Assessing and Improving Computational Models of Protein Thermodynamics and Kinetics

Elizabeth Kellogg

A dissertation

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2012

Reading Committee:

David Baker, Chair

Barry Stoddard

Philip Bradley

Program Authorized to Offer Degree:

Biochemistry

University of Washington

Abstract

Assessing and Improving Computational Models of Protein Thermodynamics and Kinetics

Elizabeth Kellogg

Chair of the Supervisory Committee:
Professor David Baker
Biochemistry

The purpose of this thesis is to rigorously assess and improve computational models of protein thermodynamics and kinetics. The first part consists of computational ΔΔG prediction; we explore the performance of protocols which sample an increasing diversity of conformations and examine their abilities to recapitulate both changes in free-energy as well as changes in structure. Application of the improved ΔΔG prediction protocol yields high performance on independent benchmarks as well as success in two blind applications.

The second portion consists of assessing and improving discrete computational models of protein kinetics. The space accessed by a folding macromolecule is vast, and how to best project computer simulations of protein folding trajectories into an interpretable sequence of discrete states is an open research problem. There are numerous alternative ways of associating individual configurations into collective states, and in deciding on the number of such clustered

states there is a trade-off between human interpretability (smaller number of states) and accuracy of representation (larger number of states). Here we introduce measure for assessing alternative discrete state models of protein folding and assess different methods of defining discrete states. Using the most predictive representation to study the folding transitions of the WW domain in very long molecular dynamics simulations we identify new states and transitions. The methods developed here should be generally useful for investigating the thermodynamics and kinetics of protein structure.

Acknowledgements

I would like to thank everyone who has been instrumental to the completion of my PhD. In particular, I'd like to thank my PhD advisor David Baker, who is an unfailing optimist, always ready to find the silver-lining in any cloud. His support and encouragement helped me to continue whenever I was discouraged. I would like to thank the Baker lab, which was an intellectually stimulating environment of diverse backgrounds and interests and has been instrumental to my development as a scientist. I would like to thank my family, first my husband Alexandre Zanghellini who has never failed to support me at my worst and celebrate at my best. Last but not least, I want to thank my dad who energetically and enthusiastically supported my studies in science, and who died shortly before the completion of this work.

Dedication

I dedicate this thesis to my father, Christopher Cameron Kellogg. He instilled in me at a young age the importance of hard work and responsibility as well as curiosity and wonder for the world around me. When I was a child, he would conduct many backyard scientific experiments to demonstrate natural phenomena in an intuitive way. He never belittled my ideas, however silly, and always encouraged me to reach my full potential, in whatever path I might choose.

Chapter One: Computational ΔΔG Prediction

ΔΔG Introduction

Accurate modeling of the impact of a mutation in a protein must recapitulate both the structural change associated with a mutation as well as the change in the free energy of the folded state. As with most other macromolecular structure prediction problems(*1*), accurately predicting the structural changes associated with a point mutation requires, first, an efficient method for conformational sampling, and second, an accurate energy function. Once the structure of the mutant protein has been computed, the change in the free energy of folding can be estimated from the difference in the free energies of the folded wild type and mutant structures, assuming the change in the unfolded state free energies depends only on the identities of the amino acids at the substituted positions. Previous studies have used conservative sampling procedures to predict differences in free-energies, ΔΔGs, allowing only the mutated residue to reconfigure within a fixed environment(*2-4*), as well as methods incorporating increased protein flexibility(*5, 6*). Although the above studies all report impressive correlations with experimental values, they employ quite different energy functions and sampling strategies, hence it is not clear which features of the approaches are sufficient and necessary for good performance.

Prompted by a recent study reporting poor performance of the Rosetta methodology in predicting the free energy changes associated with mutations (*7*), we present here a detailed analysis of the tradeoff between the resolution of the energy function and the extent of conformational sampling in ΔΔG prediction. We go beyond previous work by systematically evaluating a wide range of sampling methodologies in the context of the same forcefield (Figure

1), separating the contribution of the forcefield from that of the sampling methodology. We show that roughly equivalent overall performance can be achieved using a wide range of sampling techniques, ranging from an entirely fixed backbone approximation to full-protein flexibility, provided that the resolution of the energy function is matched to the granularity of the sampling technique. The poor results obtained by Potapov *et al.* are shown to be the result of inappropriate combination of limited sampling with an undamped potential function. We attempt to refit scorefunction weights for $\Delta\Delta G$ prediction and discover that prediction accuracy is not significantly enhanced. By studying the distributions of prediction failures, we identify areas of modeling which need to be improved for higher accuracy prediction of the changes in stability and structure brought about by point mutations.

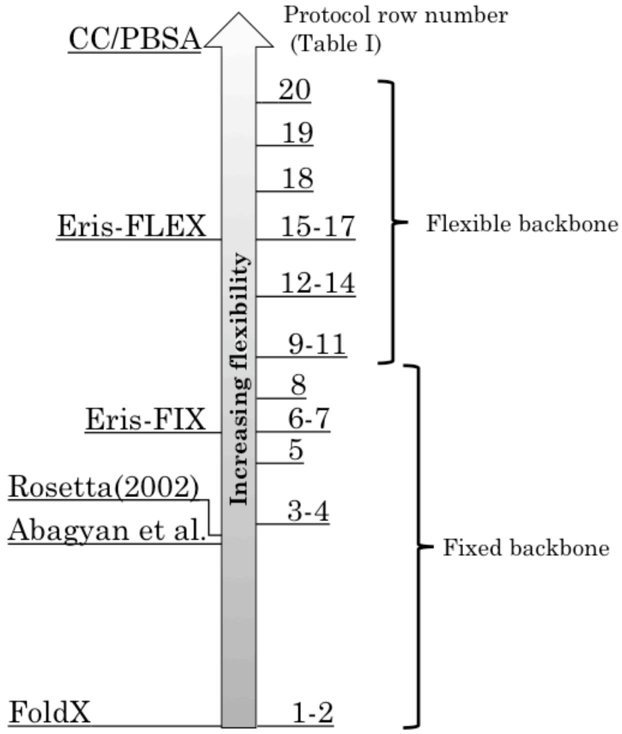


Figure 1: Extent of conformational sampling in the $\Delta\Delta G$ protocols considered here. Protocols considered here are on the right, and previously described methods (refs 1-5) on the left.

ΔΔG Methods

Data-set

Except for comparison to the results of Potapov et al. mentioned in the discussion, all tests reported in this paper utilized a benchmark set comprised of 1,210 single mutations obtained from Protherm(*8*). Duplicate entries were resolved by taking the highest resolution structure, and if multiple experimental measurements were recorded, the mean of all reported measurements was used. Structures greater than 350 residues were eliminated due to the computational intensiveness of some of the protocols tested. A representative set of 771 mutations was used to assess the most computationally intensive protocols, including the proteins barnase (1a2p), apomyoglobin (1bvc), FK506 binding protein (1fkj), staphylococcal nuclease (1stn), α-spectrin(1u5p), chymotrypsin inhibitor II (2ci2), and T4-lysozyme (2lzm).

Sampling Protocols (Table I):

The first set of protocols we considered relax the sidechains but keep the backbone fixed. Sidechains are optimized in two steps—first, discrete combinatorial rotamer optimization and second, continuous optimization of the sidechain torsion angles. The combinatorial rotamer optimization (referred to as *repacking* throughout the remainder of the text) is carried out using Monte Carlo simulated annealing with the Dunbrack backbone dependent rotamer library(*9*). The continuous optimization is carried out using quasi-Newton minimization and is referred to as *minimization* throughout the remainder of the text.

We experimented with two energy functions at both the repacking and minimization steps. The first is the standard Rosetta all atom energy function used in prediction and design calculations (*10*); we refer to this as "hard-rep" because the Lennard-Jones repulsive interactions are not damped, thus atomic clashes incur very large energetic penalties. The second has the repulsive interactions at short atomic separations damped as described but is otherwise identical; we refer to this as “soft-rep” because small atomic overlaps are not heavily penalized.

We also experimented with allowing different numbers of residues surrounding the site of mutation to be repacked. As indicated in the Table I protocol summary, we considered three possibilities: first, only repacking the mutated residue, second, only residues within 8Å of the mutated residue, and third, all residues.

We also explored protocols which carry out backbone torsion angle minimization following sidechain repacking in attempts to more accurately model the structural consequences of mutations. To prevent the backbone from moving too much from the native structure, in some protocols we included distance constraints during the backbone minimization.

Extensive backbone modification

Finally, we explored protocols which more extensively search through alternative backbone conformations. We developed a Monte Carlo simulated annealing protocol that generates backbone conformations with ideal bond lengths and bond angles that uniformly sample the space of conformations surrounding any given native structure. The protocol carries out 100,000 moves each consisting of a small random perturbation of the backbone torsion angles; the scoring function prevents sampling from deviating by more than a specified tolerance from the starting structure. Single side chain rotamer flips are attempted at one-tenth the frequency of backbone moves. The resulting structures have small and partially compensating changes in nearly all the backbone torsion angles (Figure 2). The lowest energy structure sampled during each trajectory is subjected to backbone and sidechain minimization using the hard-rep energy function. Because the resulting structures are sampled stochastically, increasing the ensemble size improves $\Delta\Delta G$ prediction and is required to obtain a converged estimate; we found that approximately 50 optimized structures are sufficient for a converged ΔΔG prediction (Figure 3).

The input structure is either the minimized wildtype crystal structure or a mutated structure produced by the single-residue sidechain-repacking protocol, (*row 2, Table I)*. The functional form of the Cα constraints is a square-well harmonic with boundaries at 0.5 Å, placed on any pair of Cα atoms that lie within 9 Å of each other. The functional form of the square-well harmonic is 0 within 0.5Å of the starting distance, deviations further than this are penalized according to a harmonic function with sigma= 0.1 (where the constraint penalty is proportional to deviation/sigma). Monte-carlo sampling is performed for 1,000 steps at constant temperature, first at a high temperature of 10 to gain structural diversity, then at a lower temperature of 2.5 so as to anneal back to the input conformation. At each step the number of backbone or sidechain moves is equal to a 1/4th the number of residues in the protein. Backbone moves are carried out 90% of the time and sidechain moves 10%.

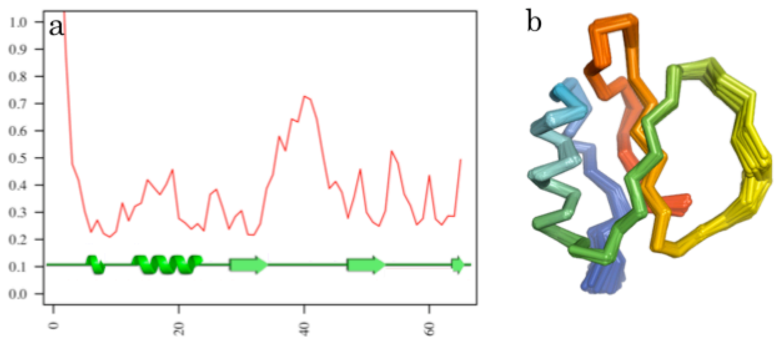


Figure 2. The variation in models produced using the Monte Carlo ensemble method (see methods) show variation across the length of the protein. a) Root-mean-squared-deviation for each position in chymotrypsin inhibitor II (2ci2) b) the ensemble of models.

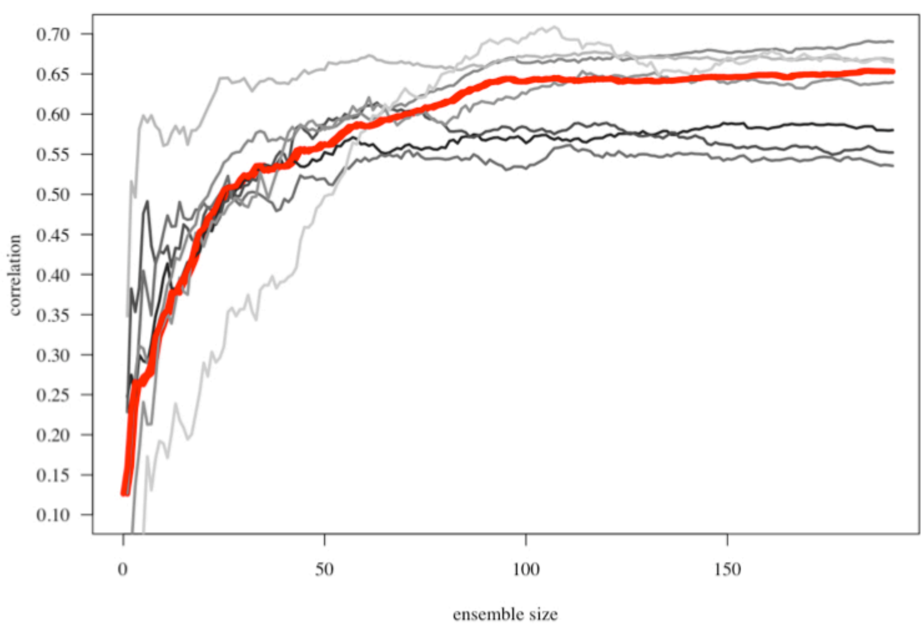


Figure 3 Increasing ensemble size in the Monte Carlo ensemble method improves performance. Red represents all mutations in set. Results for each protein in the set is colored gray.

Sidechain moves consist of substituting all rotamers in the library at that position and selecting the minimum energy rotamer. Backbone moves are either applied through random changes in ϕ and ψ angles up to 0.2 degrees or shear moves, which randomly perturb the ϕ of the current and preceding residues in opposite directions by up to 1.5 degrees so as to reduce the perturbation to the global topology of the protein scaffold. The magnitude of the perturbation depends on the secondary structure element the residue resides in: loops allow the most freedom of movement while helix and sheet allows the least freedom. After a backbone move is made, a Ramachandran check is performed. During the low temperature phase, ϕ-ψ angle perturbations are lowered to 0.1 degrees. Moves are accepted according to the metropolis criterion probability with kT set to 2 kcal/mol(*11*). After the monte-carlo trajectory is finished, minimization is performed on the lowest energy structure from that trajectory. In order to ensure convergence, 200 independent trajectories are carried out, and the predicted $\Delta\Delta G$ is defined as the difference in mean energies of the mutant and wildtype ensembles .

Flexible Backbone Protocols

Harmonic constraints are applied to every pair of Cα atoms within 9 Å (penalty = $(d-d_0)/s$; d_0 being the WT distance & s = 0.5), and quasi-Newton minimization was then performed with respect to the sidechain and backbone torsion angles three times, ramping up the weight on the repulsive component of the Lennard-Jones potential each time.

Energy function

The Rosetta high-resolution (*hard-rep*), undamped and soft-repulsive (*soft-rep*), damped energy function are described in detail in previous publications(*9, 11-13*); discussion of the two energy functions here will focus on the differences in definition of the Lennard-Jones potential. Both Lennard-Jones terms have a linear form at short distances with a slope equivalent to the slope of the Lennard-Jones potential at the switch-point, which depends on the ratio d_{ij}/s_{ij}, where d_{ij} is the distance between atoms i and j, and s_{ij} is defined as the sum of the van der Waals radii. The hard-rep energy-function switch-point is at $d_{ij}/s_{ij} = 0.6$, and the soft-repulsive $d_{ij}/s_{ij} = 0.91$. At longer distances, the potential has the standard form:

$$E_{vdw} = \varepsilon_{ij}\left(\left[\frac{\sigma_{ij}}{d_{ij}}\right]^{12} - 2\left[\frac{\sigma_{ij}}{d_{ij}}\right]^{6}\right) \text{ if } \frac{d_{ij}}{\sigma_{ij}} \geq cutoff$$

where e is defined as the geometric mean of the homoatom well-depths, e_{ii} and e_{jj}, which are taken from CHARMM. Furthermore, the atomic radii of aliphatic atoms in the soft-repulsive energy function are scaled a factor of 1.07 relative to those used in the hard-rep scoring function. The long-range interaction cut-off is set to 9.0 Å, with splines smoothing the potential to zero between 8.5 and 9 Å.

The long-range interaction cut-off, which affects the Lennard-Jones potential as well as the implicit solvation potential, is extended from Rosetta's default value of 6 Å to 9 Å, which we found improved ΔΔG predictions from correlation coefficients of approximately 0.45 to 0.66 (*repack a single-residue with soft-rep, row 1 of Table I*) and 0.60 to 0.69 (*repack all residues with soft-rep followed by minimization with hard-rep, row 16 of Table I*) on the full set of 1,210.

Optimization of weights

The energy, E(S), of a structure S is modeled as the weighted sum of the individual energy terms and the amino-acid reference energies:

$$E(S) = \sum_{i}^{|terms|} w_i E_i(S) + \sum_{i}^{|S|} ref_{aa_i}$$

where ref_{aai} denotes the reference energy for the amino acid at the i^{th} position in the structure. In this model, the weights and reference-energies are tuned to minimize the square error in the prediction of a subset of 968 ΔΔG's. That is, given a set of predicted structures for the mutant sequence and wildtype sequence for a single observed ΔΔG, *j*, we define an individual fitness function, F_j

$$F_j = \left(\Delta\Delta G_j^{obs} - \min_k(E(S_k^{mut}) + \min_k(E(S_k^{wt}))\right)^2$$

where the lowest energy structure depends on the assigned weights. We further define a total fitness function $F = \sum_j F_k$. The OptE module of Rosetta minimizes this fitness function *F* with respect to some or all of the weights (the "free" weights) and all the twenty amino-acid reference energies, using a combination of particle-swarm minimization(*14*) and quasi-Newton minimization.

The weight sets that result from optimizing this square-error fitness function yield the maximal Pearson correlation coefficient when the best-fit line is restricted to having an intercept of 0 and a slope of 1. To fit with a slope other than 1, we define an extra scaling factor, *s*, to adjust weights that are otherwise considered constant. The model for the energy of a structure becomes

$$E(S) = s\sum_{i}^{|terms|} w_i E_i(S) + \sum_{i}^{|S|} ref_{aa_i}$$

OptE computes derivatives with respect to *s* and optimizes *s* alongside those weights and reference energies that are free. The predicted structures for the wildtype and mutant sequences were generated using the best performing protocol using limited backbone minimization*(row 16 of Table I)* model-generation protocol described earlier. The minimum energy models generated by this protocol for each sequence were included in the weight-fitting optimization. 5-fold cross-validation was performed; reported correlations are based on the combined predicted values from applying optimized weights to the test sets. *Rosetta revision 33991 was used for weight-optimization. The command-lines for performing the described procedures are in the Appendix.*

Analysis Methods

In order to facilitate analysis, mutations were categorized according to features which were hypothesized to be correlated with prediction accuracy.

Mutant Categories

Mutations were categorized into different categories, defined as follows:

I. Bfactor: Residues with greater than 0.86 normalized average B-factor were considered to be in high B-factor regions, those falling below the cutoff are low B-factor regions. The cutoff is chosen to ensure equivalently sized categories.

II. Burial: Residues were classified according to the fractional solvent-exposed surface area of the wild-type residue. Residues with < 10% are buried, > 40% are exposed, and >= 10% or <= 40% as partially exposed.

III. Chemical: Non-polar mutations are those involving only amino acids A,F,V,L,W,I,M. Polar mutations are only those involving polar or charged amino acids: D,E,S,Y,C,H,Q,R,T,N,K.

Fraction correct

Experimental values of mutations were divided into three categories: stabilizing (<= -1 kcal/mol), neutral (> -1 kcal/mol and < 1kcal/mol), and destabilizing(>= 1kcal/mol). The fraction correct is defined as the number of mutations categorized correctly divided by the total number of mutations in the benchmark set. The fraction correct is displayed, along with the pearson correlation coefficient, for all protocols sampled in Table I.

Structural Analysis

The predicted mutant structure is the minimum energy structure produced with any of the algorithms studied. The mutant structure is then optimally superimposed using all Cα atoms on the mutant crystal structure, and the all-atom rmsd of the mutant residue is computed.

	extent of sidechain repacking	energy function used in repacking	extent of minimization	constraints used during minimization	energy function used in minimization	performance		
						all mutations	large-to-small mutations	small-to-large mutations
1	1 residue	soft-rep				0.66 / 0.73	0.67 / 0.73	0.55 / 0.65
2		hard-rep				0.02 / 0.53	0.25 / 0.57	0.10 / 0.52
3	within 8 Å	soft-rep				0.68 / 0.72	0.68 / 0.73	0.56 / 0.64
4		hard-rep				0.04 / 0.54	0.46 / 0.64	0.14 / 0.53
5		hard-rep	sidechain	no	hard-rep	0.26 / 0.55	0.50 / 0.66	0.38 / 0.55
6	all-residues	soft-rep				0.67 / 0.71	0.67 / 0.72	0.57 / 0.64
7		hard-rep				0.10 / 0.54	0.46 / 0.63	0.22 / 0.55
8		hard-rep	sidechain	no	hard-rep	0.25 / 0.55	0.58 / 0.72	0.32 / 0.54
9	1 residue	soft-rep	backbone and sidechain	uniform	soft-rep	0.65 / 0.69	0.63 / 0.70	0.63 / 0.66
10		soft-rep			hard-rep	0.51 / 0.69	0.65 / 0.72	0.27 / 0.55
11		hard-rep			hard-rep	0.58 / 0.72	0.66 / 0.73	0.40 / 0.58
12	within 8 Å	soft-rep			soft-rep	0.65 / 0.70	0.62 / 0.71	0.61 / 0.66
13		soft-rep			hard-rep	0.66 / 0.72	0.65 / 0.72	0.58 / 0.65
14		hard-rep			hard-rep	0.64 / 0.73	0.65 / 0.74	0.50 / 0.63
15	all-residues	soft-rep			soft-rep	0.57 / 0.69	0.53 / 0.68	0.65 / 0.67
16		soft-rep			hard-rep	0.69 / 0.72	0.67 / 0.72	0.66 / 0.67
17		hard-rep			hard-rep	0.63 / 0.73	0.67 / 0.74	0.45 / 0.62
18		soft-rep		position-specific	hard-rep	0.67 / 0.70	0.64 / 0.71	0.64 / 0.67
19		soft-rep		no	hard-rep	0.63 / 0.69	0.60 / 0.69	0.61 / 0.67
20	all-residues	Monte Carlo ensemble generation (see methods)				0.65 / 0.68*	0.66 / 0.68*	0.54 / 0.73*

Table I ΔΔG prediction accuracies for all tested protocols. Values reported for each method are the correlation/stability-classification accuracy with respect to experimental data. While overall correlations are very similar among different methods, the small-to-large class shows improved performance with the addition of more protein flexibility. *Values correspond to a reduced set of 771 mutations.

