the higher probability events). Using these developments, we demonstrate that a coarse-grained model of the four-helix bundle can be simulated in several days on current supercomputers. Furthermore, such simulations yield folding times that are in the range of time scales observed in experiments.

Molecular and Brownian dynamics simulations (1, 2), as well as lattice Monte Carlo simulations (3, 4), have been used to investigate protein-folding pathways with some success. However, the time scales accessible by the dynamics simulation techniques are in the microseconds range or less and thus fall short of the experimentally observed protein-folding time scales, which, for most proteins, lie somewhere between a few milliseconds to minutes. The limit on accessible time scales in these simulations originates from the small time steps required for the accurate integrations of the relevant equations of motion; a simulation with too long a characteristic time scale simply takes too many time steps and consumes too much computer resources. Developed specifically to handle simulations of large-time-scale processes, the Weighted-Ensemble Brownian (WEB) dynamics algorithm of Huber and Kim (5), a biased Brownian dynamics scheme, can overcome the time scale restriction in the traditional simulation techniques through judicious splitting and combining of Brownian particles. In this paper, we show that the WEB dynamics idea can be generalized to the simulation of a simple diffusion–collision four-helix-bundle protein-folding model, allowing us to investigate its folding kinetics in a reasonable time frame.

We develop our four-helix-bundle protein-folding model from the diffusion–collision approach of Karplus and Weaver

(6–8). The diffusion–collision protein-folding mechanism postulates the early-stage formation of fluctuating quasiparticles, called microdomains, which may be incipient secondary structures (α -helices and β -sheets) or hydrophobic clusters. These microdomains move diffusely, and their coalescence leads to the formation of folded proteins. Consequently, the diffusion–collision model reduces the complexity of the folding process from a consideration of the individual amino acids to that of the properties of few microdomains and their interactions. Experimental evidence supporting this model comes from folding experiments where elements of secondary structure can be detected early in the folding process (9–11), as well as other experiments that show the tendency of various polypeptide fragments to form stable secondary structures in solution (8, 12, 13). The diffusion–collision model has been used to estimate folding-rate constants in proteins where diffusion–collision of microdomains is the rate-limiting step (14–18).

The four-helix bundle is a well defined motif in several proteins (19, 20); *de novo* designs of four-helix-bundle proteins have also been reported (21). Theoretical studies of folding kinetics of a four-helix bundle protein have been carried out by Yapa and Weaver (17, 18) in the context of the diffusion–collision model and the chemical kinetics approximation. In this approximation, the analytical rate constants for pairwise coalescence of spherical microdomains or microdomain complexes can be used to construct a system of first-order differential equations that can be solved analytically to determine the folding time of the protein and the relative population of various folding intermediates (7). To obtain analytical rate constants for microdomain coalescence, one must make several simplifying assumptions. Using the WEB dynamics algorithm allows us to retain the diffusion–collision framework while allowing more details of microdomain geometry and interactions in our model and bypassing the chemical kinetics approximation all together. Although some qualitative agreement with experimental results can be obtained from our model, we emphasize that our major objective is to demonstrate the applicability of the WEB dynamics method to the protein-folding problem. More definitive simulations of protein-folding kinetics may come in the future, perhaps by reinvesting the savings from our algorithmic gains on more accurate force calculations.

The remaining portions of this paper are organized as follows. The next section describes the folding model in the context of the diffusion–collision framework and the adaptation of the WEB dynamics algorithm to the four-helix model. The subsequent section on results and discussion explores the relationship between the model parameters (e.g., helix length, strength of the electrostatic charges) and the folding time. We

The publication costs of this article were defrayed in part by page charge payment. This article must therefore be hereby marked “advertisement” in accordance with 18 U.S.C. §1734 solely to indicate this fact.

© 1998 by The National Academy of Sciences 0027-8424/98/954288-5\$2.00/0
PNAS is available online at <http://www.pnas.org>.

Abbreviation: WEB: Weighted-Ensemble Brownian.

[‡]To whom reprint requests should be addressed at the present address: Scientific Information Resources, Parke–Davis Pharmaceutical Research, 2800 Plymouth Road, Ann Arbor, MI 48105. e-mail: kim01@aa.WL.com.

conclude with some thoughts on future directions for “WEB-based” protein-folding simulations.