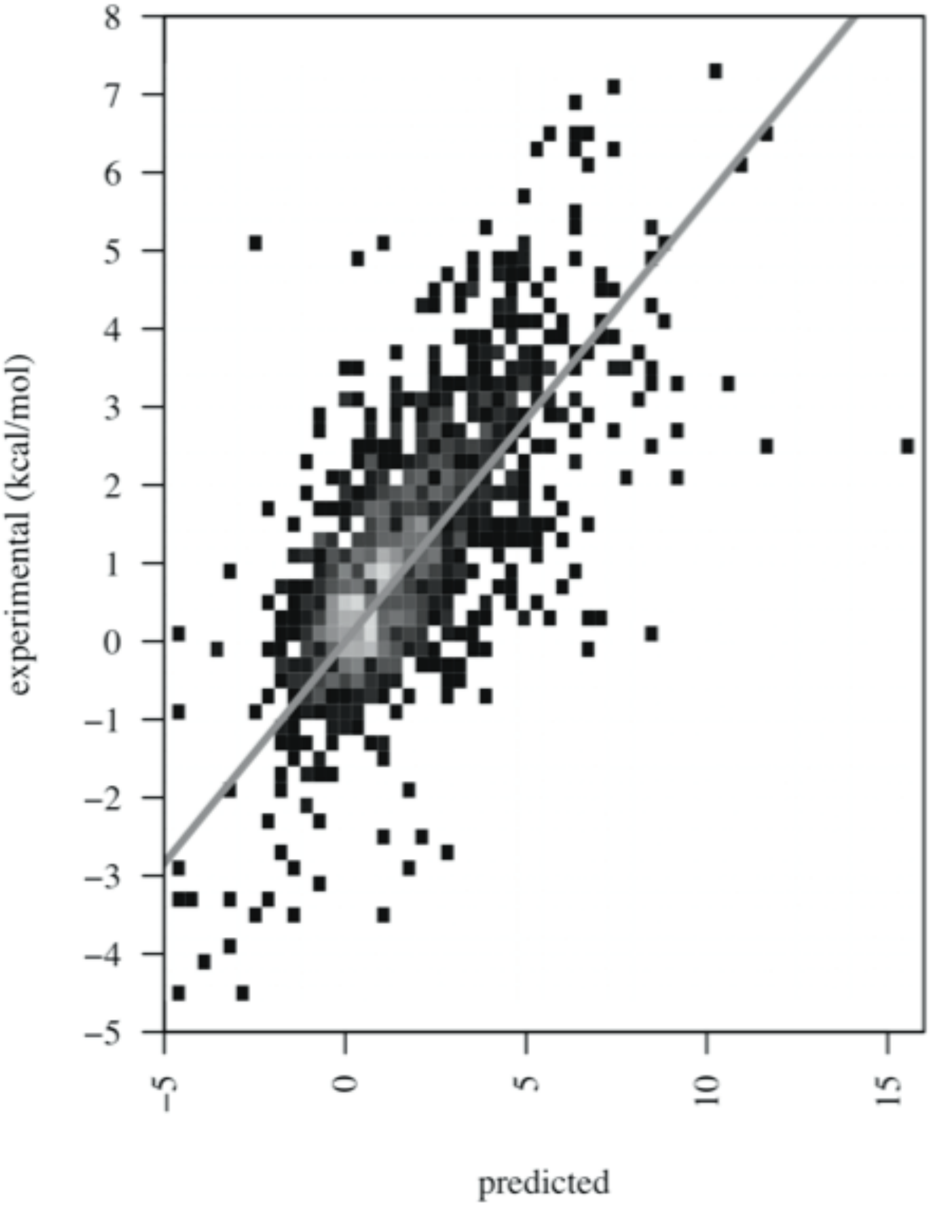


Figure 4 The best performing method involves backbone minimization after repacking all sidechains. The correlation is 0.69 on the full set of 1,210 mutations. Predicted values along the x-axis versus experimental values (kcal/mol) on the y-axis. The equation of the best-fit line is y = 0.57x. Symbols are colored according to the counts falling within the bin; lighter colors indicate more observations.

Results

As described in detail in the Methods section, we experimented with a range of different protocols for computing free energy changes accompanying mutations. In all of these protocols, the calculations focus on energy changes in the native state—changes in free energy of the unfolded state are assumed to be context independent for computational tractability. We find that the best performing protocol involves limited backbone minimization and has a pearson correlation coefficient of 0.69 over a set of 1,210 mutations (Figure 4) with a best-fit line of y = 0.57 x.

Sidechain-only optimization

In the first set of protocols the sidechains but not the backbone are allowed to relax following introduction of the amino acid sequence change. Several trends are evident in the comparison of the performance of the different fixed backbone protocols in Table I. First, the performance of protocols using the hard-rep potential improved with increasing conformation freedom (rows 2, 4, and 7,Table I), but for all of the fixed backbone sampling strategies better performance was achieved with the soft-rep potential (rows 1 & 2, rows 3 & 4, rows 6 & 7, Table I). Atomic clashes in models of the mutant structures that cannot be fully resolved with sidechain-only optimization are likely to account for both of these trends; indeed, filtering the data by removal of ΔΔGs with large clashes (> 7 E_{rep}) after repacking all residues with the hard-rep energy function (row 7, Table I) increased the correlation to 0.62 (1,117 mutations) from 0.10 (1,210 mutations). The atomic clashes that remained following sidechain repacking were not resolvable with sidechain minimization (r = 0.25)(row 8, Table I). Second, the performance of protocols using the soft-rep energy function was insensitive to the amount of conformational freedom (rows 1, 3, and 6, Table I); very similar performance was obtained whether only the mutated residue was repacked, a subset of residues were repacked or all residues were repacked (correlations of 0.67 and stability-classification accuracies of 0.73; rows 1, 3, and 6, Table I). This result is consistent with the earlier observation(*2*) that the soft-rep potential is well suited to recapitulating ΔΔGs with a fixed backbone, and does not require the optimization of neighboring sidechains to obtain a significant correlation.

The reason for the poor results obtained by Potapov *et al* is evident from the above analysis. Perhaps because of unclear documentation, Potapov *et al* used the hard-rep potential with a limited sidechain repacking protocol followed by sidechain minimization (similar to *row 5 in Table I*), and found very little correlation between predicted and observed ΔΔGs (0.26 on 1,913 mutations)(*7*). As we have discussed previously(*12*) , if a fixed backbone representation with discrete rotamer optimization is carried out, the repulsive interactions must be damped, otherwise they dominate the computed energies.

Limited backbone minimization

The set of sidechain-and-backbone protocols (rows 9-19 in Table1), extends the set of sidechain-only protocols by applying a restrained quasi-Newton minimization step to backbone and sidechain degrees of freedom starting from sidechain optimized structures while tethering the structure to the initial starting model (see methods: Extensive backbone modification). Correlations were higher when the soft-rep energy function was used during the sidechain optimization step than when the hard-rep energy function was used (R = 0.63 with hard-rep, 0.69 with soft-rep when backbone and sidechain minimization is performed after repacking all-residues, see rows 16 and 17, Table I). However, if the soft-rep energy function was used during minimization, increased conformational freedom yielded worse correlations (row 15, Table I, R = 0.57) highlighting the incompatibility of the soft-rep energy function with flexible backbone modeling.

The performance of these protocols improved as more conformational freedom was introduced to the system – to a point. As more sidechains were allowed to repack before the backbone minimization step, the correlation improved (rows 10, 13, and 16, Table I). Repacking all-residues before constrained minimization (r = 0.69, row 16, Table I) performed marginally better than repacking residues within 8 Å(r = 0.66, row 13, Table I) which performed better than repacking the mutant residue only (0.51, row 10, Table I). This improvement is likely due to sensitivity of the hard-rep scoring function to residual atomic clashes; excluding mutations for which models contained high repulsive energies (> 7 E_{rep}) restored the correlations of the protocols to 0.68 (1,207 mutations).

The minimization step in the above calculation is constrained using crystal structure derived restraints. Allowing increased conformational freedom in the neighborhood of the mutation during the minimization stage (weakening the crystal structure based distance restraints in the neighborhood of the mutation site) yielded a slightly worse correlation (from r = 0.69, row 16, Table I to r = 0.67, row 18, Table I), and complete removal of restraints during minimization (row 19, Table I), yielded a worse correlation still (0.63, Table I).

Extensive Backbone Optimization

Sampling is quite limited with quasi-Newton minimization of the backbone; it locates the nearest minimum but is unable to cross barriers into lower-energy minima nearby. To increase the exploration of the energy landscape close to the native structure we developed a protocol that generates an ensemble of structures centered on the native structure. Other methods have recently been developed using "back-rub" motions that have proven quite powerful(*15, 16*). Our goal was to generate ensembles with levels of structural perturbation similar to those generated with back rub while restricting bond lengths and angles to ideal values (see Methods section: "Monte-Carlo Ensembles"), since the addition of bond length and bond angle degrees of freedom and associated potential terms can introduce noise. The new protocol was tested with the hard-rep energy function and yields ensembles with uniform deviations from the starting native structure both in Cartesian coordinates and in the individual torsion angles (see figure 2).

Although significant correlations can be produced with stochastic sampling of backbone conformations close to the starting structure, these correlations are not as high as those obtained using limited backbone minimization (r=0.65, row 20 in Table I vs. r=0.69, row 16 Table I , both evaluated on a set of 771 mutations). As previously observed by Benedix et al(*6*), the correlations increase as more models in the ensemble are produced (figure 3). This is likely due to reduction in the noise associated with stochastic sampling of the protein backbone. The considerable improvement obtained by Benedix *et al* with conformational sampling compared to using static crystal structures likely reflects the undamped potential they used.

Comparison of sampling techniques

No one protocol significantly outperforms the others; among the best combinations of energy function and optimization method for each of the sampling regimes, the correlations ranged from 0.65-0.69 (rows 6 and 16, Table I). However, if mutations are divided according to the change in van der Waals volume, clear trends are observed. In particular, the best protocol that relaxed the backbone (row 16, Table I) showed a significant improvement over the best sidechain-only protocol (row 6, Table I) for the small-to-large class of mutations (r = 0.66 versus r = 0.57 on a set of 164 mutations, rows 6 and 16, Table I) and also on mutations involving only hydrophobic residues (r=0.68 versus r=0.57 on a set of 365 mutations)(*17*).

The inclusion of restrained backbone minimization (row 16, Table I) did not compromise the correlation on large-to-small mutations; the correlation is equivalent to the maximum obtained by other methods, 0.67 (row 16, Table I). A similar result was reported by Yin et al.(*5*). The protocol involving extensive backbone movement (row 20, Table I) has correlations similar to the fixed backbone methods in all size categories--improvements in modeling mutations that induce significant backbone changes are offset by the introduction of noise in modeling the remaining mutations. The stability-classification accuracies for the best methods were 0.73 for large-to-small mutations (rows 1, 3, and 11, Table I) and 0.67 for small-to-large mutations (rows 15, 16, 18, and 19, Table I); no protocol significantly outperforms the others using this metric.

Structure recapitulation

Overall, the variation in the protein backbones produced by the methods increases with increasing conformational searching. Constrained minimization protocols (rows 9-17, Table I) on average produce structures 0.08 Cα rmsd from the starting structure, whereas minimization with no constraints (row 19, Table I), produces structures on average 0.57 Cα rmsd from the starting structure. The Monte-Carlo ensemble method (row 20, Table I) is more aggressive than limited backbone flexibility but somewhat constrained compared to free-minimization, producing backbones of 0.44 Cα rmsd on average.

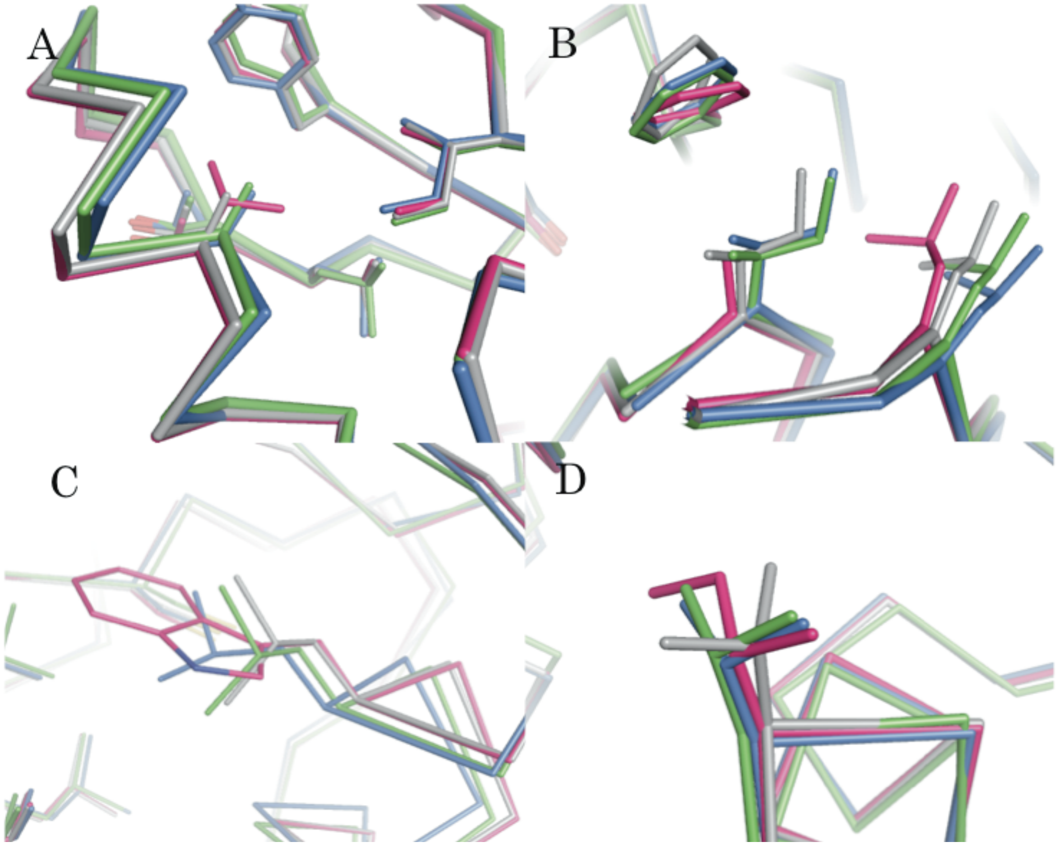


Figure 5 Examples for which modeling backbone flexibility improves structural recapitulation. (a)T4-lysozyme mutant (1qtb), V 42 A (b)T4-lysozyme mutant (241l) A 29 I (c)FK506 binding protein (1fkj) W 59 L(d) T4-lysozyme (2lzm) I 3 V. Pink, starting wild-type crystal structure; blue, mutant crystal structure; gray, structural prediction with limited backbone minimization; green, structure produced with less stringent constraints around the site of mutation and uniform harmonic constraints outside this region (row 18, Table I). In (d), green is the structure produced from perturbed backbone protocol (row 20, Table I).

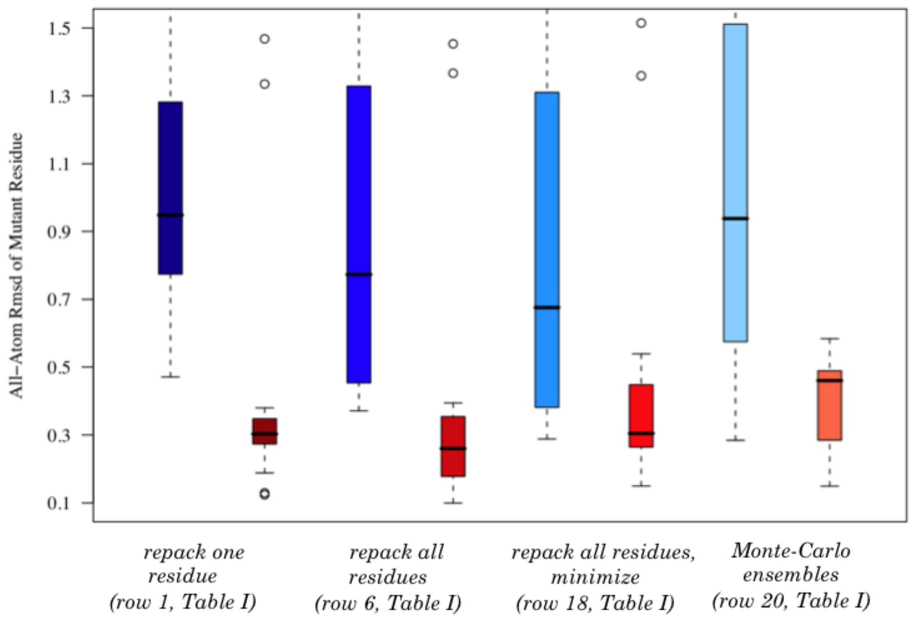


Fig 6. Incorporation of flexibility improves structural predictions when the local backbone conformation changes significantly. Blue indicates mutations for which the local-backbone conformational change is >= 0.4 Cα distance. Red indicates mutations for which backbone change is below this threshold. Only buried, small-to-large mutations are included.

We evaluated the performance of the different protocols described above in recapitulating structural changes accompanying point mutations observed in crystal structures of mutant proteins, using a set of 154 pairs for which the crystal structures of both wild type and mutant proteins were available (see Appendix Table I). The more aggressive flexible backbone methods produced quite striking recapitulations of structural changes in a number of cases (Figure 5), but overall did not result in improved predictions over the more conservative methods (Figure 6). Overall, loosening constraints around the site of mutation yielded better predictions than the uniform constraint minimization method in 62 cases (of 154) whereas the more aggressive backbone perturbation method yielded better predictions than the best limited-backbone minimization protocol in 44 cases, as assessed by comparing the all-atom rmsd of the mutant sidechain to the crystal structure (Figure 6). Prediction accuracy for small-to-large, buried mutations increases slightly with increasing structural variability, but only when the backbone is known to shift (>= 0.4 Å Cα shift). When the backbone is essentially correct to begin with, the all-atom rmsd prediction accuracy for the flexible backbone methods is not surprisingly worse than for the fixed backbone methods (Figure 6). The failure of the flexible backbone methods to give an overall improvement reflects in part the large fraction of cases where very little backbone movement actually occurs. This lack of consistent improvement in structural recapitulation also in part explains why the flexible backbone methods do not do better overall in ΔΔG prediction.

ΔΔG prediction performance with empirical structural knowledge

To determine if improved structural models necessarily lead to improved energetic predictions, we computed predicted ΔΔGs based on the solved crystal structures (data not shown). Not surprisingly, naively taking the difference in total computed energy between the wild type and mutant crystal structures resulted in zero correlation with the experimental ΔΔG data, since small differences throughout the independently solved structures drown out the energy differences due the sequence change itself. To reduce this noise, we computed the difference not in the total energies of the wild type and mutant crystal structures, but of the total interaction energies of residues at the mutation site. The correlation of this computed interaction energy difference with the experimental ΔΔG data, 0.77, is the same as that of the best limited-backbone minimization protocol over this set of mutations, a finding corroborated by other studies(*4*).

Energy function training incorporating both ΔΔG and sequence recovery data

The Rosetta energy function contains "reference energies" for each of the 20 amino acids, which represent the average energy of the residue in the unfolded state. The parameters in the standard energy function used in the calculations described thus far in this paper were determined by maximizing sequence recovery in comprehensive sequence design calculations for a large set of proteins(*9*). In this weight optimization, the reference energies are influenced by the overall frequencies of the amino acids, and hence will also incorporate effects related to the metabolic cost of making amino acids, their effects on solubility, etc. Hence, we reasoned that better performance might be achieved if these reference energies were fit directly on ΔΔG data where overall amino acid composition biases are absent. We fit the 20 reference energies, using 20-fold cross-validation, keeping all other weights fixed except for a constant term in order to adjust the energies to a kcal/mol scale, obtaining an overall correlation of 0.73 (Table II). Optimization of weights on other forcefield terms did not improve the correlation sufficiently to be justified (Table II). While the increase in performance resulting from fitting on ΔΔGs was not large, a notable advantage is that this puts the overall energy function on a kcal/mol scale matched to experimental ΔΔG measurements.

	sequence recovery (%)	$\Delta\Delta G$ correlation
optimize for $\Delta\Delta G$ recovery	20	0.71
optimize for $\Delta\Delta G$ recovery, only reference weights	23	0.73
optimize for sequence recovery	34	0.46
optimize for both sequence and $\Delta\Delta G$ recovery	29	0.69
original Rosetta high-resolution energy function weights	26	0.69

Table II Optimizing weights towards ΔΔG recapitulation produces a maximal correlation of 0.73.

For design calculations, reference energies trained on sequence recovery are likely to be desirable, whereas for ΔΔG calculations, training on thermodynamic data is more appropriate. To obtain a compromise reference weight set, we trained on both datasets at the same time using the opt-E weight-optimization suite (Leaver-Fay *et al.*, manuscript submitted), yielding a weight set with a ΔΔG correlation 0.69 and a sequence recovery rate of 29% (Table II)(parameters in Appendix).

The correlation after weight-training on ΔΔG experimental data, r=0.73, is essentially equivalent to correlations obtained by other algorithms, ranging from 0.59-0.76. Why do such widely different conformational sampling protocols and energy functions have such similar prediction accuracies? A likely explanation is that the remaining variance in the experimental data is due to factors not represented in any of the models. The first of these is experimental error in the measurements themselves—it was recently estimated based on differences in the free energy changes determined in different groups for the same mutation that the maximum correlation possible is 0.86(*7*). The second missing contribution is likely due to errors/missing features in the energy function. We survey these potential missing contributions in the following paragraphs.

Contributions to failures in prediction accuracy

To investigate potential systematic problems, mutations were categorized according to polarity, burial, and B-factor (see Category definitions in Methods Section) (Table III). We also compared the enrichment of specific structural features in mutations systematically mispredicted by all of our methods (Table IV). The outlier set is defined as the consensus of 10% worst predictions for all protocols; removal of these outliers improved correlations (r = 0.71-0.75). Features we examined included the unfolded state, hydrogen bond characteristics, as well as interactions with buried, bound water molecules or ligands.

category	correlation	fraction correct	# mutations
all	0.69	0.72	1210
low B-factor	0.69	0.75	596
high B-factor	0.67	0.7	606
buried	0.66	0.78	397
partially-exposed	0.63	0.71	421
exposed	0.54	0.72	384
non-polar	0.68	0.76	365
polar-to-nonpolar	0.58	0.68	456
polar	0.79	0.7	81

Table III Prediction accuracy for different classes of mutations. Exposed mutations and polarity changes are relatively poorly predicted. Results shown are for the best performing method, involving limited backbone minimization after repacking all sidechains.

category	outlier mutations			all mutations		
	number	total	percentage	number	total	percentage
Unfolded state significantly affected by mutation	13	38	34	23	305	8
Buried hydrogen bonds	16	85	19	106	1210	9
Buried polar-polar hydrogen bonds	11	85	13	67	1210	6
Buried charged-polar hydrogen bonds	7	85	8	59	1210	5
Introduction of buried unsatisfied hydrogen bonding partner	7	85	8	52	1210	4
Putative conformational change	6	85	7	34	1210	3
Buried, hydrogen bonded to water	8	85	9	45	1210	4
Ligand contacts	5	85	6	18	1210	1
Buried, mobile region	5	85	6	69	1210	6

Table IV Classes of mutation enriched in the outlier population. Residues making buried hydrogen bonds, hydrogen bonds to buried water molecules, or contacting ligands are enriched in the outlier population, as well as mutations affecting the unfolded state.

The largest errors in accuracy are for cases where polar residues are swapped for hydrophobic residues or vice versa, with correlations ranging from 0.55-0.6 (Table III and Table IV), which suggests the largest areas for improvement involve the delicate trade-off between polar desolvation and the formation of favorable buried polar interactions. Consistent with this, buried hydrogen bonds are two-fold enriched in the outlier population (Table IV) (19% versus 9%). Cases in which an unsatisfied hydrogen bonding group is introduced in a buried hydrophobic environment are also enriched in the outlier category (8% versus 4% percent). Finally, buried residues making hydrogen bonds to water molecules, an interaction absent in our implicit solvation model, are somewhat enriched in the outlier class as well (9% versus 4%). The development of polarizable electrostatics models and the inclusion of explicit water molecules(*18*) may help better recapitulate the energetics of these interactions.

Buried residues are in general predicted better than exposed ones, as has been reported in previous studies(*3, 5, 6, 19*). Although the correlation within the category of exposed residues is poor (r = 0.47), the stability-classification accuracy is very similar (0.71 for exposed mutations and 0.78 for buried residues, Table III). Because mutations to exposed residues are mostly neutral, they are easy to categorize even if their ΔΔGs are challenging to predict.

To examine the potential contribution of the unfolded state, we collected 305 m-values for mutations from staphylococcal nuclease(*20-23*). Mutations whose m-values significantly affected the energy of the unfolded state (>= 20% difference from the wild-type m-values) were enriched 4 times more than average in the outlier class (34% versus 8%). Previous studies have also noted difficulties in modeling this class of mutation accurately(*4, 5*). Improved modeling of such mutations may require explicit modeling of context dependent unfolded state effects.

We observe only a marginal decrease in performance for mutations in high B-factor regions when compared to low-Bfactor regions (0.68 versus 0.64, *fixed-backbone, all sidechains repacked, row 6 Table I*) (Table III). Inclusion of backbone flexibility reduces the discrepancy further (0.69 versus 0.67, limited backbone minimization after sidechain repacking, row 16, Table I). Entropic effects may contribute to prediction inaccuracy overall, but are not as evident as might be expected in this subset of the data.

Conformational sampling appears to be still in part limiting. The outlier class includes a number of mutations of large to small hydrophobic residues. The free energy changes in these cases are predicted to be extremely destabilizing due to the creation of a large hydrophobic cavity whereas the effect of the mutation is near neutral, indicating significant conformational rearrangements. Comparison of our predictions to the mutant crystal structure(*24, 25*), suggests some failures are due to the inability to sample correct conformations.

Discussion

Previous studies have shown that free energy changes accompanying point mutations can be reasonably well predicted, but the features contributing to this success are not evident as the different methods use very different sampling procedures and energy functions. Here we demonstrate that the free energy changes associated with point mutations can be predicted equally well by protocols that involve widely varying amounts of conformational sampling, provided that the resolution of the energy function matches the coarseness of the sampling. As found in previous studies(*26*), protocols involving coarse conformational sampling perform well when repulsive interactions are damped, whereas protocols involving aggressive conformational sampling perform well when repulsive interactions are not damped. We find that protocols that incorporate backbone flexibility are better suited than fixed-backbone protocols for modeling small-to-large mutations, but the preponderance of large-to-small mutations masks this improvement on the overall dataset. Expanding on the results of Dantas *et al*., we show that the best methods for modeling small-to-large mutations utilize a damped energy function for sidechain optimization followed by an undamped potential during constrained, gradient-based minimization (row 16, Table I). When used during optimization, the hard-rep energy function can select incorrect rotamer conformations (row 17, Table I), which are often not rescued during the subsequent round of gradient based minimization because of the difficulty in crossing high-energy barriers.

Our calculations model the contributions of mutations to the free energy of folding in different ways. The change in enthalpy resulting from the mutations is calculated explicitly through Lennard Jones interactions, hydrogen bonding, etc. Interactions with solvent, both enthalpic and entropic, are modeled using an implicit solvent model(*27*). The changes in the entropy and enthalpy of the unfolded state are assumed to be context independent: for example, the change in unfolded state free energy for all leucine to alanine substitions are assumed to be identical. This assumption clearly breaks down when the residue is making specific interactions and/or has restricted conformational freedom in the unfolded state(*20, 28*). The reasonable success rate in predicting ΔΔGs with this rather drastic assumption suggests that unfolded state effects are not major contributors to the ΔΔG, but as noted in the results they could well be responsible for some of the deviations between the computations and experiments.

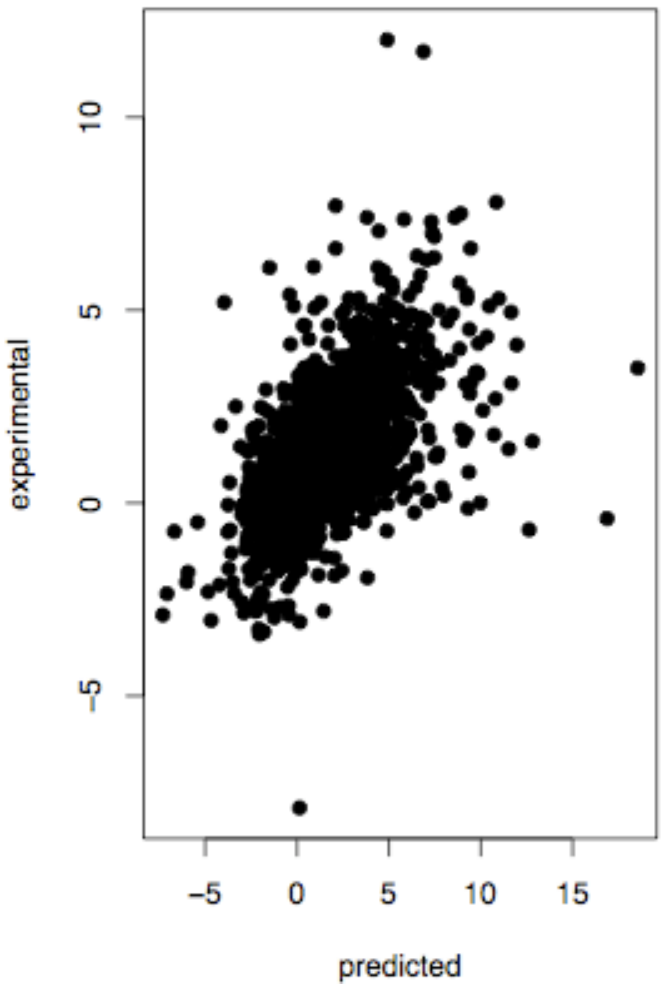


Figure 7: Best performing ddG protocol with limited backbone minimization produces a correlation of 0.59 on a set of 1,857 mutations curated by Potapov et al. Outlier mutations with a predicted ddG of Erep > 7 were removed according to the filtering procedure used by Potapov et al. and applied to the results from all protocols tested.

Method	*r*	*n*	outliers
CC/PBSA	0.56	478	-
EGAD	0.59	1065	1091
FoldX	0.50	1200	-
Hunter	0.45	1594	-
I-Mutant2.0	0.54	933	-
Rosetta	0.53	1871	0
Rosetta (remove clashes)	0.59	1857	14

Table V The best performing backbone-flexible ΔΔG prediction protocol outperforms all other published protocols on a larger set of mutations.

The poor results of Potapov et al. resulted from use of a limited sampling protocol without dampening the repulsive interactions. Consistent with the previously reported results, a protocol analogous to that of Potapov,(*row 4, Table I*), produced a correlation near 0 for our benchmark set of 1,210 mutations. On the datset used by Popatov, our best performing method using limited backbone minimization (*row 16, Table I*) yields an overall correlation of 0.57 on 1,937 mutations, and 0.62 on 1,920 mutations of the Potapov set(Figure 7, Table V) (excluding as in the Potapov study mutations with repulsive interactions of 7 units); this is equivalent to the performance of the best algorithms tested by Potapov. For comparison, the EGAD method had a correlation of 0.59 on a set of 1,065 mutations, FoldX had a correlation of 0.50 on a set of 1,200 mutations, and CC/PBSA had a correlation of 0.56 on a set of 478 mutations.

Our best-performing method for ΔΔG prediction involves limited backbone minimization; with training and 20-fold cross-validation it produces a correlation of r=0.73 on a comprehensive set of 1,210 mutations (Table II), matching that of previously published algorithms but on a larger test set. Although addition of protein flexibility in some cases improves modeling the structural response to mutation, we find that more often than not, more aggressive remodeling can decrease the ability of a method to recapitulate mutant structure and can have correspondingly negative impact on ΔΔG prediction. More extensive sampling with more accurate potential functions hopefully will reverse this disappointing fall off in predictions in the not too distant future. Analyses of consistently badly predicted mutations among all methods reveal that improvements in modeling the unfolded state, buried polar networks, and explicit water or ligand contacts may be the key to further improvements in performance. There is clearly much room for improvement in ΔΔG prediction methodology.

In conclusion, while the ΔΔG protocol has been significantly improved over the previous protocol (Kortemme et al. PNAS 2002), there is still room for improvement. Current performance of the best protocols on blind datasets is around $R^2 = 0.54$. The amount of experimental noise inherent in ΔΔG experimental values indicates that the maximal correlation possible is around $R^2 = 0.89$, indicating that quite a gap exists between theory and experiment. However, the avenues for improving thermodynamic modeling of mutation are not easy; while some evidence hints that improving models of the unfolded state would enhance the predictions of some mutations significantly, it is not clear how best to model the energetics of the unfolded state in a computationally efficient manner. While the previous analyses have indicated that surface exposed mutations and mutation of charged residues are poorly modeled, it has been a large and establish field of research to model the contribution of electrostatics to protein thermostability, and doing so in a computationally efficient manner is extremely difficult.

Applications of ΔΔG prediction

There are many industrial and scientific applications for which ΔΔG prediction is sufficiently accurate to be substantially useful. For example, thermostable enzymes are useful in industrial processes which require high heat or low pH. Such a thermostable class of enzyme, lipases, is a staple of laundry detergents used by most households today. Such industrially useful enzymes can be routinely produced through application of ΔΔG prediction to thermostabilize the enzyme without affecting enzymatic activity(*29*). Additional applications include those in which the effects of protein stability on protein function are not known but can be quickly assessed computationally(*30, 31*). One particularly useful application of ΔΔG prediction includes thermodynamic interpretation of large numbers of sequence variants, or in testing structural hypotheses, validating computational models.

Understanding the link between sequence, structure, and function has been a long-standing goal in the field of protein science. Many previous studies have relied on the painstaking work of mutating a position and then characterizing the protein both biochemically and structurally(*32*). These studies have clearly been essential contributions to our current understanding of protein biophysics, but the amount of work needed to characterize each mutant is intractable when considering all possible sequence variants. Coupling large-scale library generation with deep sequencing technology creates a high throughput method of studying the relationships between protein sequence and protein function both efficiently and thoroughly. However, the selection results are often ambiguous; the best performing (most highly selected for) variants are enhanced in both stability and activity, and given the enormous size of the sequence libraries, it is impossible to biochemically characterize each mutant. Thus, $\Delta\Delta G$ prediction, coupled with computational binding predictions, can provide a way to separate the effects each mutation has on a variants stability and function.

$\Delta\Delta G$ prediction applied to recapitulate ww-domain selection data

The ww-domain was chosen as a model system which is well studied and characterized(*33-40*). It is known to bind a peptide ligand, which was used as a criteria for selection(*41, 42*). Diversity was generated through creating a library of variants mutating the central 33 residues. Complete coverage was obtained for all single and double mutants as well as a significant fraction of triple mutants(*43*). T7 bacteriophage was used to display over 600,000 different variants of the ww-domain. The mutant population was then subjected to 6 rounds of selection for moderate enrichment of better binders to the cognate ligand, allowing the population to maintain diversity while also improving average fitness. The initial library was sequenced, as well as the libraries obtained after three or six rounds of selection. Sequence enrichment and depletions for these selected libraries were then compared to the initial library.

In order to study how sequence enrichment and depletion correlates with protein stability, computational ΔΔG predictions(*17*) were used to assess the effect of a mutation on protein stability for 18,568 variants(*43*). The ΔΔG protocol used involved limited backbone and sidechain minimization and yielded the best performance in terms of predicting the effects of non-conservative mutations as well as overall pearson correlation coefficient (0.69) (figure 4 and Row 16 of Table I). Binding energy calculations were performed for positions which are in direct contact with the ligand.

ΔΔG predictions were observed to negatively correlate with enrichment data (Figure 8). Many of the single and double mutants which are predicted to be strongly destabilizing were depleted in the sixth round of selection (Figure 8). This result is expected, and is consistent with the fact that the ww-domain was subjected to only moderate selection as well as the possibility that binding might be significantly enhanced in cases where mutant stability had little to no predicted effect on protein structure. Wild-type wwdomain was enriched by 1.75 fold, but many other variants were enriched much more than this, indicating that binding of the ww-domain to its cognate ligand can be significantly improved with just a few point mutations. This method demonstrates an application where true union of computational and experimental methods can complement one another in order to discover large-scale sequence determinants of protein structure and function.

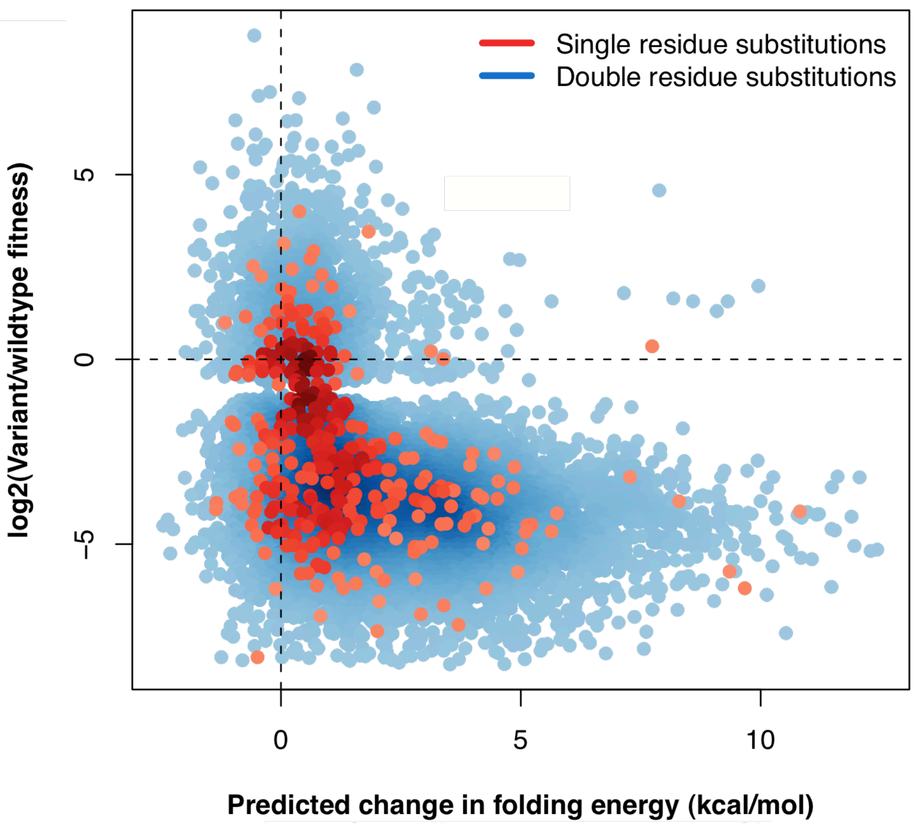


Figure 8. ΔΔG predicted changes are negatively correlated with enrichment data. Predicted changes in ΔΔG in kcal/mol (x-axis) against log2 enrichment values (y-axis) demonstrate that mutations predicted to be significantly destabilizing are selected against, whereas mutations which are neutral or stabilizing are enriched in the library resulting from six rounds of selection.

Other interesting applications include cases in which ΔΔG prediction can be used to confirm computational models. In a case of a T4 Lysozyme mutant, evidence of a higher-energy, transiently populated alternative state was tested by using the ΔΔG protocol to stabilize and thus confirm the model, which will be described in more detail in the next section.

Chapter Two: Modeling Free-energy landscapes with Rosetta

Introduction to Protein Excited States

While much attention has been paid to the process by which a protein attains its final, native state, much recent and exciting work has been in studying the excursions from the native state and, in particular, resolving the structure and importance of these higher energy, transiently populated states. I will outline the work in a few interesting cases: T4-lysozyme, Ubiquitin, NtrC, protein G, and proline isomerase.

T4 Lysozyme mutant L99A

Crystallography, while an indispensable tool for studying protein structure and function, can at times be misleading, as the order observed in crystals often falsely portrays a static picture of protein structure. An interesting case which demonstrates this nicely is the case of the mutation L99A in T4-lysozyme. The mutation L99A introduces a large cavity of ~150 Å but despite this, the crystal structure of the mutant is virtually identical to that of the wildtype crystal structure (*44*). Despite this, NMR studies of this mutant demonstrated spectral broadening which is indicative of microsecond - millisecond timescale dynamics(*45*), hinting that the cavity-causing mutation introduced new dynamics into the structure. Furthermore, the mutant but not the wild-type was shown to bind benzene(*46*), yet how the benzene molecule was able to bind within the core of the protein was not known.

This case is particularly interesting in that it is the first example of using Rosetta structure prediction and design in order to stabilize a non-native structure with an energetic ΔΔG of approximately 2 kcal/mol higher than the native structure(*47*). CS-Rosetta was used(*48*), in conjunction with excited-state chemical-shifts(*49*), to model regions of the protein whose chemical-shift significantly differed from the corresponding ground-state chemical-shift(*47*). The mutation prediction algorithm previously described(*17*) was subsequently used to predict stabilizing mutations which would favor the excited state over the ground state. The original mutant, L99A, shows an excited-state population of 3%. The Rosetta-predicted mutant, G113A, is predicted to stabilize the excited state over the ground state by −0.75 R.E.U. (Rosetta energy units) and shifts the excited state fractional population to 34%. A subsequent mutation predicted by Rosetta, R119P (predicted to stabilize the excited state over the ground state by −7.3 R.E.U.), in the context of the previous two mutations, inverts the population such that the excited state in solution is dominant at 96%. This is the first example which demonstrates that Rosetta is capable not only of modeling the ground-state structure but also excited states as well. This case demonstrates a unique approach to studying metastable transient states in the protein free-energy landscape, and its success suggests a general strategy to couple experimental and computational methods in order to further understanding of protein sequence structure relationships.

Ubiquitin

Ubiquitin recognizes a variety of different protein partners with different specificities and modes of binding. Although many crystal structures exist of Ubiquitin bound to a protein partner, the structures are heterogeneous and do not demonstrate how Ubiquitin can recognize and bind so many protein partners(*50-52*). NMR RDC measurements of Ubiquitin were refined with EROS (Ensemble Refinement with Orientation Restraints)(*53*) to produce a structural ensemble. Incredibly, the structures in this ensemble encompassed all the variation observed in the crystal structures, supporting the idea that ubiquitin undergoes conformational selection instead of an induced fit mechanism to bind to its partners. Furthermore, only five degrees of freedom were needed in order to describe the fluctuations observed in the EROS ensemble. Finally, the ensemble of ubiquitin contained three rigid residues, known to be hotspot residues for Ubiquitin binding, and suggest a general mode of interaction with the protein binding partners.

NtrC

Many proteins, such as signalling proteins demonstrate conformational changes upon phosphorylation or changes in environment. NtrC is a signaling protein which becomes activated upon phosphorylation and is demonstrated to undergo structural changes(*54*). In the unphosphorylated state, NtrC shows micro- to millisecond timescale dynamics, and is partially active. This ensemble showed NMR chemical shift differences in the same region as that which undergoes conformational change upon activation. These chemical shifts belonged to an active form of NtrC, and was found populated at 14% in the unphosphorylated state. A single mutation, D86N, was found to stabilize the active state, shifting the percentage of the population in the active state to 43%. A second mutation, A89T, stabilizes the active state further, at 65%.

Proline Isomerase

X-ray crystallography is usually performed at cryogenic temperatures, needed to preserve samples and improve sample data collection. However, crystal structures solved at ambient temperatures can reveal significant features of protein dynamics. In an interesting study on cyclophilin A(*55*), the solution to an x-ray structure solved at ambient temperature included fitting multiple sidechain conformations to electron density. A single mutation, S99T, was chosen such that the new sidechain would fill the space occupied by both Serine rotamers, 14 Å away from the active site. Subsequent NMR studies showed that protein dynamics had slowed due to the mutation, and the catalytic rate was reduced by 300 fold, indicating that the dynamic nature of the Serine residue plays an important role in the catalytic cycle.

Protein G

There are many cases of natural proteins switches, drastically changing conformations upon changes in environment or sequence. Some more subtle examples of these are prion proteins, which can fold into its native structure consisting of a mainly alpha-helical topology, which is benign and non-infectious, or can mis-fold into an infectious beta-sheet form(*56*). Another class of protein, the Cro family, functions as transcription factors and are highly similar in sequence yet populate different folds(*57, 58*). While these examples may initially seem unrelated, a very detailed understanding of the relationship between protein sequence and structure is needed in order to generally and fully explain the behavior of such examples. Traditionally, much of biochemistry operates under the assumption that the change in free energy for changing a protein's native form into an alternative structure is very high. However, this study demonstrates how a one could potentially engineer two proteins which populate very different topologies yet have very similar sequences. Furthermore, the protein need not unfold in the transition from one conformation to the other. One protein, the GA domain of protein G, binds human serum albumin and consists of a three-helical bundle, and the other, GB, binds to the constant region of IgG and consists of an alpha-beta topology with four beta strands and one helix. Alexander et al. engineered these two starting points to share 77% sequence identity yet populate different topologies(*59*). By attempting to discover the minimal path of mutations which would convert one form into the other, Alexander et al. found that a single mutation was sufficient to switch one topology into the other(*60*). Yet, as the sequences become more and more similar, subsequent mutations seemed not to affect the stability of the native state but seemed to stabilize the alternative state. Indeed, at the switch point, the protein was found to populate the alternative topology to a significant fraction and could bind both partners. This is an interesting example of how protein sequence evolution can have little negative impact on functional fitness.

These exciting cases show that the protein structure community must begin to consider not only the lowest energy, native state, but the entire ensemble of protein configurations accessible from the native state in order to comprehensively understand how protein sequence relates to structure and function. While experimental methods have proven essential in elucidating these first examples of how transient excited structures are important for understanding protein function, computational methods have yet to prove useful in providing hypothesis or prove that they can elucidate these excited-state structures in the absence of experimental data. One reason for this is that the dynamic fluctuations between the ground and excited state structures can be very slow, often on the order of milliseconds, which up until recently have been beyond the limits of computational resources. However, with recent advances in computational hardware and theory, these long timescale simulations are now within reach.

Motivation to Study Protein Kinetics and Thermodynamics within Rosetta

While protein kinetics has been shown to be important for protein function and is important for understanding fundamental problems such as protein folding. These kinetics are often slow and require prohibitively long simulation time, which is not commonly accessible with current computing resources. Molecular Dynamics is a common framework with which to study protein kinetics, but it is limited to relatively small systems due to computational resource requirements. On the other hand, Molecular Mechanics techniques such as Rosetta, although not a simulation of dynamics, can still elucidate thermodynamics of the system as long as the thermodynamic moves are sufficiently small so as to not cross meaningful kinetic barriers.

Furthermore, degrees of freedom are much easier to isolate using Molecular Mechanics. Thus, adapting a framework such as Rosetta may prove to be useful in studying the thermodynamics and kinetics of protein structure.

Although previous studies using Rosetta have often described funnel-like landscapes when performing structure prediction simulations for proteins up to 150 residues(*61*), the current framework within Rosetta does not support a kinetic interpretation of the data; because simulations were not generated within a thermodynamic framework, the thermodynamic relevance of the simulations remain unexplored. Furthermore, a comparison of Rosetta's energy function with others, such as AMBER(*62-64*) or CHARMM(*65*), may be informative in improving the discriminative power of energy functions in general.

Thermodynamic framework in Rosetta

In order to sample free-energy landscapes within Rosetta, a number of changes needed to be implemented in the Rosetta framework. Move-sets can be made more efficient by fixing bond-lengths and angles and have been successfully utilized in producing free-energy landscapes. However, bond-stretching and torquing is known to facilitate crossing of free-energy barriers, and a fixed bond-length and angle assumption (such as in Rosetta) may contribute to artificially increasing the energetic barrier. Furthermore, in order to perform thermodynamic sampling, the move-set defined must satisfy detailed balance:

$$p_i p_{i\rightarrow j} = p_j p_{j\rightarrow i}$$

In practice, this is implemented by ensuring that the probability of the forward move i → j is equal to the probability of the reverse move, j → i. If the probability of making a move i → j were not the same as making the move j → i, then the equilibrium probabilities of i and j would be influenced by the probabilities of the moves themselves and would violate conservation of mass.

Move-sets

In order to implement a suitable move-set (refered to as a 'Mover') for sampling thermodynamics in Rosetta, we implemented a Metropolis Hastings Mover along with a variety of backbone movers, the Backbone Biased Gaussian Mover, the Concerted Rotation Mover, and a Sidechain Mover.

Metropolis Hastings Mover

The metropolis acceptance criteria is a fundamental part of Rosetta sampling; proposed structures are accepted or rejected according to the probability obtained by the metropolis criteria. The probability is influenced by the difference in energy between the proposed and the current structure as well as the temperature:

$$p_{accept} = e^{-E_{diff}/k_b T}$$

where kb is the boltzmann constant. Because detailed balance must be enforced in thermodynamic sampling, an additional term, the probability ratio of the forward move over the reverse move is an additional term in the acceptance probability:

$$p_{accept} = e^{-E_{diff}/k_b T}\left(\frac{p_{forward}}{p_{reverse}}\right)$$

Backbone Biased Gaussian Mover

The Backbone Biased Gaussian Mover (BBG Mover) solves sets of equations, using polypeptide phi and psi as variables, to ensure that the chosen set of residues are altered such that downstream perturbation of the polypeptide chain is minimized. This method is particularly attractive and amenable to implementation in Rosetta because it operates in torsion-space and assumes fixed bond length and angles. (*66*). The algorithm for performing a move operates in the following manner. Of a stretch of four consecutive amino acids and eight torsion angles, 4 torsion angles are perturbed according to a biased gaussian distribution, controlled by a user-defined parameter, A. The remaining 4 torsion angles are determined by solving a set of equations to ensure minimal downstream perturbation, the amount of perturbation allowed is also controlled by a user-defined parameter, B. For more details, refer to the Favrin et al(*66*).