MATERIALS AND METHODS

Diffusion–Collision Four-Helix-Bundle Protein Model.

Four rigid nonoverlapping sphero-cylinders connected to one another by frictionless strings were used to model four helical microdomains, which must diffuse, collide, and aggregate to form the folded four-helix bundle (see Fig. 1). The first interaction scheme serves only as constraints that preserve the topological contiguity of the protein, and they exert no additional force on the microdomains. In addition, two coarse-grained helix–helix interaction schemes are employed and compared in this work to gauge the effects of simulation details on our results. The interaction schemes are designed to mimic the helix-orienting effects of helix dipoles (22). The first interaction scheme puts pairs of opposite charges of $\pm 0.5e$ at the cap centers of the four helical microdomains (Fig. 2a). The other scheme employs point dipoles located at the centers of microdomains (Fig. 2b). The magnitudes of these dipoles are set to $0.3e$ per helix length (measured, in Å, from one cap center to another); the reason for selecting this value will be discussed in detail in the following section. Both helix–helix interaction schemes are expected to help steer the microdomains into the desired up–down–up–down helix arrangement in the folded form.

In defining the folding criteria of our protein model, it is noticed that in the folded configuration (Fig. 1a), the cap centers of the sphero-cylindrical microdomains line up to form two quadrilaterals (denoted by white sticks in Fig. 1b), one at each end of the helix bundle. When the protein

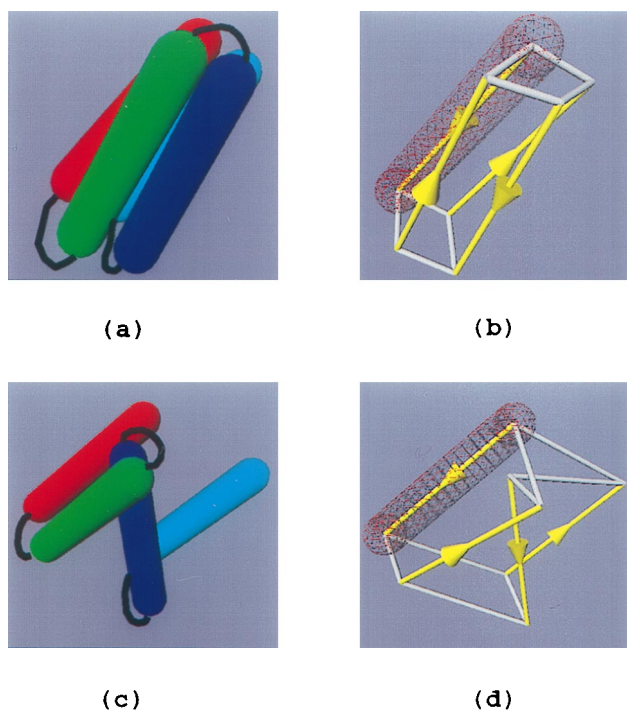


FIG. 1. Four-helix bundle protein model. (a) Folded protein. (b) Schematic drawing of *a*. Helix major axes, connecting the center of hemispherical caps, are in yellow (arrows point to N terminus). The two quadrilaterals that are formed by the helix ends are outlined in white. The length of the longest edge of the two quadrilaterals, d , is used as the folding criterion, as well as the guideline for splitting and combining WEB particles. The wire-framed rendition of the exclusion volume of the first helix (red) is shown for comparison. In this case, $d = 10\text{Å}$. (c) Unfolded protein. (d) A schematic drawing of *c*, showing major axes and distances between helix ends ($d = 27.2\text{Å}$).

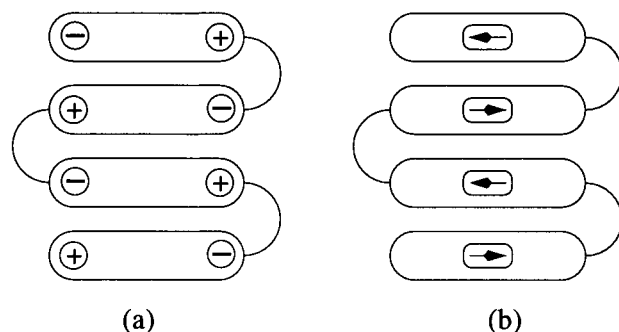


FIG. 2. Helix–helix interaction models. (a) Point-charge model. (b) Point-dipole model.

unfolds (Fig. 1c), some of the edges of the quadrilaterals defined by cap centers elongate (Fig. 1d). A protein satisfies our folding criteria when the *longest edge* of the aforementioned quadrilaterals is *shorter* than 10Å (1.25 times the helix diameter). Because the microdomains are nonoverlapping, this folding criterion is found to be very stringent.