Concerted Rotation Mover

Similar to the BBG Mover, the Concerted Rotation Mover (Conrot Mover) is similar in approach but enforces exact loop closure as well as allowing bond length and angle variability, making it attractive for thermodynamic sampling of loop conformations(*67*)

Backrub Mover

The backrub mover is an algorithm(*16*) which attempts to mimic the natural concerted motion observed in high-resolution (< 1 Å) crystallographic structures(*15*). This is implemented as a rotation around an axis defined by the starting and ending residues. The segment in between the starting and ending Cα residues is treated as a rigid body and can vary between 2 and 12 residues in length and is rotated between 11 and 40 degrees, depending on the segment size. Bond angle and length strain is minimized by analytically bracketing the maximal angular rotation and the introduction of energetic terms intended to restrain bond-lengths and angles to appropriate values. The application of this algorithm to sampling in Rosetta was relatively successful in a number of tests of structural recapitulation, ranging from point mutant structure prediction to loop prediction(*68*).

Sidechain Mover

The sidechain mover used for thermodynamic sampling had three different independent moves, each of which could be executed by a user-defined probability. The first is a random uniform perturbation of chi angles; not surprisingly, the acceptance rate for such moves tends to be low. The second is a small perturbation (maximally ten degrees) on chi angles based on the dunbrack rotamer distribution. The third is to randomly sample rotamer wells according to the dunbrack probability distribution.

Modifications for fast sidechain mover

In order to increase the efficiency of sidechain moves, a fast version of the sidechain mover was implemented. Because only one sidechain is altered at a time, most of the energetic interactions remain the same and need not be updated. This observation led to an implementation of the sidechain mover in which it maintains its own energy interaction graph. This simple energy graph is more efficient because only the altered sidechain energies are evaluated with respect to its neighbors. This efficient implementation of the sidechain mover is approximately 10-fold faster than the original mover (Figure 9).

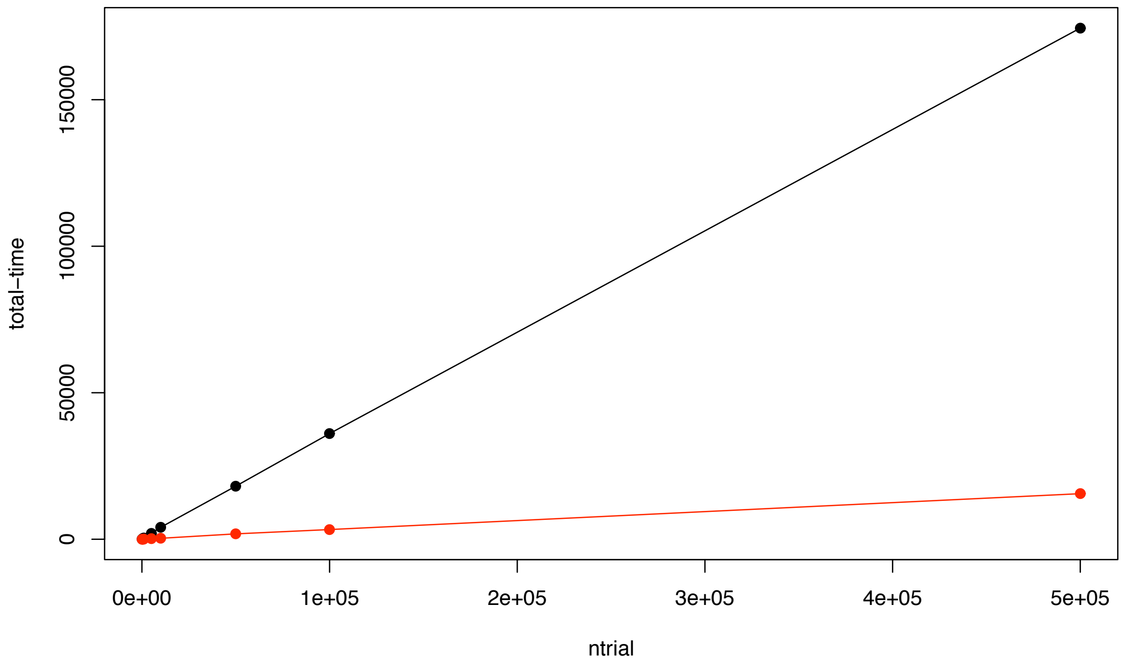


Figure 9. The fast sidechain mover is approximately ten times faster than the original sidechain mover. For increasing numbers of sidechain perturbations (x-axis), the total time required to complete the sampling is much lower for the fast sidechain mover (red) than the original sidechain mover (black). As expected, the efficiency gains for the fast sidechain mover increase linearly with number of trials.

Observers for data-collection/organization

To allow for parallelization of thermodynamic sampling, a variety of observers were implemented to facilitate data collection and organization. If all simulation data were stored, the size of the resulting simulation files can easily span hundred of GB. Because the landscape explored is vast and many structures could potentially be redundant, the observer served to store only significantly different structures (defined by an rmsd-threshold). The observer was implemented using MPI, and is structured such that one processor functions as a 'master' node. This master node collects information from all processes and determines whether or not newly discovered structures are redundant or unique. If unique, then the master will store the structure in its pool of known structures and if not, the structure will be discarded. In the case of Markov State Model construction (see section on MSM model construction), this is especially useful as the unique structures represent discrete states, a fundamental component to Markov State Model construction. By using this observer during the simulation, the process of collecting data and assigning discrete states is done on-the-fly.

Parallel Tempering

Parallel Tempering, also known as replica-exchange, is useful for increasing the efficiency of sampling and is often used to construct free-energy landscapes of protein folding. Multiple parallel simulations are maintained at different temperatures, and periodically the temperatures for the trajectories are swapped according to a modified metropolis probability of acceptance:

$$p_{accept} = e^{-E_{diff} / k_b T_{diff}}$$

Choosing temperature intervals is extremely important; despite this, there is no consensus on how to best choose temperature intervals(*69-71*). General considerations include ensuring the highest temperature is sufficiently high to allow the system to escape local-minima. Conversely, the lowest temperature should be sufficiently low to allow the system to explore energetic minima. Finally, sufficient mixing between the energy levels is equally important and related to the efficiency of sampling. One strategy to ensure that temperature intervals are sufficiently well chosen is to construct energy histograms sampled by each temperature level; if the energy histograms overlap, then one can be reasonably sure that sufficient mixing will occur. An orthogonal check is to verify that the acceptance ratio for temperature swaps is sufficiently high, ideally around 50%. (*72*). Finally, it has been suggested that the temperature schedule take on an exponential form:

$$T_f = t_i e^{(bn)}$$

where n is the processor number and t_i and b are chosen quantities(*70*). In order to define temperature intervals in our simulations, we picked values of 0.1 for t_i and 0.14 for b. For 24 processors, the temperature levels range from 0.08 to 3. From ΔΔG predictions and melting simulations (data not shown), we estimated Rosetta's room-temperature to be around 0.5. We verified that the energy histograms for each trajectory overlapped sufficiently and that the acceptance rate of temperature swaps was between 20% - 50%.

In order to construct a free-energy landscape from Replica-exchange data, we note that when free energy is expressed in terms of a reaction-coordinate, it is called a Potential of Mean Force, PMF. The PMF is referred to as W and is a function of the reaction coordinate, q(*73*):

$$W(q) = -kT\ln\left(P(q)\right)$$

Great care must be taken in order to construct an accurate and meaningful reaction coordinate, as poorly chosen reaction coordinates can hide important features of the free-energy landscape(*74*). For our purposes we found it sufficient to construct free-energy landscapes using the simplest reaction coordinate: rmsd to the native structure. However, subsequent analyses of this data may benefit from constructing two-dimensional reaction coordinates and using percentage of native contacts as an indicator of distance from the native structure instead of rmsd.

Benchmark Set

To test the sampling efficiency of Rosetta parallel tempering, we chose four proteins, well-studied both computationally and experimentally, in a range of sizes: trip-zip(*75, 76*) (12 residues), trp-cage(*77*) (20 residues), villin(*78*) (35 residues), and the ww-domain(*33-40*) (35 residues). We initiated each trajectory from a fully extended polypeptide chain and monitored the trajectories for two weeks for the smallest system, and up to one month for the larger systems: villin and the ww-domain.

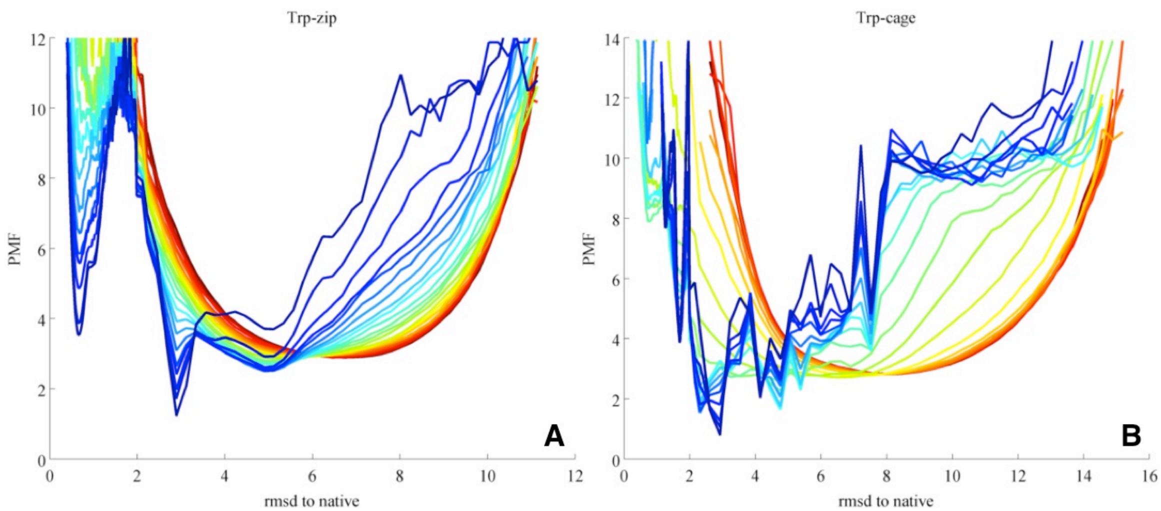


Figure 10. The free-energy landscapes of Trpzip and Trpcage show that although native-like structures are sampled, they are not the lowest in free-energy; non-native structures distinct from the native structure are the lowest in free energy. Blue indicates the lowest temperature, whereas red indicates the highest temperature simulation.

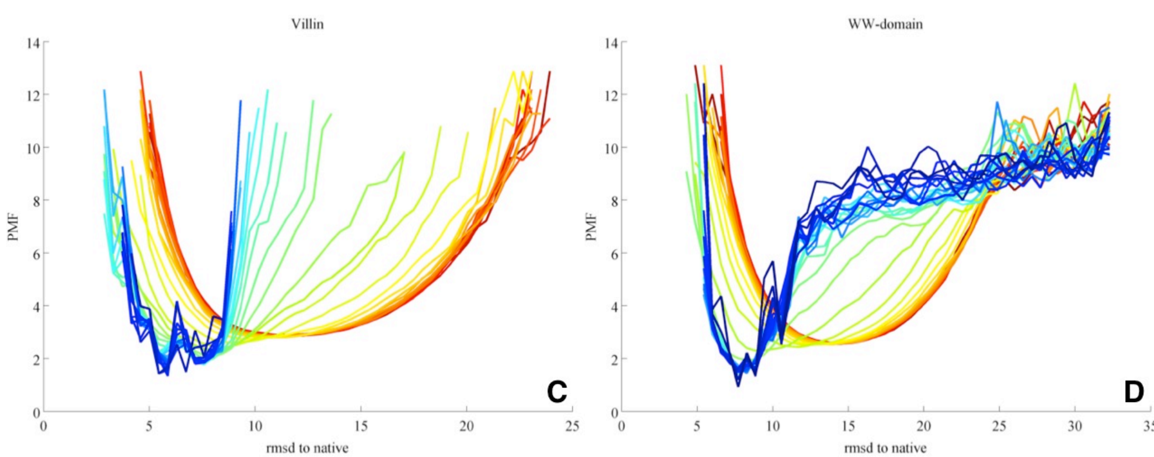


Figure 11. The free-energy landscapes of Villin and the WW-domain show that near-native structures are not sampled, but similar to the case of Trpzip and Trpcage, the lowest free-energy minima are far from the native structure. Blue indicates the lowest temperature, whereas red indicates the highest temperature simulation.

Results

The free-energy landscapes for each system under 20 residues were reproducible. The features of the landscapes were also identical, suggesting simulation convergence. Trpzip and Trpcage produced native-like decoys, the closest being 0.37 and 0.36 Å rmsd, respectively. However, in both cases the minima corresponding to the native structure were not the lowest in free-energy (Figure 10). In the case of the Trpzip protein, one minima was observed in accordance with the native structure at ~0.5 Å rmsd, but another minima lower in free-energy was observed at 3 Å rmsd (Figure 10A). In the case of the Trpcage protein, the minima corresponding to the native structure is among the highest in free-energy, whereas a set of minima ranging from 2 – 4 Å rmsd from the native structure were among the lowest in free-energy (Figure 10B).

For systems larger than twenty residues, villin (35 residues) and the ww-domain (35 residues), native-like structures were not sampled despite nearly a month of simulation (Figure 11). The closest structures sampled were 2.96 Å rmsd and 4.63 Å rmsd, for villin (Figure 11A) and the ww-domain (Figure 11B) respectively. Furthermore, the free-energy landscapes do not demonstrate funnel-like behavior, with the lowest structures in free-energy being approximately 7 Å rmsd (villin) and 8 Å rmsd (ww-domain). Despite the large range of Rosetta temperatures sampled (ranging from 0 to 3 in arbitrary units) as well as multiple simulation runs which vary the temperature schedule, no improvement in sampling native-like structures was observed.

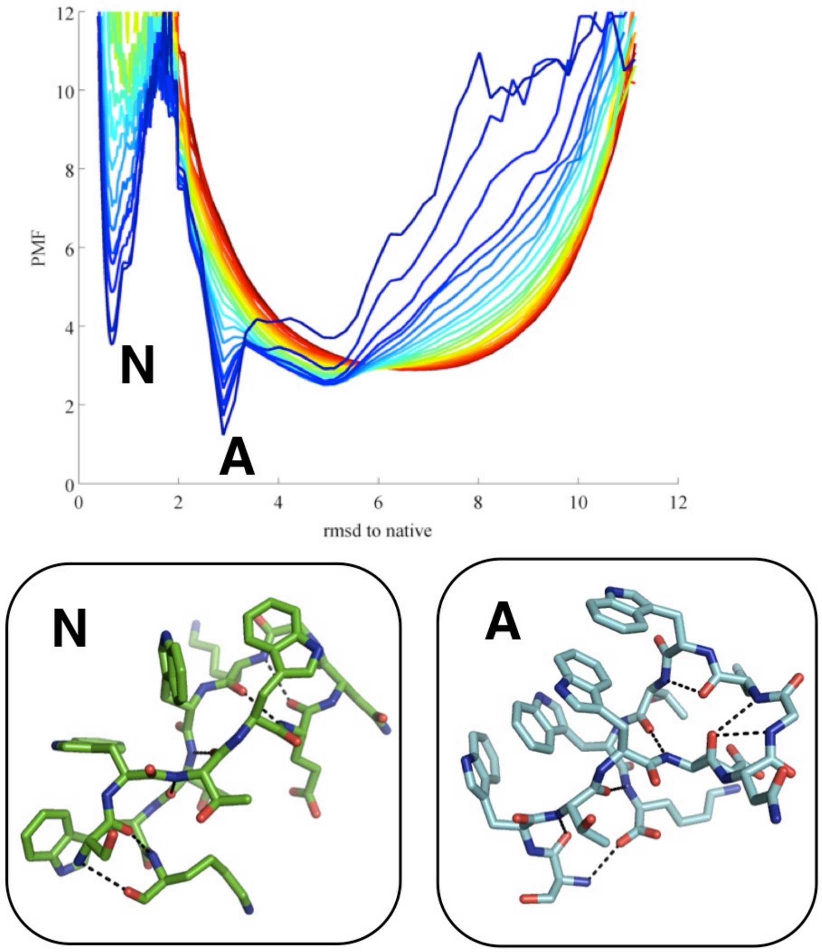


Figure 12. The structure corresponding to the native minima (labeled N) and the lowest free-energy minima (labeled A) are shown. The alternative labeled A exhibits non-native like features, such as p-stacking of Tryptophan residues and an altered hydrogen bonding structure.

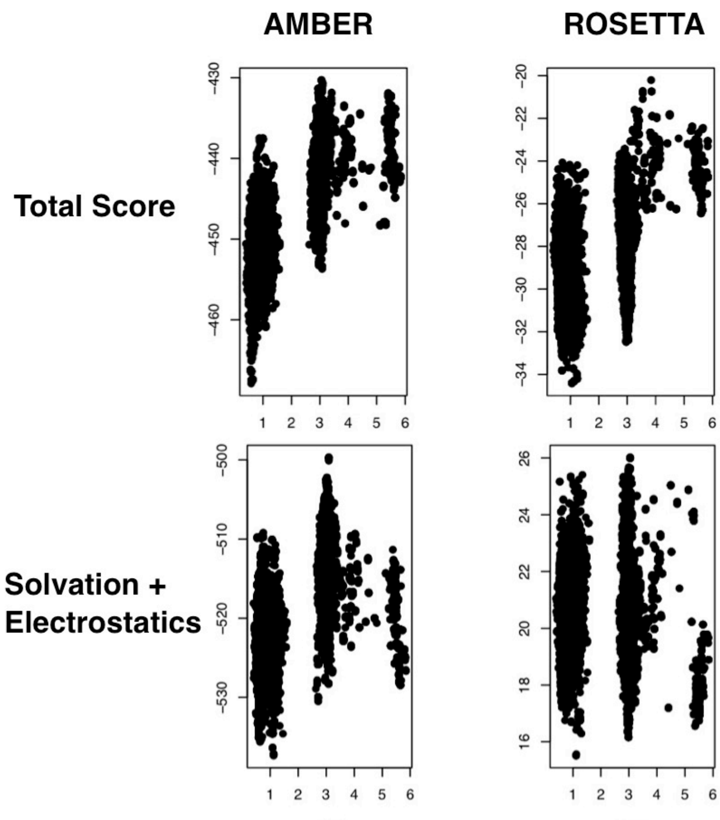


Figure 13. Comparison of AMBER and Rosetta scores on structures corresponding to the native and alternative minima show that both energy functions rank structures similarly. The top panel shows the comparison of total score between AMBER (left column) and Rosetta (right column). The bottom panels show the comparison of the sum of solvation and electrostatic potentials.

Alternative minima are artifacts of the energy function

The alternative minima observed in the Trpzip free-energy landscape (Figure 12 labeled A) is 3 Å rmsd from the native structure and demonstrates interesting features which are not usually seen in protein structure. For example, the four tryptophans p-stack in order to maximize Van der Waals contacts, and the hydrogen bonding structure of the beta hairpin is altered such that hydrogen bonds are still satisfied but regular secondary structure is eliminated. These structural features suggest that the alternative structure is an energy function artifact, and accordingly, is not supported by any experimental evidence.

To investigate further the Rosetta energy terms that might be incorrectly favoring this structure, we took a low energy subset of the ensemble and minimized the structures using either the Rosetta energy function or the AMBER f99SB energy function(*62, 79*). The total score difference between the Trpzip native and its alternative structure are separated by roughly equivalent energy units in total score (Top panel, Figure 13).

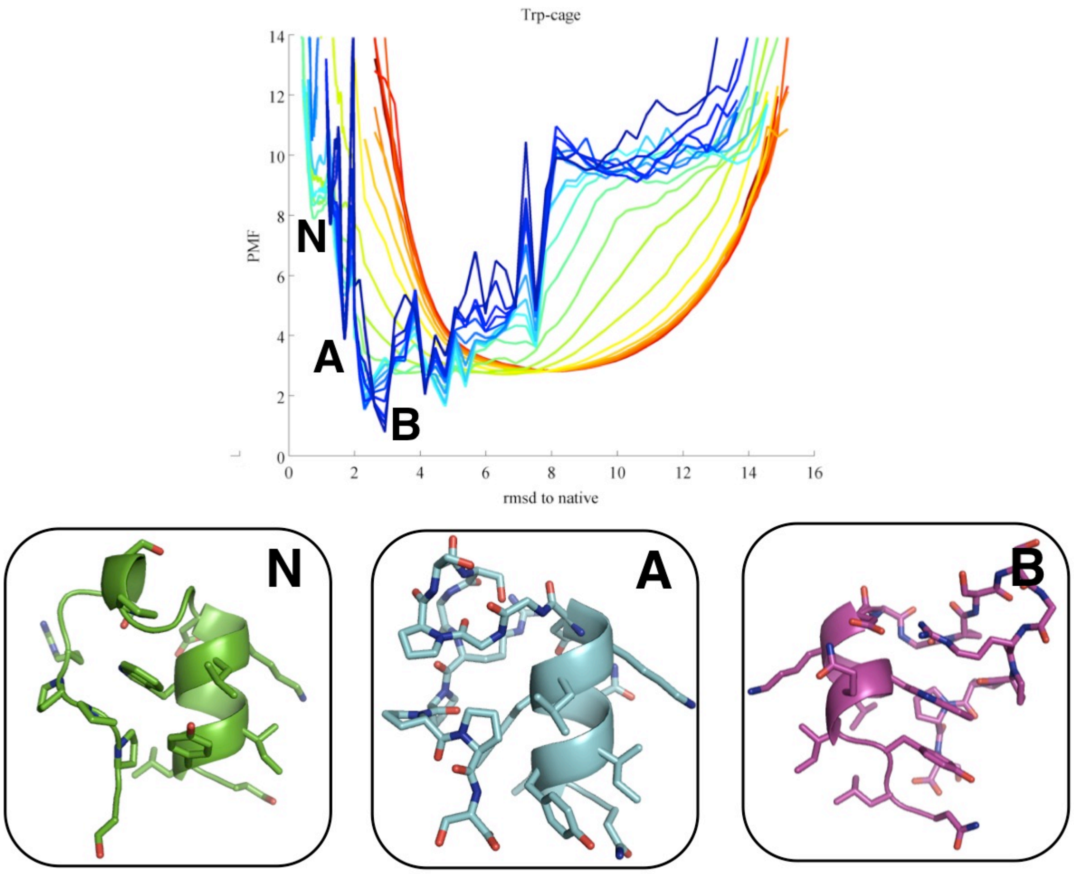


Figure 14. The free-energy landscape of the Trpcage system shows distinctly non-native alternative structures (labeled A and B), which are lower in free-energy than the native (labeled N).

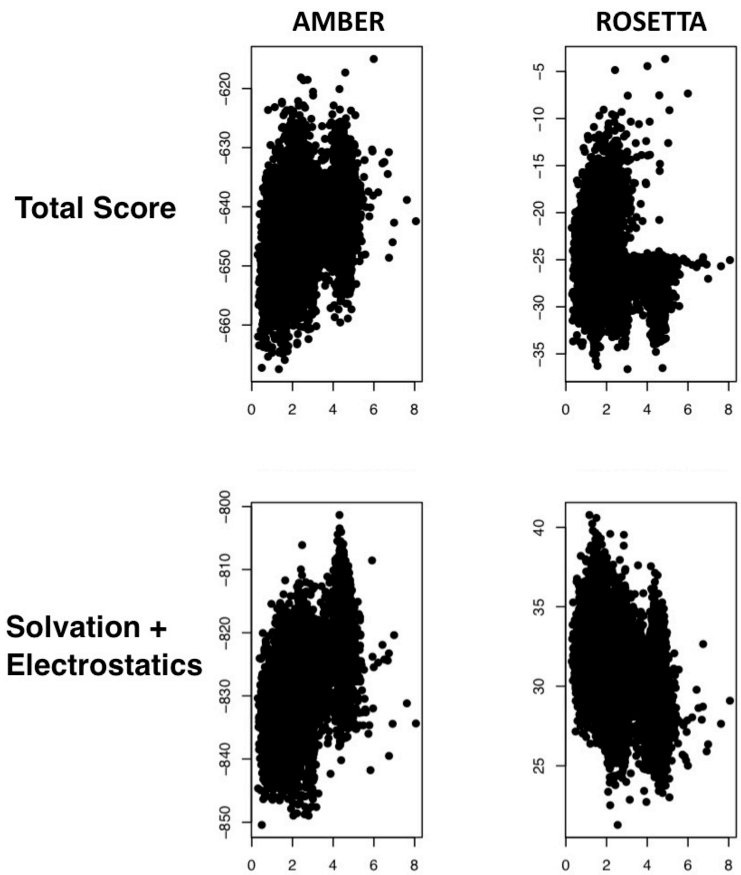


Figure 15. Comparison of the lowest scoring structures using AMBER (left column) or Rosetta (right column) total score (top panels) or the sum of solvation and electrostatic terms (bottom panels). While AMBER shows good native discrimination, Rosetta shows poor native discrimination, which is demonstrated most clearly in the solvation and electrostatic components.

For the Trpcage system, similar to that of Trpzip, native-like structures were simulated (N in figure 14). However, here too multiple alternative minima were observed, distinct from the native structure and lower in free energy. One of these minima (labeled A in Figure 14) had an alternative packing conformation for the central tryptophan residue, exposing it to solvent instead of burying it in the core as in the native structure (N in Figure 14). Furthermore, the unstructured turn contains an unsatisfied carbonyl backbone at position 16, which is hydrogen bonded to the central tryptophan (position 6) in the native structure. Interestingly, this structure corresponds to the lowest scoring structure resulting from an ab-initio structure prediction of the Trpcage structure, which is described as a result of artifacts in the Rosetta energy function(*80*). Finally, the minima corresponding to the free-energy minimum (B in Figure 14) corresponds to p-stacking of the tryptophan and the tyrosine and is 3 Å rmsd from the native structure. The same analysis as that performed with Trpzip was performed with Trpcage, the results of which are displayed in Figure 15. Unlike Trpzip, the Trpcage ensemble shows clear differences when comparing total AMBER score with total Rosetta score; while the AMBER scored structures show a clear preference for the native, with a funnel-like landscape, the Rosetta ensemble shows no preference. When the sum of solvation and electrostatic terms between AMBER and Rosetta are compared, the AMBER scores again shows native structure discrimination, whereas the Rosetta scores are anticorrelated with distance to native. Based on this comparison, energy function improvements might be guided by emulating solvation or electrostatic models utilized by the AMBER force field.

Conclusions

In systems where sampling is sufficient, energy function artifacts are observed

Seeding parallel tempering runs starting with an extended polypeptide chain yielded free-energy landscapes which were reproducible and produced near-native decoys for the 12 residue Trpzip peptide and the 20 residue Trpcage peptide. Unfortunately, significant alternative minima are also reproducibly observed, corresponding to structures which exhibit features that are not generally observed in protein structure. In the case of the Trpcage system, one of these stable minima corresponds to a structure reported previously as proof of Rosetta energy function artifacts(*80*). Because previous simulation studies have successfully utilized implicit solvation potentials (similar to that of Rosetta) to create folding free-energy landscapes for these particular systems, these artificial minima may be due to artifacts specific to the Rosetta energy function. Initial investigations into the source of these energy function artifacts through comparison to other Molecular Mechanics (MM) potentials, such as AMBER, hint that improvements to Rosetta solvation or electrostatic potentials may be sufficient to eliminate these minima.

In systems where sampling is insufficient, source of issue remains unresolved

Two of the four systems simulated, villin and the ww-domain, did not produce near-native decoys. Because of this, it is difficult to determine whether defects in sampling algorithm or energy function are responsible for failure to produce native-like free-energy funnels as expected. However, parallel tempering molecular dynamics simulations were previously used to obtain free-energy landscapes for these systems successfully with implicit solvation potentials(*81-84*). Furthermore, other studies which utilized the same Monte-Carlo based move-sets were able to obtain native-like structures and free-energy landscapes for the systems studied here(*81*)}(*84*) so it is unlikely to be due to the Monte-Carlo based move-set, the assumption of fixed bond-length and bond-angle, or implicit solvation potentials. Future work will include performing ab-initio simulations of these proteins, as well as re-examination of the villin and ww-domain data-sets after addressing the clear artifacts observed in smaller systems. Utilization of these datasets may well guide energy function improvement as well as benefit from studies to evaluate and increase sampling efficiency.

In parallel to our attempts to construct free-energy landscapes with Rosetta, we investigated alternative methods of constructing free-energy landscapes which do not require monopolizing computing resources for long periods of time. We discovered a recently developed framework, called Markov State Models (MSMs), that could obtain long timescale information from combining the information contained in many short, parallel simulations. Current methods require long simulations that wait for biologically interesting but rare conformational transitions, such as protein folding. However, MSMs would allow increased sampling of transition points, given that some information is known about these transitions, allowing one to resolve the interesting and important features of the free-energy landscape instead of sampling in more probable but less interesting regions of the free-energy landscape (such as a high entropy, unstructured states). As a first step towards utilizing MSMs, we reasoned that toy models would allow us to test analysis techniques, yield insight into the determinants of MSM accuracy, and identify potential shortcomings.

Chapter Three: Assessing and Improving Discrete Computational Models of Protein Kinetics

Markov State Model (MSM) toy model introduction

Previous toy studies of protein folding, such as lattice models, have yielded much insight into the process of how a protein folds(*85-89*). While these models are useful and also yield insight into the folding process, they are, by definition, discrete in nature, which makes them poor systems for examining the effect of discretizing continuous space inherent in MSM construction.

Previous methodological improvements to MSM construction have validated methods on the alanine dipeptide(*90*), for which the free-energy landscape is clearly defined. Because the system has only two degrees of freedom, the free-energy landscape is also easily visualized. However, Markov State Model construction does not depend on the details of the simulation. Thus, some important insights into the properties of the MSM framework itself could be obtained with the use of toy models. For example, Prinz et al. elegantly examined the error introduced by discretizing the simulations(*91*). They subsequently showed that this error can be made arbitrarily small by adding more detail in discretizing the transition regions, contrary to current opinion that the most accurate MSMs are those that define the most metastable states. In order to better study and understand the properties of Markov State Models, a small system which can be sampled to completion in a short amount of time and for which the energy landscape is easily visualized and known would be essential. Accordingly, we wrote and tested a suite of matlab scripts which would provide toy free-energy landscapes as well as scripts which would construct and visualize the resulting Markov State Models.

Because many simulations are required in order to sample fully the free-energy landscape, a common problem is in balancing the trade-off between number of states and sparseness of data. The number of states (clusters) determines the size of the resulting count-matrix (N clusters results in an N by N matrix), and a matrix consisting of thousands of elements quickly becomes sparse and numerically intractable. To avoid this problem, a common solution is to reduce the number of states by combining states with very few observations with the closest neighboring state by distance(*92, 93*). Another technique is to randomly sample the population, clustering only 10% of the dataset and then assigning the rest of the configurations to the nearest cluster center(*93, 94*). However, it is not clear whether these data-reduction techniques would result in obscuring important kinetic relationships or alter the resulting Markov State Model. In order to test this, we tested different data reduction techniques and described how they affected the accuracy of the resulting MSM.

Related to the issue of data-sparseness, fully sampling the free-energy landscape is impossible for any real system. Many simulations start from user-chosen starting configurations, but with the large size of the systems being studied, some biasing must result simply due to the choice of these starting configurations. Previous work has focused on determining the transitions which contribute the most uncertainty to quantities such as the mean first passage time (MFPT)(*95, 96*) and serves to focus sampling in underdetermined regions of the free-energy landscape(*96*). However, no work has been done to investigate the effect of biased sampling on the resulting MSM, an unavoidable result of imperfect sampling. Here, we conduct another study to examine the effects of biased starting points on a resulting MSM. To test this, we perform a toy-model simulation to investigate the effects of biased sampling, and in particular we focus on whether or not current MSM construction and validation techniques would indicate that incorrect results have been obtained.

Methods

Energy Function

The free-energy landscape is defined by the properties of the energetic minima. The minima, also called wells, have three different properties: weight, position, and width. Additionally, concavity of the landscape is controlled by a harmonic constant, k. Larger values produce a more funneled landscape, whereas a value of 0 would produce a completely flat landscape. Energetic noise can be modeled with an additional sine term, with two parameters for amplitude and frequency. By independently summing the energetic contributions of each of these features to the position on the two-dimensional landscape, we obtain the total energy E.

$$E = \sum_{i=1}^{N_{wells}} w_i e^{\left(-2(x_i - wc_i)^2 / \sigma_w\right)} + E_{gauss}$$

where w_i is the well-weight, x_i is the position-vector, wc_i is the position of the well centerpoint, s_w is the well-width, and Egauss is the non-gaussian energy, or a 'noise' factor, described by the sine term and the harmonic potential.

Sampling

Sampling mimics molecular mechanics simulations. Moves are made randomly, according to the step size determined by

$$stepsize = \frac{\left(2\max\_val \big/ N_{state} \right)}{f_{ss}}$$

f_{ss} is the step-size factor, max_val is the maximum value obtainable on either the x- or y-coordinate, and N_{state} is the number of states. For the studies described here, we set f_{ss} to be 10 and the maximum value to be 10.

Simulation memory (or the time it takes for the simulation to become markovian) can be increased by reducing step-size. If a step is taken beyond the boundaries of the toy model system, the move is automatically rejected. Moves are accepted (energetically or thermally) according to the metropolis criteria (described in detail in Metropolis Criteria section)

MSM construction

We begin by briefly describing the steps involved in constructing a Markov state model (MSM) from molecular dynamics or Monte Carlo trajectories(*96-98*). The first step involves discretizing the simulation; the individual configurations representing trajectory snap-shots are clustered based on geometric similarity without consideration of their kinetic proximity. The resulting clusters are called "microstates". The frequencies of transitions within and between microstates are computed, populating a count matrix. Groups of kinetically related microstates-called macrostates—are then identified from the eigenvectors of the resulting transition matrix by Perron clustering(*90, 99*). The macrostates and transitions among them constitute a reduced complexity description of the longer time-scale dynamics of the original system.

Data reduction techniques

To mimic techniques used by other groups to manage simulation data, we employed two different techniques. One was to reduce the number of total discrete states by grouping states with fewer counts than a certain threshold with the closest discrete state (by euclidean distance between grid center-points). The second technique was devised to maintain correct kinetic relationships by grouping sparsely populated states with the most kinetically-connected neighboring state. The state j counts are lumped with that of the state i, which it occupied before transitioning into state j (if a transition is described by i → j). The top n rates are lumped at once in order to produce the (Nstates – n) desired lumped states.

Results

Because the free-energy landscape is fully known, one can assign discrete states based on position (No clustering need be performed). Assignment is performed by drawing a square grid with spacings of 0.25. Future interesting directions might include the effects of approximate clustering algorithms have on the discretization error and the effects of current accepted clustering practices on MSM construction and interpretation. MSM construction follows the usual way, constructing a transition-count matrix from the trajectory described by transitions between discrete states. Count-matrices are symmetrized in accordance with current practices, and serves to increase numerical robustness. (comparisons with results obtained with un-symmetrized matrices showed little difference, data not shown).

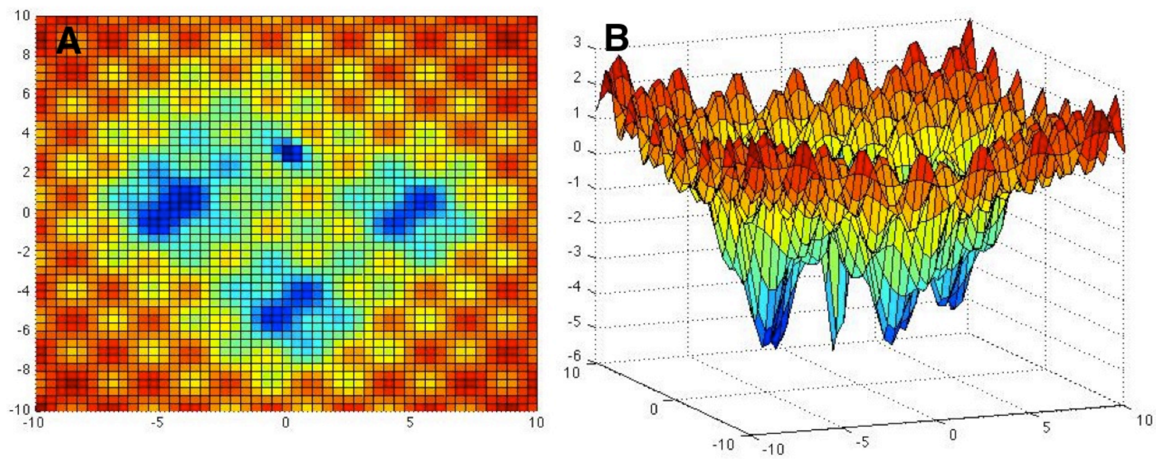


Figure 16. Top view (A) and Side view (B) of the toy model free-energy landscape used in the toy model studies. Red indicates higher free-energy and blue indicates lowest free energy. The free-energy (color) is plotted as a function of the X and Y coordinates (A and B). In the side view (B), the free-energy is plotted as a function of X and Y.

Well-no	**Well center (x,y)**	**Well-sigma**	**Well weight**
1	-5, 0	2	4
2	5, 0	2.67	4
3	0, 3	0.2	4.8
4	0, -5	3.33	4
5	-3, 3	6.67	2

Table VI. Well parameters for toy model shown in Figure 16

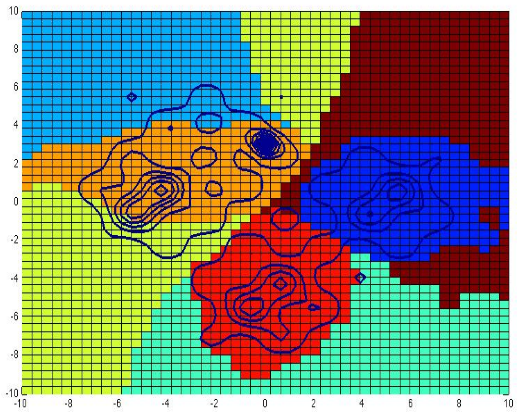


Figure 17. Six macrostate definition obtained using Perron clustering for the toy model landscape shown in Figure 16.

Toy Model Definition

The toy model free-energy landscape is shown in Figure 16A (top-level view) and Figure 16B (sideview). The x- and y- values ranged from [-10, 10], and well parameters are shown in Table VI. The harmonic constant, k, was set to 100, and the ruggedness factor was set to 1, rugged-frequency set to 2. Sampling was performed with beta(or k_bT) = 0.5. Monte-carlo sampling was performed for several millions steps until convergence, and all analysis was performed at a lag-time of 200 steps, according to the convergence of the implied-timescales. Macrostates are defined as described previously(*99*), and an example of a six state macrostate model, with contours drawn to indicate well-positions, is shown in Figure 17. As expected, three of the macrostates correspond to the positions of the wells; two wells which are kinetically connected are grouped in a combine macrostate. The rest of the macrostates define regions of the unfolded state which, due to diffusion limits, are considered kinetically related.

Geometry-based methods of state reduction can obscure important kinetic relationships

We tested eliminating the least occupied 10%, 25%, and 50% states (Figure 18) and discovered that with minimal data-reduction, the macrostate relationships are not significantly altered (Figure 18 B). However, as more data is removed, macrostate definitions become significantly altered compared to the macrostate definitions obtained without data-reduction techniques (Compare Figure 18A with Figure 18 C and D). In particular, the minima at (-5,0) and (0,3) are incorrectly partitioned into three different regions of the unfolded state (Figure 18 Panel D).

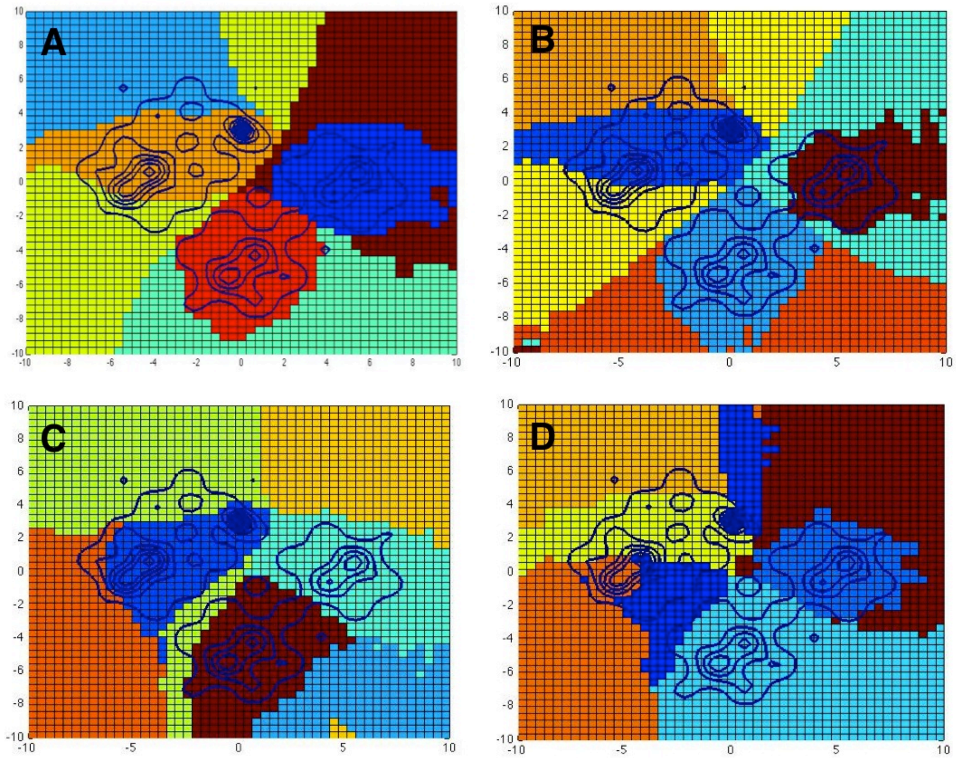


Figure 18 Lumping by geometry does not preserve kinetic relationships

A no data removed B 90% data C 74% data D 50% data

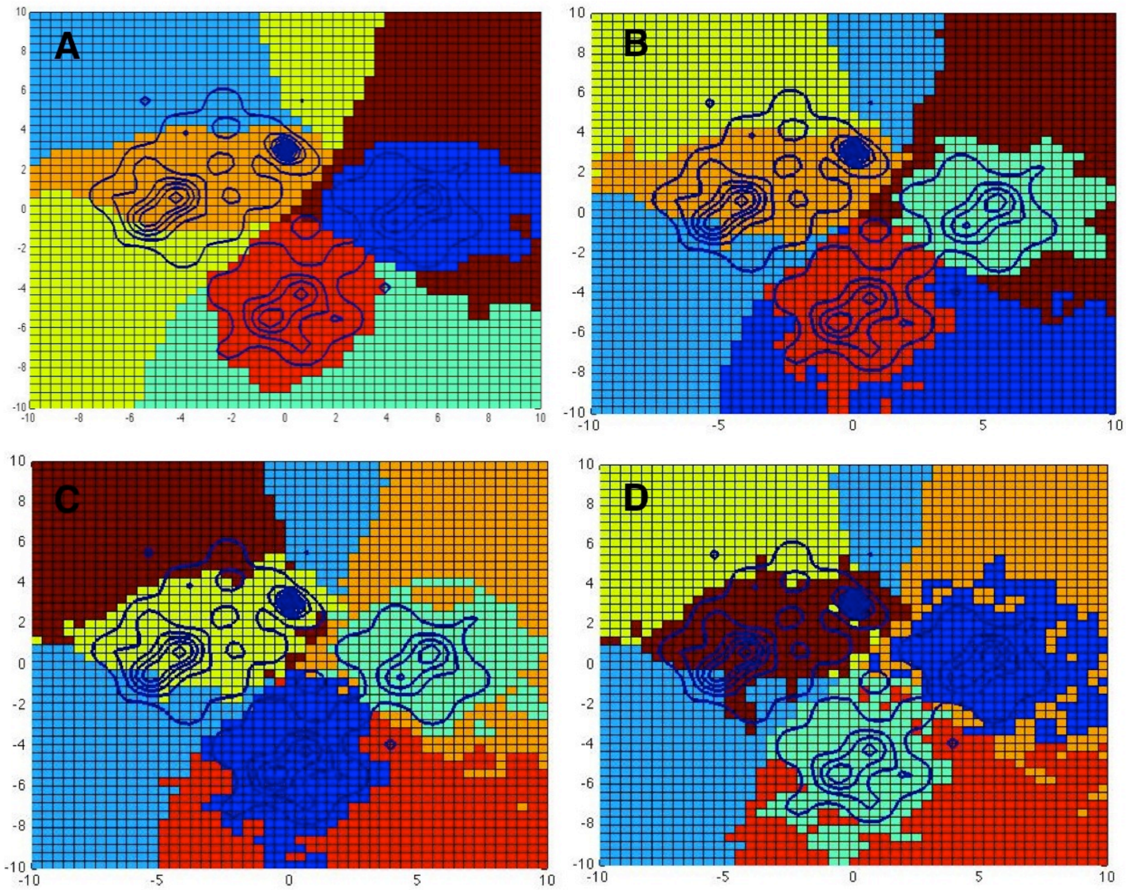


Figure 19 Lumping by kinetic-rates preserves kinetic relationships

A no data removed B 90% data C 74% data D 50% dat

Kinetic-based methods of state reduction preserves kinetic relationships and MSM accuracy

In order to better preserve the kinetic relationships between states, we hypothesized that reducing the number of states by combining those with the highest kinetic rates would better preserve kinetic relationships between states. For each state with few counts, we combined the counts with the state which it transitioned from the most. In other words, we combined state j with the state i for which the value Tij is the highest (where Tij is the transition probability describing the transition from i to j). Reducing the number of states by 10%, 25%, or 50% using this technique better preserved the essential kinetic relationships in the resulting macrostate definitions (Figure 19). In contrast to the results obtained from geometric lumping, removing 50% of the states (Figure 19D) resulted in similar macrostate definitions as that obtained with no state elimination (Figure 19A). Interestingly, even unfolded state macrostate definitions remained similar to the reference state definitions (no states eliminated), again in contrast to results obtained using geometric state-reduction techniques in which the unfolded state macrostate definitions changed dramatically (Compare Figure 18 with Figure 19).

Sparseness is an issue in any application using MSM construction, but little attention has been paid to dealing with this sparseness in a rigorous way. We show here that geometry-based methods of decreasing sparseness can lead to altered MSM definitions, whereas kinetic-based methods preserve them. Future work will include further development of rules for reducing the number of sparse states as well as extension of this work to analyze real systems.

Effect of biased sampling on MSM construction

We performed two simulations, one in which sampling was seeded uniformly on each grid-point, and another in which sampling was seeded in a biased configuration. Biased starting points are shown in Figure 20A, along with the ending population counts in Figure 20B. Sampling was performed in the same way described in the methods section titled: “Sampling”. For comparison, the log(counts) are shown for the uniform sampling run (Figure 21 left panel) and for the biased sampling run (Figure 21 right panel).