Because rigid microdomains are used, the configuration of a protein in our simulation is defined by the Cartesian coordinates of helix centers, \mathbf{x}_i , and quaternion parameters, \mathbf{q}_i , representing their orientations (23–25). Starting from randomly generated configurations, the configurations of the four helices are simulated by using the conventional equations of motion:

$$d\mathbf{x}_i = \frac{1}{k_B T} \mathbf{D}_i^t \cdot \mathbf{F}_i dt + \sqrt{2\mathbf{D}_i^t} \cdot d\mathbf{W}_i^t \quad [1]$$

$$d\omega_i = \frac{1}{k_B T} \mathbf{D}_i^r \cdot \mathbf{T}_i dt + \sqrt{2\mathbf{D}_i^r} \cdot d\mathbf{W}_i^r \quad [2]$$

$$d\mathbf{q}_i = \begin{pmatrix} q_{0i} & -q_{1i} & -q_{2i} & -q_{3i} \\ q_{1i} & q_{0i} & -q_{3i} & q_{2i} \\ q_{2i} & q_{3i} & q_{0i} & -q_{1i} \\ q_{3i} & -q_{2i} & q_{1i} & q_{0i} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ d\omega_{ix}^{\text{BF}} \\ d\omega_{iy}^{\text{BF}} \\ d\omega_{iz}^{\text{BF}} \end{pmatrix} \quad [3]$$

where \mathbf{F}_i and \mathbf{T}_i denote force and torque on helix i , \mathbf{D}_i^t and \mathbf{D}_i^r denote the translational and rotational diffusion matrices of helix i , $d\mathbf{W}$ denotes the Gaussian noise with variance of dt (26, 27), and the superscript BF denotes the evaluation in the body-fixed frame of reference. Hydrodynamics interaction (28) is neglected in all of our simulations. Because it has been shown that the hydrodynamics interaction plays only a secondary role to electrostatic steering in the enzyme–substrate association process (29), it is expected that the hydrodynamics interaction also plays a minor role in the folding process as well. In this work, the translational and rotational diffusion constants of each isolated helix are calculated by using the completed double-layer boundary element integral method (30, 31) and are plotted in Fig. 3. The equations of motion above are solved by using the algorithm of Ermak and McCammon (32) with the quaternion parameters being renormalized at every time step.

At this point, we have not included the effects order-disorder transition of the microdomains (8) and short-range stabilizing interactions such as van der Waals and hydrophobic interactions (33, 34). Because of the very simplistic nature of our protein model, the principal emphasis of our investigation lies in the applicability and efficiency of the simulation algorithm, rather than the simulation results obtained thereof. In light of previous experimental results (8–10, 35), the use of preformed rigid secondary structure elements as microdomains in our simulation, as well as our rudimentary helix–helix interaction schemes, evidently is unrealistic. Nevertheless,

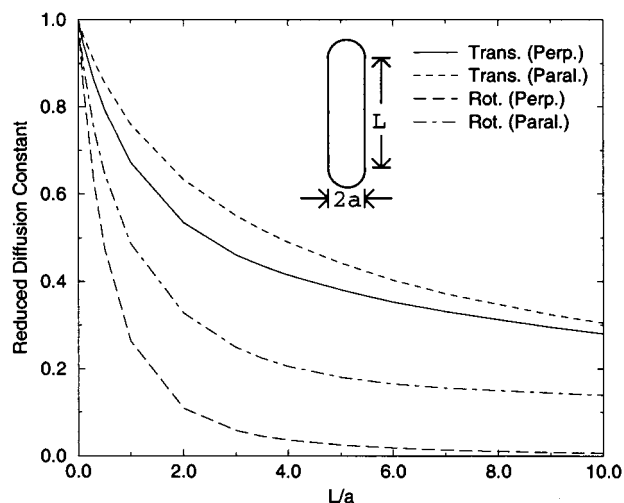


FIG. 3. Reduced translation/rotation diffusion constants of the sphero-cylinder model of helix microdomain in the direction parallel/perpendicular to the helix major axis are reported. In this figure, the diffusion constants are scaled by the corresponding translation/rotation diffusion constants of a sphere of the same radius and plotted against the ratio of the length of major axis of the sphero-cylinder L and its radius a .

irrespective of these simplifications, it suffices that this model, by the virtue of its characteristic folding time scale being comparable to that of the protein-folding process, should be an effective test for the WEB dynamics simulation algorithm.

WEB Dynamics Simulation. The underlying theoretical principle for the present Brownian dynamics simulation is identical to that described in Huber and Kim (5). For the most general formulation of the Brownian dynamics simulation, we consider diffusion in an N -dimensional configuration space, with an arbitrary potential and (possibly varying) diffusion matrix. Each configuration of the microdomain assembly would be described by a set of coordinates in this configuration space and thus be represented by a "particle" at that point in space. The folding time or the mean passage time of this particle in the configuration space is the average amount of time it takes a particle to traverse the configuration space to the region that corresponds to folded proteins.

Intuitively, one can obtain the mean passage time in question from computer simulations by either simulating many Brownian dynamic trajectories and averaging the results, or using the flux overpopulation method (36). The latter refers to a type of simulation where multiple Brownian dynamic trajectories are simulated at once, with the trajectories being immediately reinitialized once their destination is reached. The two methods have been shown to be mathematically equivalent, and the mean passage time for the second method equals the ratio of the number of the trajectories in the simulation and the number of the trajectories reaching their destination per unit time at the steady state (37). The main difficulty in determining the mean passage time using either simulation method is that most interesting systems require particles to cross one or more substantial free-energy barriers before reaching their destination. As a result, the particles spend most of their time wandering within local free-energy wells and only rarely surmount the barriers. To obtain good estimates of the mean passage times of such systems via computer simulation would have consumed vast amounts of computational resources. However, it was pointed out that the flux overpopulation method can be modified to obtain meaningful results in a more efficient manner; this is the essence of the weighted-ensemble exposition in Huber and Kim (5).

There are two ways to imagine a large collection of particles in a configuration space. The first way is to view them as actual

copies of the physical system in the real world. The second viewpoint is to regard the collection of particles as an estimate of the probability distribution of states taken by *one* system. In other words, each particle represents a packet of probability. If all particles are considered to carry the same amount of probability, then these two viewpoints are equivalent. However, if each particle is allowed to carry a different amount of probability, then adopting the second viewpoint introduces additional flexibility. For example, the probability distribution at the top of a substantial free-energy barrier would be poorly sampled by an ensemble of particles with equal weights, because very few, if any, particles would be near the barrier top. On the other hand, if the particles are endowed with variable statistical weight, then many particles with small statistical weights can be present at the barrier top allowing the corresponding region in the configuration space to be adequately sampled. The weighted ensemble scheme allows particles to be split and combined with one another so that some particles can reach the barrier top without breaking any rules of probability theory.

The implementation of the weighted-ensemble method first requires selection of a measure that represents the "progress" of each configuration in the simulation toward its destination. This measure, calculated from the current configuration, will be used in deciding whether WEB particles are split, combined, or left alone at each time step. This progress measure can be identified with the *reaction coordinate* when the simulation concerns the association of an enzyme and a substrate (5). For the protein-folding problem, the progress indicator can be identified with the "order parameter" in the spirit of the energy landscape treatment of protein-folding kinetics (38, 39). In our simulations, we define the progress measure, d , as the maximum length of the edges of the two quadrilaterals formed by helix ends. It should be noted that the same measure is used to determine the folding criteria, and when the progress measure falls below 10 \AA , the folding criterion is automatically satisfied. Because the progress measure is used simply as a guideline for the splitting and combining of particles, the folding pathway needs not be known in advance, and any reasonable progress measure, such as the free energy of each protein configuration, could be substituted without any loss of accuracy.