We constructed an implied-timescale plot, a common validation technique, but found that the implied-timescales converged well for both uniform and biased runs(Figure 22). This is not surprising, as biased sampling does not preclude markovian behavior, and this test is not designed to detect effects of biased sampling. When we construct the Markov State Model, we find that the kinetic relationships are significantly altered; none of the minima occupy their own macrostate, as observed in the case of uniform sampling (Figure 17). Instead, the macrostate definitions mimic the biased sampling of the inputs, forming a cross-shape (Figure 23 A and B). Constructed flux diagrams confirm that kinetic relationships between macrostate are also significantly altered and often contain incorrect kinetic connections(data not shown). Thus, a biased choice of starting structures can significantly influence the resulting Markov State Model to the point of altering macrostate definitions as well as altering the kinetic relationships between macrostates. While biased choice of starting structure is inevitable in the application of MSM construction on real systems, these findings indicate that detection of biasing effects as well as ways to eliminate them are needed in order to construct as accurate a model as possible.

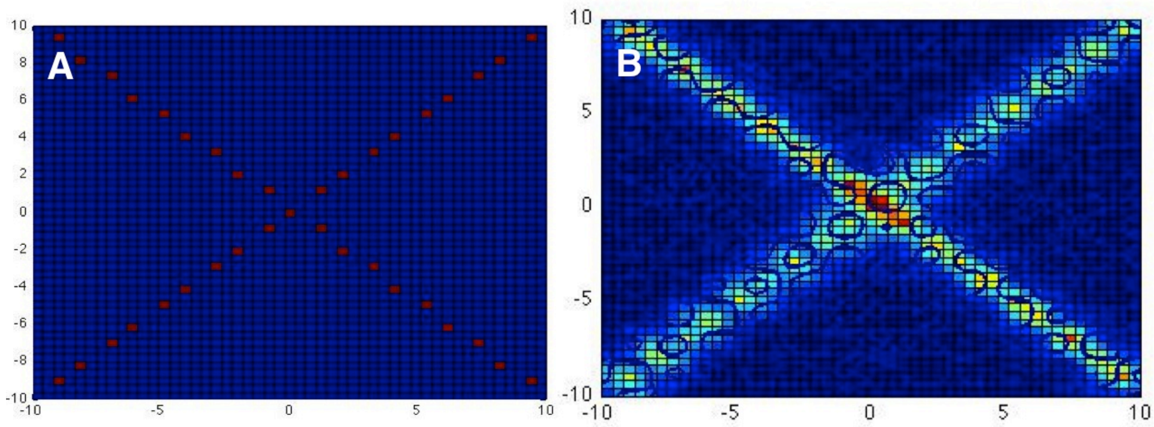


Figure 20. The starting points (red dots) on the free-energy landscape for the biased simulation study are shown in A. The ending counts are shown in B. Blue indicates lower number of counts whereas red indicates high numbers of counts.

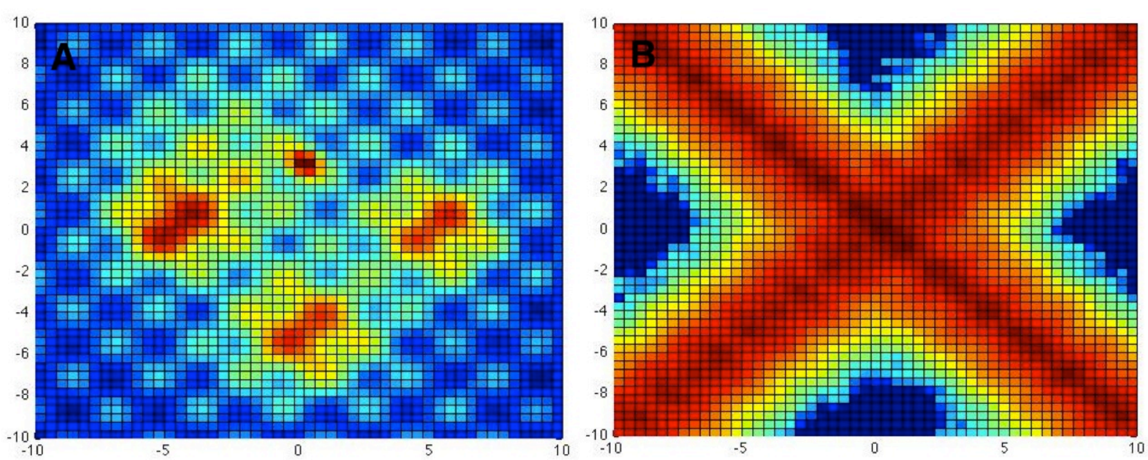


Figure 21. The ending log of the counts are shown for each state on the free-energy landscape for the (A) unbiased and (B) biased simulations. Blue indicates low numbers of counts whereas red indicates high numbers of counts.

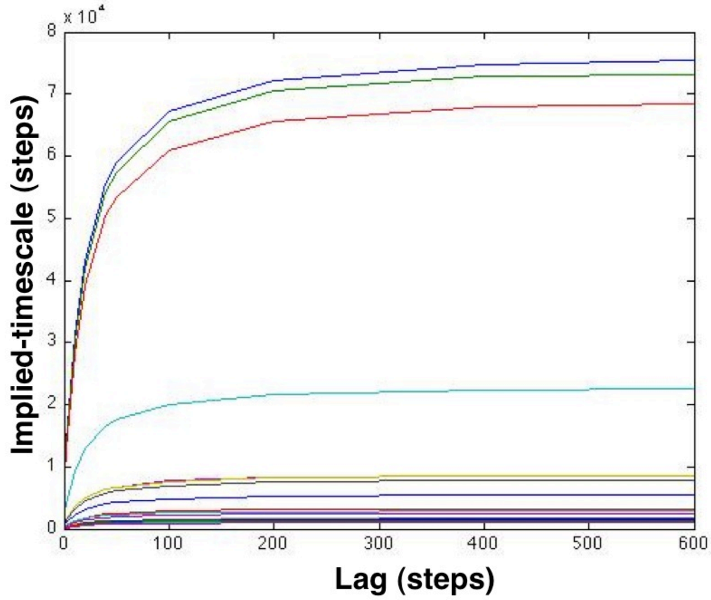


Figure 22. The implied-timescale plot for the biased simulation shows no indication of errors.

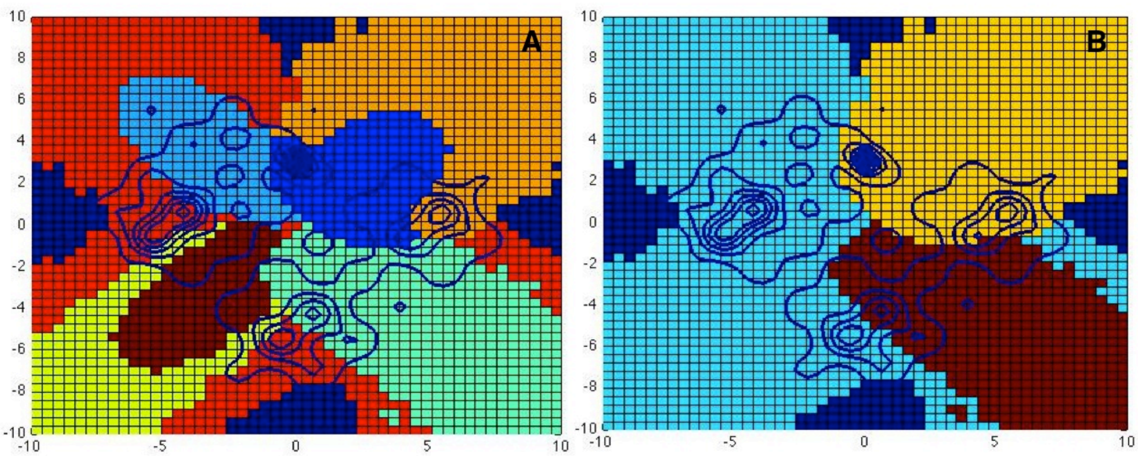


Figure 23. The macrostate definitions for (A) six and (B) four macrostates are shown for the biased simulation study. The kinetic relationships as well as the definitions themselves are altered relative to the original macrostate definitions (figure 17)

Conclusions

Toy models can be a powerful way to examine the effects of common assumptions on MSM construction. We discovered that commonly used techniques of reducing the number of states used to encode the free-energy landscape may lead to obscured kinetic relationships, and we suggest new techniques which serve to accomplish the same end but through preservation of the macrostate definitions and the kinetic relationships. We furthermore investigate the effects of using biased starting points without attempts to correct for undersampled regions and we discovered that both macrostate definitions as well as kinetic relationships between macrostates can be significantly altered. Development of novel techniques for MSM construction and validation may benefit from thorough testing and examination of methodological improvements on toy-model systems such as this one.

Applications of MSM construction and validation to Real Datasets

Although the toy-model systems are useful for studying how different MSM construction techniques affect the resulting interpretations, the ability to map the landscape without use of approximate clustering methods makes the system artificially simple. In order to investigate model construction techniques on real systems, we chose a system in which long timescale information is known to exist in the simulation; any MSM construction technique should be able to represent the long timescale information in the simulation, and validation can be performed easily simply by analyzing the simulation with previously established, simpler techniques. With such a gold-standard benchmark set, one could easily evaluate MSM construction validation tools as well as different MSM construction techniques.

Exciting breakthroughs in computer hardware have made possible simulation of proteins for time scales longer than the time required to fold, allowing observation of multiple folding and unfolding events(*100, 101*). Simulations spanning the ~us folding times of even the smallest proteins require trajectories of > 10^6 configurations, and methods for associating these configurations into a much smaller number of significant states are of considerable importance for analyzing the key structural transitions(*102, 103*). Traditional methods have required projection of simulation data onto one or two-dimensional reaction coordinates(*104-118*), but thus can obscure important features of the folding free-energy landscape(*74*).

Markov state models (MSMs) have overcome this limitation by representing dynamics as a network of transitions between discrete states and have been used to analyze folding-pathways for many proteins of interest(*92-94, 97, 98, 119-124*). A MSM is constructed by first using geometric similarity to combine very similar configurations into discrete microstates, assuming that high geometric similarity implies high kinetic connectivity. Subsequently, these microstates are assembled into sets of kinetically related microstates, called macrostates. Care must be taken in defining the initial set of states, as incorrect initial grouping of configurations can result in incorrect conclusions about folding dynamics if, for example, a microstate contains configurations separated by high-energy barriers(*125*).

There are many open questions in constructing MSMs of protein folding. In order to assign configurations from different trajectories to discrete states, some clustering based on geometric similarity is clearly necessary. What distance measure to use, to what extent configurations should be clustered, and how this impacts the resulting model are not fully understood. To address these and related problems, quantitative metrics for evaluating alternative models are needed(*126, 127*). Here we describe a likelihood measure for assessing alternative MSMs and investigate the trade-off between geometric and kinetic based lumping as well as alternative structural similarity metrics for grouping configurations.

Methods

Representations and distance measures

Secondary structure pairing

Hydrogen bonding patterns were identified and classified for each configuration using the Rosetta software(*11, 128-130*). The peptide conformation in each trajectory snapshot (configuration) was represented by a feature vector describing the sets of paired strands and their registers, pleatings, and orientations. Configurations with a common feature vector are assigned to the same microstate. Details of the feature descriptions and the assignment procedure are in the Appendix. Depending on the detail of the description the number of distinct feature vectors ranged from 175 to 4857.

Contact-map

Residues separated by more than one residue in sequence were considered in contact if the Cα residues were within 8 Å. Clustering of contact-maps was performed according to the greedy K-centers algorithm(*131*), using Euclidean distance to measure similarity. K-means refinement(*132*) was performed for twenty iterations using the greedy K-centers cluster assignments as input.

Rmsd

Cα coordinates were clustered using greedy K-centers clustering and root-mean-squared-deviation (rmsd) as the distance measure as described previously(*92, 133*).

MSM model construction

Transition matrices were constructed with a lag-time of 100 ns, as determined from convergence of implied-timescale plots(*97*) (Figure 24)(*17*). Macrostates were obtained from the eigenvectors of the transition matrices using Perron clustering(*90*) as described previously (see MSM construction section for more details).

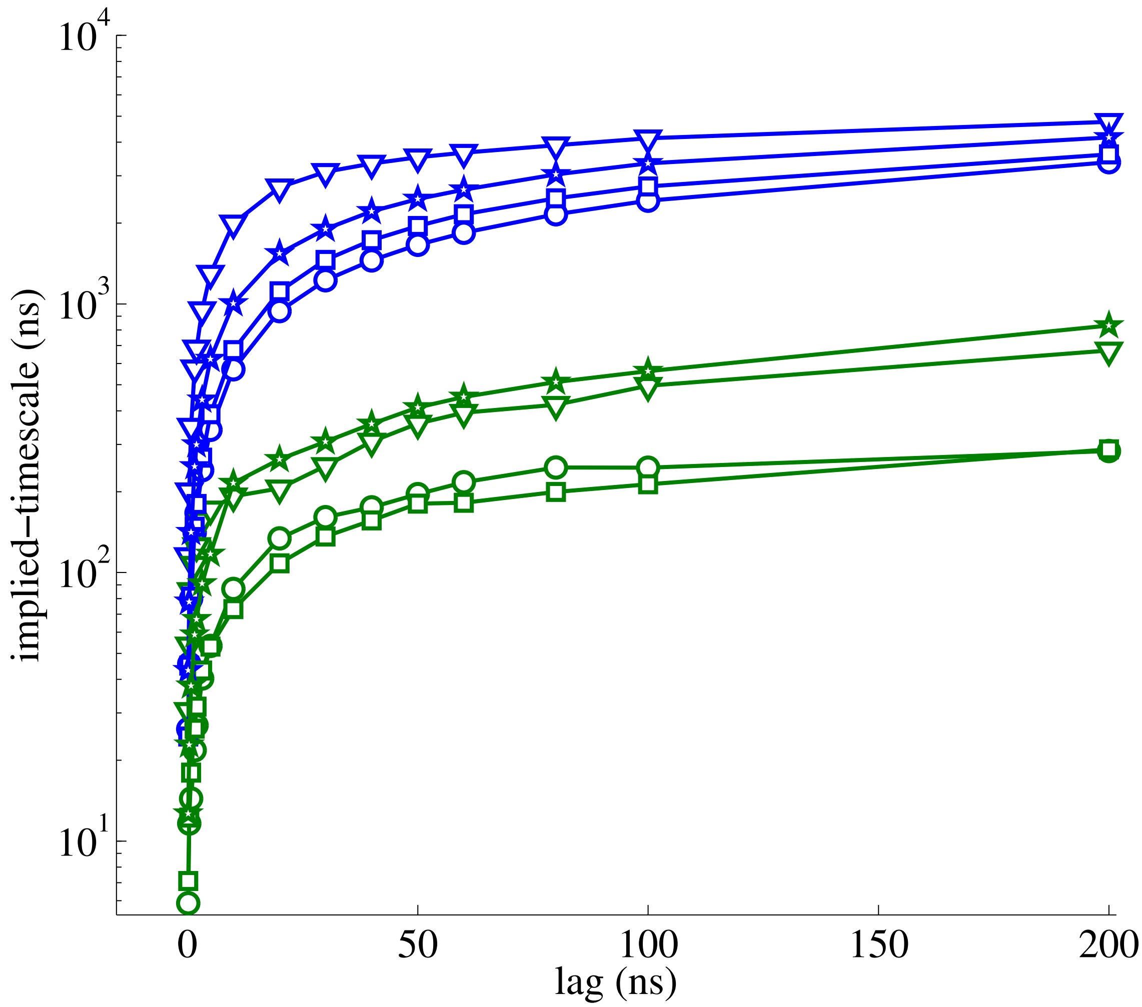


Figure 24: lag versus implied-timescale for the two slowest non-stationary eigenvectors (green and blue) for each of the different assignment methods: rmsd (circles, 10,000 microstates), Kmean contact-maps (triangles, 100 microstates), Kcenter contact-maps (squares, 1,000 microstates), and strand-pairing microstates (stars, 4,857 microstates).

Log-likelihood metric

We define the likelihood of a Markov model given a set of trajectories to be the probability of observing the trajectories given the model. Given a set of assignments of configurations $\{c_i\}$ to states s_i, the probability of a particular configuration trajectory is given by a two level Markov model:

$$p(\{c_i\}) = p(\{s_i\})p(\{c_i\}|\{s_i\}) = p(s_1)\prod_{i=1}^{N-1} p(s_{i+1}|s_i)\prod_{i=1}^{N} p(c_i|s_i)$$

where s_i is the state occupied at step i in the trajectory, $p(s_i)$ is the probability of state s_i, $p(s_{i+1}|s_i)$ is the probability of transitioning to state s_{i+1} from state s_i, and $p(c_i|s_i)$ is the probability of configuration c_i given that the system s in state s_i. $P(\{c_i\})$ is the probability of the configuration trajectory, $p(\{s_i\})$ the probability of the state trajectory, and $p(\{c_i\}|\{s_i\})$ the probability of the configuration trajectory given the state trajectory.

Computation of the likelihood requires estimation of the above quantities based on the training data. $p(s_1)$ is well modeled by the frequency of state 1, n_1/N_{total}, where n_1 is the number of configurations in state 1, and N_{total} the total number of configurations. $P(s_{i+1}|s_i)$ is well modeled by the transition-frequency from state i to state i+1 observed in the trajectories. A simple choice for $p(c_i|s_i)$ that follows from the assumption that configurations are uniformly distributed within the states is the inverse of the phase space volume spanned by state i. However, it is difficult to compare cluster volumes between the different structural representations (rmsd, contact-map, strand-pairings). Instead, we chose to estimate $p(c_i|s_i)$ as $1/n_i$, where n_i is the number of configurations assigned to state i. This choice has the obvious benefit that models in different representations can be readily compared, and the further advantage that the likelihood of the null model (see below) is independent of the number of states. However, the implicit assumption that configurations are uniformly distributed in phase space (so the volume becomes proportional to the number of configurations in the cluster) is clearly an oversimplification. Improving the estimate of p(ci|si) is an important area for future work.

We normalize by computing the likelihood of a null model in which all states have equal size and the probability of all transitions are equal. For the null model with m states and N_{total} total configurations, the probability of observing a configuration given a state is m/N_{total} and the probability of transitioning to any state is 1/m, so the probability at each step of the null-model trajectory is $(m/N_{total} * 1/m) = 1/N_{total}$ as it should be since all configurations are equally probable. To avoid round-off errors, we compute the log likelihood instead of the likelihood, and subtract the log-likelihood of the null model, yielding the final form of the log-likelihood shown in figures 25 and 26.

$$\log\left(p\left(traj \mid MSM\right)\right) = \log\left(\frac{n_1}{N_{tot}}\right) + \sum_{i=1}^{N_{test}-1} \log\left(T\left(i,i+1\right)\right) + \sum_{i=1}^{N_{test}} \log\left(\frac{N_{tot}}{n_i}\right)$$

Cross-validation procedure

We experimented with a number of ways to cross-validate Markov State Models using independent trajectory data. The most rigorous is to construct an MSM purely from the training set data, and then assign each test set configuration to a training set microstate based on geometric similarity. This turned out to be computationally intractable since obtaining good statistics required many repeated cross validation calculations with different randomly selected training set/ test set partitions, and it was not feasible to carry out the microstate clustering large numbers of times for each parameter set and representation considered. We settled on a considerably more tractable approach in which the entire dataset is clustered to obtain microstate definitions, but the transition matrix construction and subsequent spectral clustering to obtain macrostate definitions are based on training set data only. Microstates observed in the test set but not in the training set are reassigned to the closest microstate in the training set based on geometric similarity. The model quality metric, which we refer to throughout the rest of the paper, is the log-probability of observing the test set data given the model constructed from the training set data. Cross-validation was performed a total of 1000 times for each model. One randomly selected segment of the data comprising a total of 1% of the total data was removed prior to transition matrix construction and spectral clustering but after the initial geometric clustering step, and the 99% which remained was used to compute the transition matrix and state occupancies. To reduce sensitivity to the original clustering results for models with less than ten thousand microstates, the cross validation procedure was repeated for 4-10 different initial clusterings obtained using different random seeds.

Likelihoods of microstate models were computed using the microstate occupancies and microstate transition-probabilities computed from the training set. Similarly, likelihoods of macrostate models were computed using macrostate occupancies and macrostate transition matrices computed solely from the training set.

Results

The process of discretizing a simulation is a non-trivial task and involves a number of critical choices(*91*). If the clustering is too coarse (too few clusters), the resulting assignments will likely contain configurations separated by large kinetic barriers. On the other hand, if clustering is too fine (too many clusters), the resulting transition matrix will quickly become sparse, resulting in bad statistics and reduced generalizability. In the limit, each configuration is in its own microstate and the resulting model has no predictive power. In order to assess the choices made during assignment, a metric is needed for assessing the extent to which the resulting MSM accurately represents the dynamics of the system.

We have found that a simple likelihood statistic coupled with cross-validation provides a very useful model-quality metric (see methods: Log-likelihood metric). Given choice of distance metric and geometric clustering threshold (number of microstates), we partition the data into training and testing sets and compute the log-probability of observing the test set data given the training set data using transition matrices compiled exclusively from the training set (see methods: Cross-validation).

To investigate the utility of the log-likelihood statistic to guide MSM construction and evaluation, we used the RMSD based clustering approach pioneered by the Pande group(*90, 93, 99, 133, 134*) and others(*125, 135, 136*), to build MSMs from the 200usec MD trajectories of the WW domain from the DE Shaw group(*100*). We compared the likelihood of the microstate based models resulting from the initial geometric clustering step to the likelihood of macrostate models produced by lumping microstates together using Perron clustering. Without cross validation, the likelihoods of both the microstate and macrostate models increase monotonically with increasing numbers of microstates (triangles, fig 25). With cross-validation, the likelihood decreases sharply for higher number of microstates as the predictive power of the model (generalizability) decreases (circles, fig 25). The macrostate models retain high likelihoods indicating greater generalizability even with higher numbers of microstates (compare open and closed circles, fig 25). This result reinforces the idea, central to the motivation for MSMs, that grouping based on kinetic connectivity is superior to clustering based solely only on geometric similarity.

The cross validated likelihood statistic provides a metric for evaluating alternative ways to construct macrostate MSMs. Given a fixed number of final macrostates, the coarseness of the initial geometric clustering is controlled by varying the number of microstates. With increasing number of microstates, the geometric lumping is less coarse, and correspondingly, the final macrostate composition is more dominated by kinetic connectivity. As shown in table VII, all macrostate models have higher log-likelihoods than the corresponding microstate model with the same final number of states (table VII, "rmsd"). The cross validated log-likelihood of the RMSD based macrostate model reaches a maximum between 1,000 and 10,000 microstates, which correspond to 6.5 Å and 5.3 Å rmsd cluster-radii, respectively (fig 25, open circles). In practice, for such a small protein system, configurations with significant differences (i.e. strand register shifts) can have RMSDs much less than 5.3 Å rmsd and hence be assigned to the same microstate even though this violates the assumption that geometric similarity implies kinetic similarity. RMSD based clustering thresholds below 3Å may well be necessary to preserved kinetic relationships, but partitioning the data this finely would result in orders of magnitude more microstates and a poorly determined transition matrix, decreasing the generalizability of the model (this may be responsible for the small decrease in likelihood for 50,000 microstates evident in Figure 25).

representation	no. initial states	no. final states	log-likelihood
rmsd	20	20	0.21
	100	20	0.43
	1000	20	0.53
	10000	20	0.55
secondary-structure	175	20	0.68
	175	100	0.56
	175	175	0.54
K-centers contact-map	20	20	0.05
	100	20	0.25
	1000	20	0.52
	10000	20	0.31
K-means contact-map	20	20	0.84
	100	20	0.68
	1000	20	0.63
	10000	20	0.36

Table VII. Trade-off between kinetic versus geometric lumping as measured by log-likelihood. For each representation (column one), the log-likelihood (column three) is measured as a function of the number of initial geometrically defined states (column two) and the number of kinetically clustered final states (column three). If the number of initial states is equal to the number of final states, the model was constructed purely using geometric-clustering. The more initial states, the finer the partitioning of space before the kinetic-clustering step (number of final states, column 3).