In WEB simulations, we first subdivide configuration space into regions, or bins, in intervals along the d -coordinate. Although the regions (bins) that correspond to small d are not easily accessible in standard Brownian dynamics simulations because of the free-energy barriers to folding, the WEB algorithm is designed to sample configurations in these regions of configuration space effectively by splitting a single particle in these regions into many particles with smaller weights. Moreover, after splitting, there is a good chance that at least one of the newly split particles will progress further to bins with even smaller d parameters. In addition, to prevent the number of particles in the WEB simulations to grow exponentially, we also need to combine Brownian particles in regions in configuration space that already contain enough Brownian particles for effective sampling. Hence, in our implementation of WEB dynamics, we split and combine particles to have the same number of particles in each bin. At every iteration, the particles are stepped forward one time step according to the standard Brownian dynamics algorithm. During the time step, particles might move from one bin to another. For bins with too many particles, we pick the two lightest particles and combine them using the combine algorithm as in Huber and Kim (5). The weight of the conglomerate particle is the sum of the weights of the constituents. The position of the conglomerate is the position of one of the original particles; the probability of a position being chosen is proportional to the weight of the original particle at that position. The combine algorithm is repeated until the number of particles in the bin has been

reduced to the target quota. For bins with too few particles, we pick the heaviest particle and split it into two particles of equal weight. Both children are inserted into the position in configuration space that was occupied by the parent. The splitting algorithm is repeated until the number of particles in the bin has been increased to the target quota.[§] After the particle readjustments, the next time step is taken and the process is repeated. The net result is that intervals with smaller values of d (closer to the folded state) receive particles, albeit of minute weight, and a greatly reduced computational effort is expended in “uninteresting” regions of configuration space. Lastly, as in the flux overpopulation method, particles that satisfy the folding criteria are immediately reinitialized in a random configuration, and because the total weight of WEB particles in our simulation is normalized to one, the folding time of our folding model in WEB simulations equals the inverse of the particle flux, which is the total weight of particles satisfying the folding criteria per unit of time.

Simulation Details. To demonstrate the effectiveness of the WEB dynamics algorithm, we chose to study the effects of microdomain geometry on the folding time because these effects cannot be thoroughly investigated with analytical methods such as the chemical kinetics approximation. The dimensions of the helix microdomains used in this study are as follows. The radius of each microdomain is 4 Å, and the length of the connecting string, measured between cap centers, is 12 Å. To study the effects of microdomain geometry, the length of the major axis is varied between 24, 32, and 40 Å. The simulations are carried out at 298 K, where the viscosity of solution (water) is 0.89 centipoise. In addition, because the folding process takes place in aqueous solution, the dielectric constant is set to that of water ($\epsilon = 78$). The random initial configuration is generated by first laying down the cap center that corresponds to the N terminus of the first helix, and a random unit vector is generated for the direction of the first helix. Then the cap center that corresponds to N terminus of the second helix is generated such that the connecting string between the two helices is less than 12 Å, and a unit vector for the direction of the second helix is generated such that none of the helices generated so far overlap. The process is repeated for the third and fourth helices. Time steps of 0.1 ps are used in the integration of Eqs. 1 and 2. The WEB dynamics algorithm is applied using 1,200 bins, 5 particles per bin, and at least 1×10^7 Brownian steps. Because the error distribution of particle flux is slightly skewed, the Bootstrap Monte Carlo method (5, 40) is used to calculate confidence interval of the data, assuming independent block size of 50,000 steps. The reported “standard error” is taken to be the half-width of the 68.3% confidence interval.[¶] The calculation stops when the aforementioned half-width is within 10% of the average value.

RESULTS AND DISCUSSION

Given the crudeness of the model, it is quite encouraging to see that the predicted folding times from the simulations, as compiled in Table 1, fall in the range observed for typical proteins. However, the 24 Å bundle folding-time scale is approximately two orders of magnitude slower than the experimentally observed folding time scale of the similarly sized four-helix-bundle bovine acyl-CoA-binding protein (ACBP) (41). The discrepancy might arise from the crudeness of our

Table 1. Simulation results for folding time of a four-helix bundle protein

Helix length, Å	Point-charge folding time, s	Point-dipole folding time, s
24	1.27 ± 0.12	0.665 ± 0.052
32	5.91 ± 0.56	1.50 ± 0.12
40	27.1 ± 2.0	1.74 ± 0.16

The table lists folding time ± 1 SD for helices of different length obtained from simulations using either the point-charge or point-dipole interaction model.

model or from the rather arbitrary definition of folding criteria employed here.

It is also interesting to note the effect of model parameters such as the helix length and the dipole strength. Shorter helices have greater mobility (larger diffusion constants). Also, in the present model helix length is coupled to the force model. For example, the dipole model has a helix–helix interaction energy that varies as the *square* of the length, whereas the dependence of interaction energy on helix length is somewhat weaker in the model with charges at the helix ends. Consequently, the folding time is more sensitive to helix length in the dipole model. In fact, with the helix dipole per length of $0.5e$, the model yields the incorrect result that the helix bundle folding time is proportional to the helix length (data not shown). It is only when the dipole strength per length is reduced to $0.3e$ that reasonable results are obtained.

In general, our computational experiments with two different models for the distribution of charges over the helix microdomains show that the pattern of the charge distribution has a major impact on the folding time. The key point here is not the veracity of a particular model, but that such computational experiments could be performed at all. Most of the simulations were conducted on just four processors of a Silicon Graphics Power-Challenge array at the National Center for Supercomputing Applications. Each entry in Table 1 consumed about 3 weeks of CPU time. Although our CPU consumption seems large, it is estimated that brute-force Brownian dynamics using a comparable number of trajectories would have required about 10,000 processor-years. Furthermore, if the error bars in Table 1 are relaxed to 50%, i.e., order-of-magnitude estimates of the folding time, each WEB-based folding simulation can be completed in a matter of 2 or 3 days.

In future work, an expanded simulation capability permitted by the WEB simulation method could be reinvested in more detailed models. Although a simple microdomain-level model and the simulation results given above are sufficient to illustrate the underlying concepts and algorithmic details, a more realistic protein model must be employed in the simulations to answer relevant kinetics questions, such as folding cooperativity and folding intermediates. In this regard, a more detailed residue-level model of small proteins (18, 42–44) seems to be a good candidate for future simulation studies. Based on our computer usage for this work and the currently available CPU cycles at supercomputer centers, it is believed that the WEB simulations using residue-level models of small proteins, 50–80 residues in length, already can be carried out at the present time.

This work was supported in part by grants from IBM, the Biological Sciences and Technology Program of the Office of Naval Research, and Divisions of Advanced Scientific Computing and Biological Infrastructure of the National Science Foundation. We thank the National Center for Supercomputer Applications and the Maui HPLC Computing Center for computational resources. We also thank Dr. Iasonas Mustakis for his help with calculations of diffusion coefficients.

[§]The implementation of the combine–split algorithm in the present work allocates the same quota of particles to each bin and in this respect differs from the original algorithm described in Huber and Kim (5). Having the same number of particles in each bin facilitates data layout on modern parallel supercomputers.

[¶]For normally distributed data, the 68.3% confidence interval corresponds exactly to ± 1 SD about the average value.