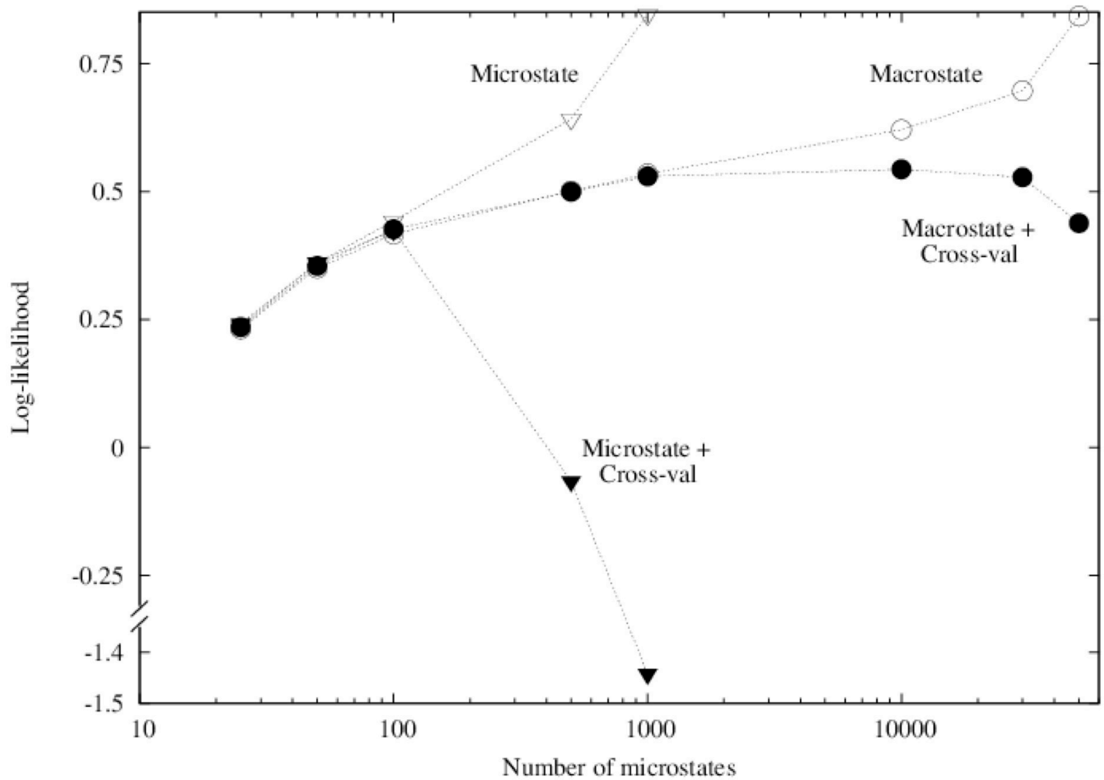


Figure 25. Assessment of RMSD based MSMs using the likelihood metric. Triangles, microstate models; circles, macrostate models. Open symbols represent training set likelihoods, closed symbols, test set (cross-validated) likelihoods. The number of microstates is indicated on the x axis; the clustering radius decreases from 8.1Å for the 20 microstate model to 4.4Å for the $5x10^4$ microstate model. For small numbers of microstates all models have similar likelihoods. For more than 100 microstates, the cross validated likelihood drops steeply but is rescued by lumping of the microstates into macrostates. To generate the macrostate models, microstates were lumped into 20 macrostates using Perron clustering.

We reasoned that more economical models might be obtainable using geometric similarity measures other than RMSD to group configurations into microstates. With the aim of building discrete state models for much larger systems, we first considered a variety of reduced representation in which configurations are described by their secondary structure pairings (see Methods section). With orders of magnitude less initial states, the resulting macrostate models (same number of final states) had significantly higher likelihoods (table VII, “ss-pair”) than the corresponding RMSD based models (table VII rmsd). The secondary structure pairing representation captures key features of the dynamics and thus provides more kinetically relevant microstate definitions--this is not surprising as previous descriptions of WW domain folding have focused on the formation of the two hairpin structures(*36, 92, 100, 137*).

While the secondary structure based models appear to be more economical in representing the dynamics, they are unable to describe contributions from more general interactions, like hydrophobic core formation(*128*). On the other hand, RMSD-based models are more general and the resolution of clustering is easily controlled, but kinetically irrelevant structural elements such as flexible loops and termini may contribute significantly to state-assignment whereas formation of hydrogen bonds might be missed, resulting in the need for large numbers of states to accurately describe the dynamics. To combine the advantages of the secondary structure and rmsd-based representations, we explored a contact-map based representation(see Methods). We reasoned this representation would capture strand pairings and registers, like the secondary structure model, while remaining general by also capturing important long-range contacts not necessarily involving hydrogen bonding. At the same time, such a representation would be less sensitive to loop and termini fluctuations than rmsd-based models.

We experimented with two clustering methods for grouping configurations into microstates using the contact-map based representation. The likelihoods of models constructed from microstates obtained using the very fast greedy K-centers method (table VII), were similar to those of the RMSD based models, but not as high as the secondary structure pairing models. We reasoned that more accurate clustering of configurations into microstates could produce better models, and took advantage of the fact that the contact map representation allows a simple definition of the cluster center --the average of all the contact maps in a cluster—and used K-means optimization to refine the cluster boundaries (table VII, figure 26A). The likelihoods of models produced using K-means clustering were considerably higher than models created with K-centers clustering (table VII). Improved state definitions as expected yield more predictive models.

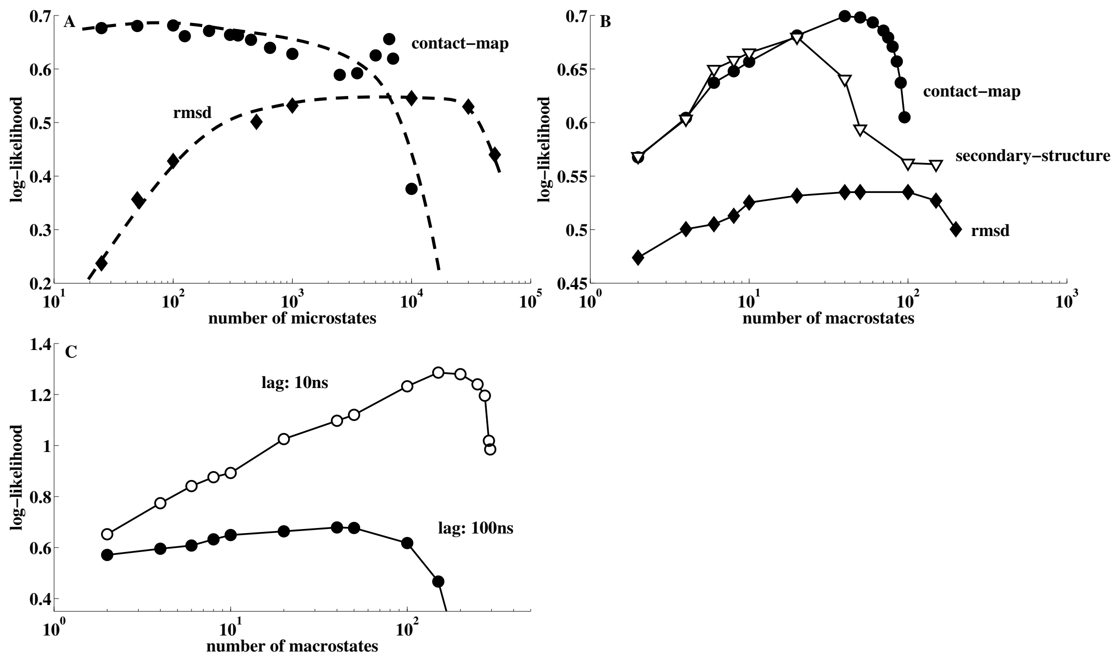


Figure 26. Evaluation of MSMs using the likelihood metric. A) Dependence of likelihood on the number of microstates. Configurations were first grouped into the indicated number of microstates and these were then further lumped into 20 macrostates using Perron clustering. Diamonds, RMSD based microstate assignments; circles, K-means contact map based assignments. Lines are manually drawn to guide the eye. B) Dependence of likelihood on number of macrostates. For each representation, models were constructed using the number of microstates producing the highest likelihood in panel A (diamonds, rmsd 1,000; circles, K-means contact-map 100, and triangles, secondary structure-pairing 175). C) More states are required to model short time scale dynamics. For the K-means contact-map representation with 300 microstates, the dependence of the log-likelihood on the number of macrostates is shown for

a lag-time of 100 ns (filled triangles) and 10 ns (open circles). The optimal number of macrostates is higher at shorter lag times.

The optimal number of microstates differs considerably in the different representations. For the RMSD and K-centers contact map representations, the highest likelihoods are obtained for models with 1000-10,000 microstates. In contrast, for the secondary structure pairing and K-means refined contact-map microstate definitions, the optimal number of microstates is between 100 and 200, and the initial partition into microstates represents the dynamics better for the latter models than the former (compare table VII "rmsd" and "K-means contact-map"; the log-likelihoods of the K-means contact-map microstate models are similar to those of the K-means contact-map macrostate models with the same number of states). The large numbers of microstates for the best RMSD-based models is likely necessary to avoid excessively large cluster radii (~8Å, in the 20-microstate model) (see Fig 25 legend and table VII). The contact map and secondary structure pairing models have significantly higher likelihoods while requiring many fewer microstates probably because clustering in these representations better preserves kinetic connectivity. Improved clustering methods can yield significantly higher likelihood models as illustrated by the difference in likelihood of the contact map K-centers and K-means based models. Thus, improved methods of RMSD-based cluster refinement(*99*) could perhaps yield significantly higher likelihood models than the RMSD models constructed in this study. The relatively small number of microstates required to build a predictive model using the contact map and secondary structure pairing representations (100 – 300) suggests that building accurate models of the folding of larger proteins with these approaches should be feasible.

We next considered the dependence of the likelihood on the number of macrostates. We anticipated that the log-likelihood would initially increase with number of macrostates due to the improved representation of folding dynamics but subsequently decrease due to over-fitting to the training data. This was indeed observed. As shown in Figure 26B (circles), the likelihood peaks at 40 macrostates for K-means-refined contact assignments, 40-100 for rmsd assignments and 20 macrostates for secondary structure pairing based assignments. These results suggest that the most predictive, and hence in a sense most accurate model of WW domain folding involves less than one hundred discrete states. To further investigate what effect the lag-time has on the optimal number of macrostates, we decreased the lag-time from 100 ns to 10 ns (figure 26C). The maximum in the log-likelihood shifts to higher numbers of macrostates as expected since additional states are required to model the short time dynamics of the system.

To obtain an intuitive feeling for the usefulness of the different models in providing descriptions of the folding kinetics, we compared the descriptions of the folding transitions observed in the MD simulations provided by the contact-map, rmsd, and secondary structure based models. For reference, we computed for each folding-transition the rmsd-to-native for the global structure, the first hairpin, and the second hairpin(*100, 101*), and compared this to a contact map based microstate trajectory and macrostate trajectories produced by the highest log-likelihood rmsd, contact-map, and secondary structure pairing models (Figure 27). In the contact map representation, intermediates with either the first hairpin or the second hairpin are formed in different macrostates (Fig 28), and the formation of the first hairpin and the second hairpin is clearly evident in the contact-map macrostate trajectory (Figure 27 second-row color bar). The contact map based microstate trajectory is very similar to the macrostate trajectory (compare top color-row with rows 1-2) as expected given its similar log likelihood (table VII). The secondary structure pairing based MSM trajectory (fig 27, third color bar) yields similar qualitative results, but assigns a larger portion of the unfolded state to a single macrostate. The rmsd MSM trajectory, constructed using the maximum log-likelihood microstate model of 1,000 microstates, does not clearly identify either of the intermediates with just 20 macrostates (fig 27, fourth color bar), although a previous study identified these intermediates in a 200 macrostate model and approximately 26,000 microstates(*97*). The full set of comparisons are displayed in the appendix (Appendix figure 1).

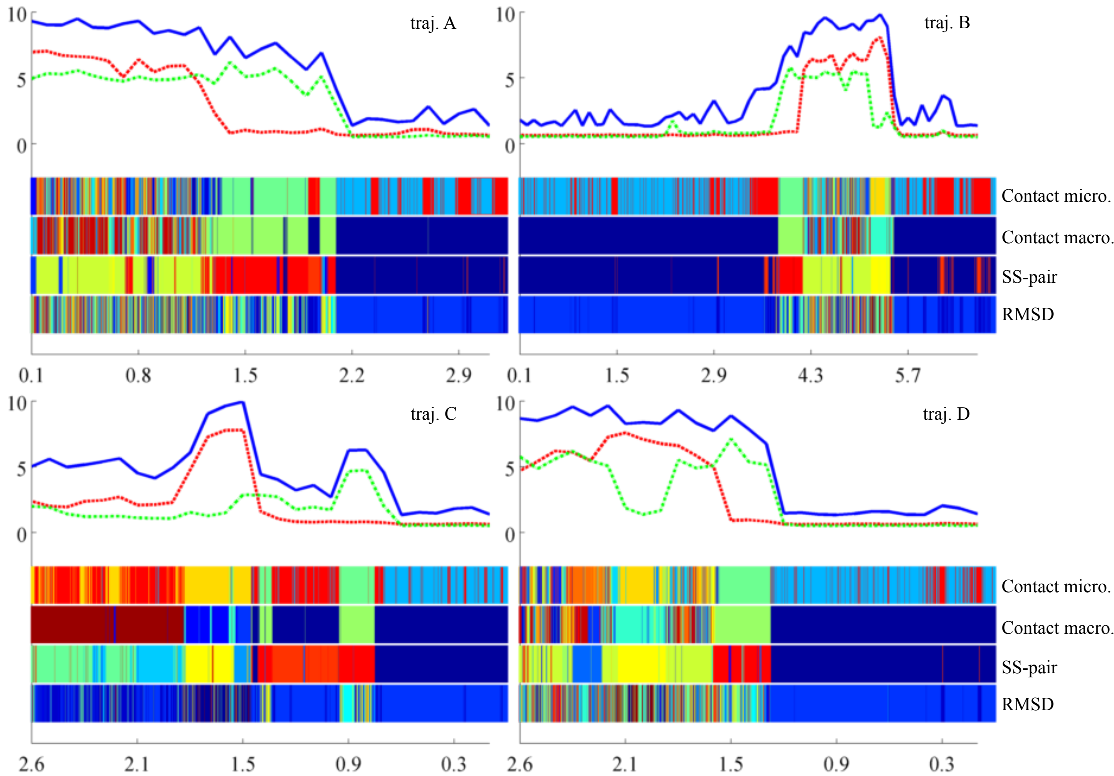


Figure 27 Comparison of microstate and macrostate trajectories in the three representations. The panels show portions of trajectories bracketing four folding/unfolding transitions. Upper-portion of panel Top panel: blue- rmsd to native structure over residues 1-34; red, rmsd to native over strand one, and green, rmsd over strand 2. Lower-portion of panels: macrostate trajectories for (first row) a twenty-five state K-means refined contact-map microstate model, (second row) a twenty macrostate contact map based model (100 microstates), (third row) a 20 macrostate secondary structure-pairing based model (175 microstates) and (fourth row) a 20 macrostate rmsd based model (1000 microstates). Each color represents a different macrostate.

The contact-map and strand macrostate models reveal the existence of a ‘kinetic-trap’ state, which has a global rmsd to native of approximately 5 Å (fig 27C color bar rows 1-3). The rmsd macrostate model, constructed from 1,000 microstates, does not detect this state, instead it assigns this state to the native macrostate (fig27C, bottom color-bar). Further analysis of this state shows that it persists for approximately 2 microseconds(data not shown), and consists of a shifted strand-one register but a correct strand-two register (appendix fig 2b). The contact-map and secondary structure-pairing based macrostate models evidently identify meta-stable states that cannot easily be detected using rmsd metrics.

We next considered how to obtain a comprehensive overview of WW domain folding from the contact map based macrostate model. Previous work with MSMs utilized overall flux diagrams showing the flow of probability density along the model. We constructed such a model (appendix figure 3) built from the entire set of simulation data, which is similar to that reported previously for the WW domain based on the same simulation data by Pande and coworkers with multiple paths to the native state(*97*). A flux diagram focused on the transition regions (Figure 28B) had fewer connections to the native state (4 instead of 8) which is surprising given that this subset of the data includes all transitions between unfolded and native states.

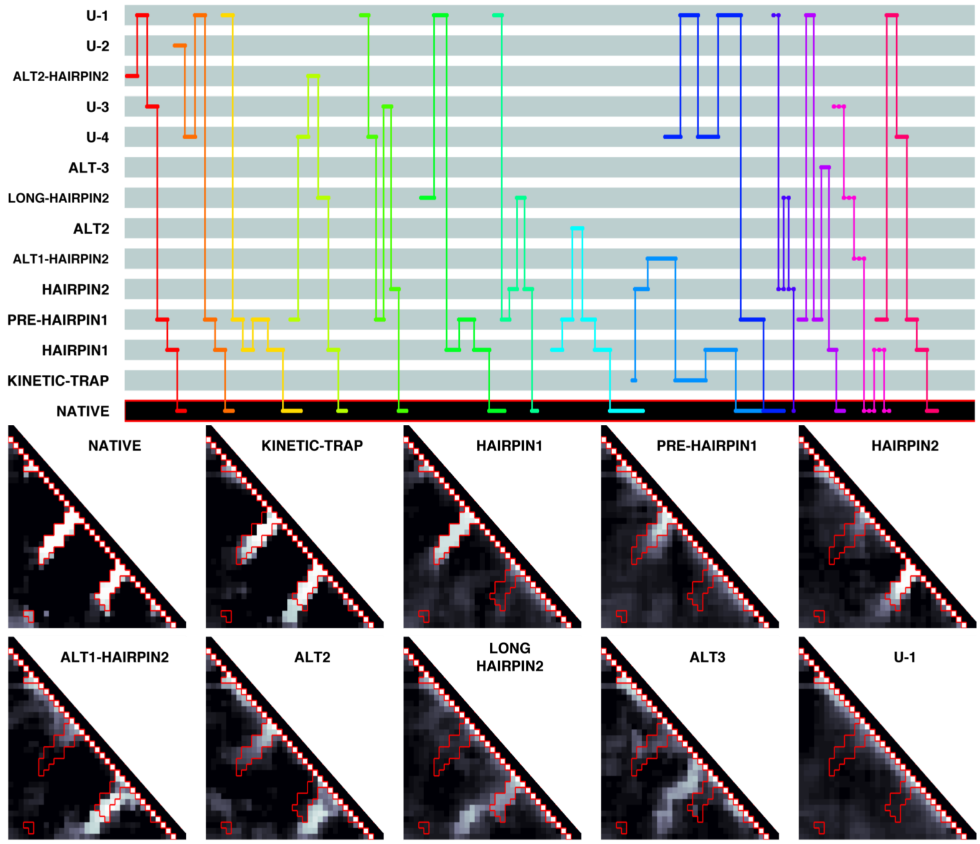


Figure 28A) Analysis of folding transitions. Discrete state representation of the fourteen folding/unfolding events in the long time scale WW domain folding simulations. In the top panel, each color represents one of the 14 unfolding-folding transitions, and the lines depict the sequence of states visited enroute to the native state. The lower panels show contact maps of the most commonly visited states in these folding transitions. While there is considerable heterogeneity early on, in most of the trajectories the "hairpin1" state immediately preceeds the transition to the native state. ALT abbreviations denote alternative structures which are non-native, ALT-HAIRPIN describe alternative macrostates which are similar to either native hairpin. U abbreviations denote macrostates with little to no regular structure.

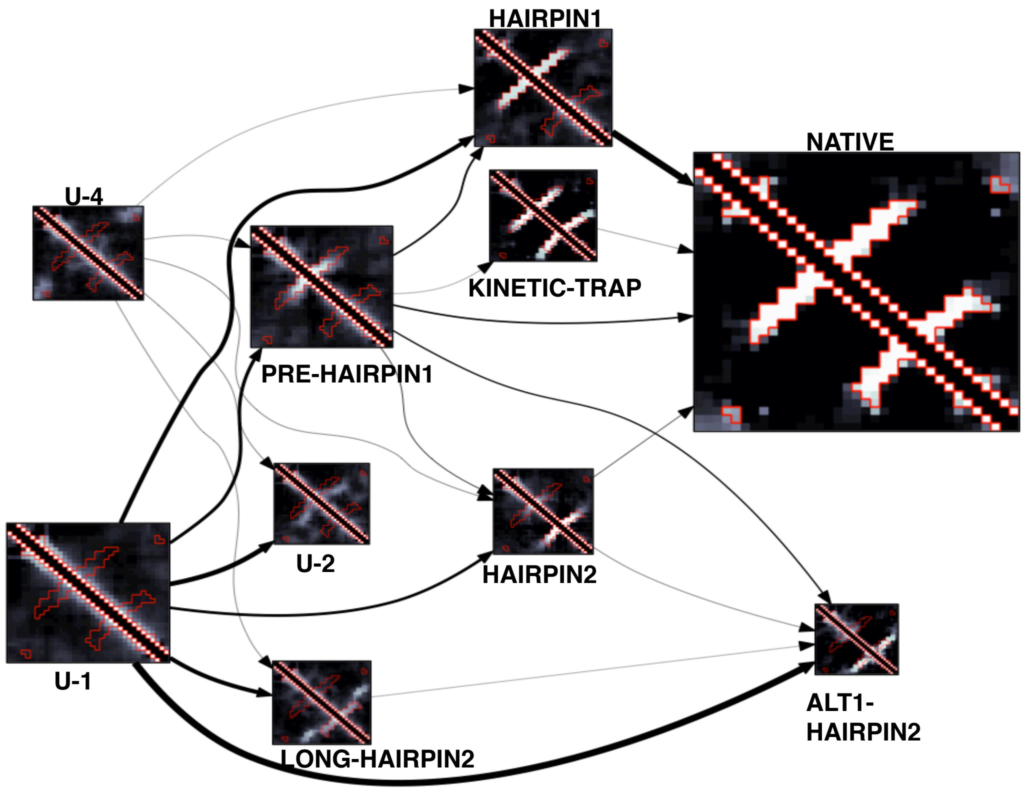


Figure 28B) Network representation (flux) of macrostate model constructed only from folding-transition regions. Model shown is for the 100 microstate K-means-refined contact-map microstate model which is kinetically lumped into a 20 macrostate model (constructed from a 10 ns lag-time). The individual macrostate names are the same as in Figure 4A.

To investigate this further, we examined the individual folding/unfolding transitions in the context of the contact map based model. Figure 4 exhibits the 14 folding/unfolding transitions observed in the trajectories in the contact map based models (see Kellogg et al.(*138*) for more details). We find that many states with residual structure are traversed before folding to native (figure 28A). These states contain some helix or strand secondary structure and many are compact (appendix figure 2). One state corresponds to the extension of the first hairpin to pair with the N and C-termini of the protein (appendix figure 2). Other states correspond to altogether non-native strand-pairings (appendix figure 2 F,G,I,K, and L). In accordance with this variety of non-native alternative states, we find that each non-native alternative is visited only once or twice. For ten of the fourteen transitions (figure 28A), we find that folding dominantly proceeds through the first hairpin and for four of the pathways, folding proceeds through formation of a second hairpin intermediate.

Whereas the flux diagram represents a prediction of the folding dynamics based on the MSM transition matrix and computed p-fold values(*135*), the folding-pathway reconstruction only uses the state assignment of the MSM model for analysis and otherwise relies solely on the actually observed transitions in the trajectories. While the conclusions drawn from the flux-diagram and the folding-pathway reconstruction are qualitatively similar (folding proceeds along a single dominant route with the formation of hairpin one), small discrepancies (the kinetic-trap state is kinetically related to the native state in the flux-diagram but not the folding-pathway reconstruction) remain. These discrepancies may arise because sampling is too limited, even with the ground-breaking computing resources used for the simulations(*100*), to put meaningful statistical weights on the different folding pathways or because of errors in MSM construction resulting from microstate misassignments (grouping of configurations separated by large kinetic barriers in the same microstate).