1. Dagget, V. & Levitt, M. (1994) *Curr. Opin. Struct. Biol.* **4**, 291–295.
2. McCammon, J. (1996) *Proc. Natl. Acad. Sci. USA* **93**, 11426–11427.
3. Kolinski, A. & Skolnick, J. (1996) *Lattice Models of Protein Folding, Dynamics and Thermodynamics* (R. G. Landes Company, Austin, TX).
4. Shakhnovich, E. (1997) *Curr. Opin. Struct. Biol.* **7**, 29–40.
5. Huber, G. & Kim, S. (1996) *Biophys. J.* **70**, 97–110.
6. Karplus, M. & Weaver, D. (1976) *Nature (London)* **260**, 404–406.
7. Karplus, M. & Weaver, D. (1979) *Biopolymers* **18**, 1421–1437.
8. Karplus, M. & Weaver, D. (1994) *Protein Sci.* **3**, 650–668.
9. Matthews, C. (1993) *Annu. Rev. Biochem.* **62**, 653–683.
10. Roder, H. & Elöve, G. (1994) in *Mechanisms of Protein Folding*, ed. Pain, R. (Oxford Univ. Press, New York), pp. 26–54.
11. Ballew, R., Sabelko, J. & Gruebele, M. (1996) *Proc. Natl. Acad. Sci. USA* **93**, 5759–5764.
12. Dyson, H., Merutka, G., Waltho, J., Lerner, R. & Wright, P. (1992) *J. Mol. Biol.* **226**, 795–817.
13. Waltho, J., Feher, V., Merutka, G., Dyson, H. & Wright, P. (1993) *Biochemistry* **32**, 6337–6347.
14. Bashford, D., Weaver, D. & Karplus, M. (1984) *J. Biomol. Struct. Dyn.* **1**, 1243–1255.
15. Bashford, D., Cohen, F., Karplus, M. & Weaver, D. (1988) *Proteins Struct. Funct. Genet.* **4**, 221–227.
16. Bashford, D., Karplus, M. & Weaver, D. (1990) in *Protein Folding: Deciphering the Second Half of the Genetic Code*, eds. Gierasch, L. & King, J. (Am. Assoc. Advancement Sci., Washington, DC), pp. 283–290.
17. Yapa, K. & Weaver, D. (1992) *Biophys. J.* **63**, 296–299.
18. Yapa, K. & Weaver, D. (1996) *J. Phys. Chem.* **100**, 2498–2509.
19. Stenkamp, R., Sieker, L. & Jansen, L. (1983) *Acta Cryst. B* **39**, 697–715.
20. Sheriff, S., Hendrickson, W. & Smith, J. (1987) *J. Mol. Biol.* **197**, 273–296.
21. Hecht, M., Richardson, J., Richardson, D. & Ogden, R. (1990) *Science* **249**, 884–891.
22. Hol, W., Van Duijnen, P. & Berendsen, H. (1978) *Nature (London)* **273**, 443–446.
23. Evans, D. (1977) *Mol. Phys.* **34**, 317–325.
24. Evans, D. & Murad, S. (1977) *Mol. Phys.* **34**, 327–331.
25. Allen, M. & Tildesley, D. (1987) *Computer Simulation of Liquids* (Oxford Univ. Press, New York).
26. Gardiner, C. (1990) *Handbook of Stochastic Methods* (Springer, New York), 2nd Ed.
27. Öttinger, H. (1996) *Stochastic Process in Polymeric Fluids* (Springer, New York).
28. Brune, D. & Kim, S. (1994) *Proc. Natl. Acad. Sci. USA* **91**, 2930–2934.
29. Antosiewicz, J. & McCammon, J. (1995) *Biophys. J.* **69**, 57–65.
30. Brune, D. & Kim, S. (1993) *Proc. Natl. Acad. Sci. USA* **90**, 3835–3839.
31. Kim, S. & Karrila, S. (1991) *Microhydrodynamics: Principles and Selected Applications* (Butterworth–Heinemann, Boston).
32. Ermak, D. & McCammon, J. (1978) *J. Chem. Phys.* **69**, 1352–1360.
33. Creighton, T. (1984) *Proteins: Structure and Molecular Properties* (Freeman, New York).
34. Dill, K. (1990) *Biochemistry* **29**, 7133–7155.
35. Fersht, A. (1997) *Curr. Opin. Struct. Biol.* **7**, 3–9.
36. Farkas, L. (1927) *Z. Phys. Chem.* **125**, 236–242.
37. Hänggi, P., Talkner, P. & Borkovec, M. (1990) *Rev. Mod. Phys.* **62**, 251–342.
38. Bryngelson, J. & Wolynes, P. (1987) *Proc. Natl. Acad. Sci. USA* **84**, 4028–4030.
39. Bryngelson, J., Onuchic, J., Socci, N. & Wolynes, P. (1995) *Proteins Struct. Funct. Genet.* **21**, 167–195.
40. Press, W., Teukolsky, S., Vetterling, W. & Flannery, B. (1992) *Numerical Recipes in C* (Cambridge Univ. Press, New York), 2nd Ed.
41. Kragelund, B., Hojrup, P., Jensen, M., Schjerling, C., Juul, E., Knudsen, J. & Poulsen, F. (1996) *J. Mol. Biol.* **256**, 187–200.
42. Rey, A. & Skolnick, J. (1991) *Chemical Phys.* **158**, 119–219.
43. McCammon, J., Northrup, S., Karplus, M. & Levy, R. (1980) *Biopolymers* **19**, 2033–2045.
44. Subramaniam, S., Tcheng, D. & Fenton, J. (1996) in *Proceedings of the Fourth International Conference on Intelligent Systems in Molecular Biology*, eds. States, D., Agarwal, P., Gassterland, T., Hunter, L. & Smith, R. (AAAI Press, Menlo Park, CA), pp. 218–229.