Discussion

We show that the likelihood of an independent test set is a powerful metric for assessing alternative discrete state models of protein folding based on molecular dynamics simulations. As suggested by previous work with MSMs, we find that, when using RMSD, grouping configurations based on their kinetic connectivity is considerably more effective than grouping based on geometric similarity to obtain microstates. The highest likelihood models require on the order of 1,000 to 10,000 microstates, in the range used in previous studies. We find that contact-map and secondary structure-pairing representations are considerably more economical, achieving higher likelihoods with many fewer microstates (100 and 175, respectively). Furthermore, the K-means refined contact-maps and strand-pairing representations are more effective at preserving kinetic connectivity and identify intermediates clearly.

In this paper we used the very long MD simulations of WW domain folding to evaluate our methodological developments. Using the improved combination of contact-map and clustering procedure or strand-pairing based representations we identified a kinetic-trap state, which is clearly metastable and is not easily detected by either simpler rmsd-based metrics(*100*) nor rmsd-based macrostate models(*97*). Comparing individual folding-pathways, we find considerable heterogeneity at the beginning of the transitions, but convergence in the late stages of folding, with the majority of the folding transitions involving formation of the first hairpin as suggested in the initial study(*100*). Still more sampling (longer MD simulations) would be required to assign statistical weights to the different observed folding transition pathways.

The methods described here should be generally useful for building discrete state models of folding dynamics from long time scale molecular dynamics simulations. The likelihood measure provides a means to assess alternative model formulations, and the combination of contact map representation and cluster refinement strategy should scale considerably better than RMSD based clustering to larger systems. It will be particularly interesting to use the approach outlined here to build discrete state models based on the long time scale simulations recently reported for a number of larger proteins by Shaw and coworkers(*101*)—we anticipate these will provide more new insight than the model obtained here for the very simple WW domain.

Appendix

ΔΔG Appendix

I. ΔΔG weight optimization command-lines

ΔΔG-only weight optimization

optE_parallel.linuxgccrelease -s empty.lst -optE:no_design -optE:optimize_starting_free_weights true -ex1 -ex2 -linmem_ig 10 -optE:free free_wts.txt -optE:fixed fixed_wts.txt -optE:optimize_ddGmutation in.score -no_optH true -fa_max_dis 9.0 -in:file:silent_struct_type binary -skip_set_reasonable_fold_tree -mpi_weight_minimization -optE:constrain_weights nonegLJrep_pos_hbonds.txt -optE:no_hb_env_dependence

The input files are as follows: the sequence recovery file: empty.lst is empty, in.score specifies the location of the wild-type structures, the mutant-structures, and the experimental ΔΔG. The files free_wts.txt and fixed_wts.txt specify the weights allowed to vary and the weights to be kept fixed, respectively.

Sequence-only weight optimization

optE_parallel.linuxgccrelease -s input_list -optE::optimize_nat_aa -ex1 -ex2 -linmem_ig 10 -optE:free free_wts.txt -optE:fixed fixed_wts.txt -no_optH true -skip_set_reasonable_fold_tree -optimize_starting_free_weights -mpi_weight_minimization -optE:constrain_weights nonegLJrep_pos_hbonds.txt -fa_max_dis 9.0 -optE:fit_reference_energies_to_aa_profile_recovery true -optE::no_hb_env_dependence

In order to perform sequence recovery, a list of protein structures in pdb file format are specified by the –s option.

ΔΔG plus sequence recovery

ΔΔG data was upweighted by a factor of 5 relative to the sequence recovery data.

optE_parallel.linuxgccrelease -s tyr_r500_take3.list -optE::optimize_nat_aa -ex1 -ex2 -linmem_ig 10 -optE:free free_wts.txt -optE:fixed fixed_wts.txt -no_optH true -skip_set_reasonable_fold_tree -optimize_starting_free_weights -mpi_weight_minimization -optE:constrain_weights nonegLJrep_pos_hbonds.txt -fa_max_dis 9.0 -optE:fit_reference_energies_to_aa_profile_recovery true -optE::no_hb_env_dependence -optE:optimize_ddGmutation in.score -optE:component_weights upweight_ddg.txt -in:file:silent_struct_type binary

II. Structure optimization protocols

Command-lines for the protocols tested:

All results were produced with revision 32231 of rosetta, and revision 32257 of the rosetta database.

The following command-line options were used for all methods:

-in:file:s <INPUT_PDB> -resfile <RESFILE> -database <DATABASE> -ignore_unrecognized_res –in:file:fullatom –constraints::cst_file <CONSTRAINTS_FILE>

Specifying scoring functions:

To use the soft-rep scoring function during sidechain repacking: -score:weights soft_rep_design

To use the hard-rep scoring function during sidechain repacking: -score:weights standard –score:patch score12

The hard-rep scoring function is used by default during minimization. To change the minimization weights: -ddg::minimization_scorefunction standard –ddg::minimization_patch score12

Sidechain repacking:

Sidechains conformations were repacked using Metropolis Monte-Carlo simulated annealing(*9*), according to the backbone-dependent rotamer library of Dunbrack and Cohen(*139*). c1 and c2 dihedrals are sampled one standard deviation away from the mean. In addition, for wildtype simulations the input sidechain is added to the rotamer library at that position.

Single-residue sidechain repacking (row 1, Table I)

fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 1 -ddg::local_opt_only true -ddg::min_cst false -ddg::mean false -ddg::min true -ddg::sc_min_only false -ddg::opt_radius 0.1

The mutant residue is repacked within a fixed context according to the protocol for sidechain repacking. The side-chain packing procedure is first performed on the wild-type structure. The mutant residue is introduced and the same procedure is repeated. The predicted ΔΔG is the energy of mutant structure minus the energy of the wild-type structure.

Repack all sidechains within 8 Å of mutation (row 3, Table I)

./fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 50 -ddg::local_opt_only true -ddg::min_cst false -ddg::mean true -ddg::min false -ddg::sc_min_only false -ddg::ramp_repulsive false -ddg::opt_radius 8.0

All sidechain Cβs (or Cα in the case of Glycine) which fall within an 8 Å radius of the mutant sidechain Cβ were selectively repacked; all others are held fixed. The predicted ΔΔG is the mean of the mutant energies subtracted by the mean of the wildtype energies.

Repack all residues within 8 Å of mutation followed by sidechain minimization (row 5, Table I)

./fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 50 -ddg::local_opt_only true -ddg::min_cst true -ddg::mean false -ddg::min true -ddg::sc_min_only true -ddg::ramp_repulsive false -ddg::minimization_scorefunction standard -ddg::minimization_patch score12 -ddg::opt_radius 8.0

All sidechain Cβs (or Cα in the case of Glycine) which fall within an 8 Å radius of the mutant sidechain Cβ were selectively repacked; all others are held fixed. After sidechain repacking, sidechain degrees of freedom are minimized. The predicted ΔΔG is the difference between the mutant and wildtype mean of the lowest energy three structures.

Repack all sidechains (Row 6, Table I)

./fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 50 -ddg::local_opt_only false -ddg::min_cst false -ddg::mean true -ddg::min false -ddg::sc_min_only false

All sidechain residues are repacked according to the sidechain sampling protocol detailed above, keeping the backbone fixed. 50 models are produced for both the wild-type and mutant sequence contexts. The predicted ΔΔG is the mean of the mutant energies subtracted by the mean of the wildtype energies.

Repack all sidechains followed by sidechain minimization (Row 8, Table I)

./fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 50 -ddg::local_opt_only false -ddg::min_cst true -ddg::mean false -ddg::min true -ddg::sc_min_only true -ddg::ramp_repulsive false -ddg::minimization_scorefunction standard -ddg::minimization_patch score12

All sidechains are repacked. After sidechain repacking, sidechain degrees of freedom are minimized. The predicted ΔΔG is the difference between the mutant and wildtype mean of the lowest energy three structures.

Single-residue repacking followed by backbone and sidechain minimization (row 10, Table I)

fix_bb_monomer_ddg.linuxgccrelease -ddg::minimization_scorefunction standard -ddg::minimization_patch score12 -ddg::weight_file standard_plus_score12.wts -ddg::iterations 1 -ddg::local_opt_only true -ddg::min_cst true -ddg::mean false -ddg::min true -ddg::ramp_repulsive true -ddg::sc_min_only false -ddg::opt_radius 0.1

Models are produced according to the single-residue, fixed backbone protocol (row 1 of Table I). After sidechain repacking, all backbone and sidechain degrees of freedom are minimized. The predicted ΔΔG is the difference in energy between the mutant and wild-type optimized models.

Repack all sidechains within 8 Å followed by backbone and sidechain minimization (row 13, Table I)

./fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 50 -ddg::local_opt_only true -ddg::min_cst true -ddg::mean false -ddg::min true -ddg::sc_min_only false -ddg::ramp_repulsive true -ddg::minimization_scorefunction standard -ddg::minimization_patch score12 -ddg::opt_radius 8.0

All sidechain Cβs (or Cα in the case of Glycine) which fall within an 8 Å radius of the mutant sidechain Cβ were selectively repacked; all others are held fixed. After sidechain repacking, backbone and sidechain degrees of freedom are minimized, and the predicted ΔΔG is defined as the difference between the mutant and wildtype mean of the lowest 3 energy models out of 50.

Repack all sidechains followed by backbone and sidechain minimization (Row 16, Table I)

fix_bb_monomer_ddg.linuxgccrelease -ddg::weight_file soft_rep_design -ddg::iterations 50 -ddg::local_opt_only false -ddg::min_cst true -ddg::mean false -ddg::min true -ddg::sc_min_only false -ddg::ramp_repulsive true -ddg::minimization_scorefunction standard -ddg::minimization_patch score12

All sidechain residues are repacked, analogous to the protocol described in row 6 of Table I, producing 50 models according to the protocol. After full sidechain repacking, all backbone and sidechain degrees of freedom for each model are subjected to minimization. The predicted ΔΔG is the difference between the mutant and wildtype mean of the lowest energy three structures.

Repack all sidechains followed by backbone and sidechain minimization using position specific constraints (Row 18, Table I)

Command is identical to the limited backbone minimization protocol described in row 16 of Table I. Constraint file is generated in a different manner:

make_cst_file.linuxiccrelease –ddg::distance_from_mutsite 10.0 –ddg::strict_cst 0.5 -ddg::loose_cst 2

The procedure for computing the predicted ΔΔG is identical to the all-residues, backbone minimization protocol *(row 16, Table I*), except the constraint definition is altered. Instead of applying uniform constraints with sigma=0.5 over the entire structure, all residue constraints whose sidechain Cβs (or Cα in the case of Glycine) fall within a 10 Å radius of the mutant sidechain Cβ have a sigma= 2, allowing more freedom of movement. The predicted ΔΔG is the difference between mutant and wildtype mean of the lowest 3 energy models out of 50.

All-sidechains repacked followed by free-minimization (row 19, Table I)

Command is identical to (row 16, Table I), but constraint file is empty

All sidechain residues are repacked, analogous to the protocol described in row 6 of Table I, producing 50 models according to the protocol. After full sidechain repacking, all backbone and sidechain degrees of freedom for each model are subjected to free minimization. The predicted ΔΔG is the difference between mutant and wildtype mean of the lowest 3 energy models out of 50 models produced.

Monte Carlo ensemble method

ensemble_generator_score12_sidechain_ver2.linuxgccrelease -sc_min_only false -ddg::ramp_repulsive true -ddg::constraint_weight 1.0 -nstruct 200 -ddg::min_with_cst true -ddg::temperature 10.0 -ddg::use_bound_cst true

	mutant PDB ID	wild-type residue	position	mutant residue	wild-type PDB ID	mutant PDB ID	wild-type residue	position	mutant residue	wild-type PDB ID	mutant PDB ID	wild-type residue	position	mutant residue
107l	1l63	G	44	S	1rex	1cj8	T	40	A	1stn	2ey1	T	17	V
160l	1l63	A	120	M	1rex	1cj9	T	40	V	1stn	2ey5	T	36	S
161l	1l63	A	116	N	1rex	1ckc	T	43	A	1stn	2ey6	T	36	V
162l	1l63	A	122	Q	1rex	1ckd	T	43	V	1stn	2eyf	T	39	V
163l	1l63	A	123	Q	1rex	1ckf	T	52	A	1stn	2eyh	T	57	S
164l	1l63	A	119	R	1rex	1ckh	T	70	V	1stn	2eyj	T	57	V
165l	1l63	A	117	S	1rex	1oua	I	56	T	1stn	2eyl	T	77	S
								10						
171l	1l63	A	45	E	1rex	1oub	V	0	A	1stn	2eyo	T	115	S
								11						
1a2p	1ban	S	89	A	1rex	1ouc	V	0	A	1stn	2eyp	T	115	V
								12						
1a2p	1bao	Y	76	F	1rex	1oud	V	1	A	1stn	2f0d	I	87	V
								12						
1a2p	1bns	T	24	A	1rex	1oue	V	5	A	1stn	2f0h	V	61	L
1a2p	1brh	L	12	A	1rex	1oug	V	2	A	1stn	2f0j	I	67	V
1a2p	1bri	I	74	A	1rex	1ouh	V	74	A	200l	1l63	A	121	L
1a2p	1brj	I	86	A	1rex	1oui	V	93	A	227l	1l63	A	104	F
1a2p	1brk	I	94	A	1rex	1ouj	V	99	A	235l	1l63	A	111	V
1a2p	1bsa	I	49	V	1rex	1tcy	Y	63	F	236l	1l63	A	87	V
								12						
1a2p	1bsb	I	74	V	1rex	1wqm	Y	4	F	237l	1l63	A	149	V
1a2p	1bsc	I	86	V	1rex	1wqn	Y	20	F	238l	1l63	A	103	V
1a2p	1bsd	I	94	V	1rex	1wqo	Y	38	F	239l	1l63	A	17	I

1a2p	1bse	L	87	V	1rex	1wqp	Y	45	F	240l	1l63	A	27	I
1bpi	1bti	F	22	A	1rex	1wqq	Y	54	F	241l	1l63	A	29	I
								10						
1dyb	1lyd	G	131	V	1rex	1yam	I	6	V	242l	1l63	A	50	I
1fkj	2dg4	W	59	F	1rex	1yan	I	23	V	243l	1l63	A	58	I
1fkj	2dg9	W	59	L	1rex	1yao	I	56	V	244l	1l63	A	100	I
1hz6	1k50	V	52	A	1rex	1yap	I	59	V	246l	1l63	A	67	F
1l00	1lyd	A	105	Q	1rex	1yaq	I	89	V	247l	1l63	A	84	L
								10						
1l02	1lyd	A	157	T	1rex	2hea	I	6	A	253l	1l63	A	20	D
1l08	1lyd	G	157	T	1rex	2heb	I	23	A	2bqa	2bqb	I	106	V
1l21	1lyd	G	55	N	1rex	2hec	I	56	A	2bqa	2bqc	I	23	V
1l22	1lyd	G	124	K	1rex	2hed	I	59	A	2bqa	2bqd	I	56	V
1rex	2hee	I	59	G	2bqa	2bqf	I	89	V	2bqa	2bqh	V	110	A
1l33	1lyd	A	131	V	1rex	2hef	I	89	A	2bqa	2bqi	V	121	A
1l65	1l63	A	47	D	1rex	2meb	I	56	L	2bqa	2bqj	V	125	A
1l66	1l63	A	43	K	1rex	2mee	I	59	L	2bqa	2bql	V	2	A
											2bq			
1l67	1l63	A	46	L	1rex	2mef	I	59	M	2bqa	m	V	74	A
1l68	1l63	A	44	S	1rex	2meg	I	59	S	2lzm	1dyb	V	131	G
1l69	1lyd	A	133	L	1rex	2meh	I	59	T	2lzm	1l00	Q	105	A
1l85	1l63	A	153	F	1shg	1bk2	D	43	G	2lzm	1l02	T	157	A
1l90	1l63	A	99	L	1shg	1qkw	N	42	G	2lzm	1l08	T	157	G
1l99	1lyd	G	105	Q	1shg	1qkx	N	42	A	2lzm	1l14	T	157	S
1lyh	1l63	G	59	T	1stn	1ey4	S	54	A	2lzm	1l15	T	157	V
1lyj	1l63	A	59	T	1stn	1ey5	T	28	V	2lzm	1l52	T	152	S

wild-type PDB ID	mutant PDB ID	wild-type residue	position	mutant residue
2lzm	1l57	N	116	D
2lzm	1l69	L	133	A
2lzm	1l98	Q	105	E
2lzm	1l99	Q	105	G
4lyz	1heo	I	55	V
5azu	2tsa	M	121	A
1qtb	1l63	V	42	A
1rex	1b5u	S	24	A
1rex	1b5v	S	51	A
1rex	1b5w	S	61	A
1rex	1b5x	S	80	A
1rex	1b5y	S	36	A
1rex	1cj6	T	11	A
1rex	1cj7	T	11	V
1stn	1ey7	S	123	A
1stn	1kaa	K	111	A
1stn	1kab	K	111	G
1stn	1snm	E	38	D
1stn	1syc	P	112	G
1stn	1sye	P	112	T
1stn	1syg	P	112	A
2lzm	1l17	I	3	V
2lzm	1l20	N	144	D
2lzm	1l21	N	55	G

2lzm	1l22	K	124	G
2lzm	1l28	P	86	G
2lzm	1l33	V	131	A
2lzm	1l38	Q	123	E

Appendix table I. Table of wildtype and mutant crystal structure pairs used for studying prediction of mutant sidechains

Markov State Model Assessment Appendix

Methods

I. Microstate assignment methods

1. Secondary structure based microstate assignment

In order to assign microstates based on secondary structure, strand-pairing characteristics, called features, were extracted from beta-sheets assigned using DSSP(*140*). These features are combined to create feature-vectors used to describe any configuration, or a structure from a snap-shot of the simulation. The unique set of feature-vectors are referred to as microstate assignments.

Strand-pairing characteristics extracted from beta-sheet DSSP assignments include the paired residue positions of the strand, the direction of the strand, register, and pleating. Register is defined as the difference between the residue-positions for the paired residues making the hydrogen bond (j – i where j > i) for a parallel strand-pair and the sum of the residue positions for the residue-pair making the hydrogen bond (j + i) for an anti-parallel strand-pair. Bulges in the strand were defined using both registers comprising the bulge. The *direction* of a strand pairing is *parallel* or *antiparallel* and the *pleating* is *inward* or *outward* and denotes whether the Cα atom of the two paired residue point toward each other or away from each other, respectively.

Other characteristics, such as sequence separation (the number of residues separating the paired residue positions), and extent of hydrogen bonding (i.e. 3 or more consecutive hydrogen bonds in a strand) were further used to filter the feature-set as detailed below. Different microstate assignments result from exclusion of a subset of the characteristics listed above to define features.

a. Featurization 1 (4,857 states)

A strand pairing feature consists of the register and direction. All strand-pairing features are combined into a feature-vector such that each feature occurs only once (i.e, multiple strand pairings with the same register and direction might co-exist and thus would give rise to the same feature which is only included once in the feature-vector). Bulges are represented by the two registers that comprise the bulge but do not overlap with existing features of the same register. Two feature-vectors are equivalent if they contain the same number of features and if the individual features are identical. Each unique feature-vector forms a discrete microstate assignment. This procedure yielded a total of 4,857 microstates for Featurization 1.

b. Featurization 2 (548 states)

In Featurization 1 we found many sparsely populated states. To reduce the size of state-space we removed rarely occurring features. Additionally we found that the sequence-position of a formed strand should matter, as a strand with the same register and directionality formed at the two opposing ends of the peptide is likely to be kinetically disconnected. To include this new information, we defined strand-pairing features as any strand-pair which falls within a defined range of sequence-positions and have the appropriate register as defined in Appendix Table II. Any strand-pairs that fall into the admissible range of sequence-positions for their register receive the same feature-definition A pleating characteristic is further assigned to each feature according to the extrapolated pleating at the first sequence position of the admissible range. Extrapolation of pleatings assumes strict alternation between *inward* and *outward* for each residue position. In total this featurization yielded 548 unique microstates.

c. Featurization 3 (175 states)

Removing the pleating information from the feature definition in Featurization two resulted in 175 microstates.

d. Featurization 4 (402 states)

We observed that many spurious pairings in Featurization 2 came from pairings that were close in sequence. Thus, starting from Featurization 2 we excluded strand-pairings which did not satisfy a minimum sequence separation of 14 residues. In total, this state-assignment definition resulted in 402 unique states.

e. Featurization 5 (371 states)

In this featurization we included a hydrogen bond threshold to throw out spurious strand-pairings which were less likely to persist. Starting from Featurization 2 we kept only strand pairings belonging to a strand that had 3 or more hydrogen bonds, This yielded a total of 371 states.

f. Secondary structure based microstate reassignment

The cross-validation procedure involves geometric reassignment of microstates observed in the test-set to the next closest microstate occurring in the training-set (see main text). Because the criteria used to define secondary structure microstates are categorical, the geometric reassignment is non-trivial. In order to reassign feature-vectors occurring in the test-set but not in the training-set, a distance metric was used to look for the most commonly occurring sets of features contained in another feature-vector. Under-specification is preferred over over-specification. For example, if the test-set feature-vector consists of the two features (F1, F2) and the training-set ensemble contains feature-vectors (F1, F2, F3) and (F1), the test-set feature-vector will be assigned to (F1). Even though there is a training-set state which has both features F1 and F2, the additional presence of F3 would exclude a conformation with feature vector (F1,F2). Test-set feature-vectors with no matching training-set feature-vector will be assigned to the feature-vector with no secondary structure.

2. Kmean refinement of contact-maps

We applied the Kmean algorithm as follows. Clusters obtained with the much faster Kcenter algorithm were used as starting point (because good initial estimates for starting clusters lead to significantly faster convergence of Kmean refinement). In the assignment step each contact map is assigned to the nearest cluster center using an Euclidean distance norm on the individual matrix elements. In the subsequent update step for each cluster a new center is computed as the matrix whose elements consists of the arithmetic mean of all contact maps assigned to the cluster. The assignment and update steps were repeated until assignments remain unchanged with more iterations. Our tests showed that most improvements in average cluster-radii as well as log-likelihood occurred within the first ten iterations, thus refinements were limited to 20 iterations of Kmean for each initial cluster-assignment. As the procedure’s bottleneck occurs during assignment (due to the large numbers of assignments which need to be performed for each iteration) we implemented a parallelized version of the Kmean-refinement program. To keep memory utilization minimal on the worker processes, assignments are distributed to and collected from the worker processes by the master process. The resulting protocol takes on the order of one or two days to complete the 20 rounds of refinement using 120 processes for an initial assignment comprising around 10,000 clusters, versus the original serial protocol, which took 1 day to complete one round of refinement. As the assignment procedure is dependent on the number of clusters specified (number of distance computations is equal to the number of configurations multiplied by the number of clusters for each round of assignment), this refinement procedure is significantly faster for microstate set sizes of one thousand clusters or less. Future improvements of this clustering implementation will include sparse matrix implementation, addressing the substantial memory requirements.

II. Analysis methods

1. Macrostate assignment

A suite of matlab scripts were used to perform all analyses described. First the data is read in and a count-matrix is computed using a lag-time of 100 ns. The resulting count-matrices were then symmetrized, as it is common practice(*93, 94, 121*). Eigenvalues and eigenvectors were obtained from the normalized transition matrix, and perron clustering(*90, 99*) was performed with one additional modification. We found that improved perron clustering results were obtained on toy-model systems if we first subtracted off the mean of the normalized-eigenvector (data not shown). We found that while the boundaries of the macrostates changed slightly (based on 2-dimensional toy-models), the macrostate definitions were generally the same. Once macrostates were assigned, the macrostate count-matrices and the corresponding macrostate-trajectories were partitioned into training and testing sets, and the cross-validation procedure was performed according to that detailed in the methods section (see main text).

2. Flux-diagram representation

The flux diagrams shown in the Appendix and figure 28B were constructed as described previously(*97*). Because it is not clear what states should be defined as the unfolded state, only macrostates with cluster-center contact-maps that contained no secondary structure were classified as unfolded. Alternative sets of states were used to test the effect of unfolded state definition on the resulting flux diagram, but the diagrams remain essentially identical. First the pfold for each macrostate was computed according to transition path theory(*135*). Then, the flux between states i and j is computed as the product of the transition probability from i to j, the equilibrium probability of state i, and the difference in pfold of i and j. The flux is displayed only if the pfold of j is greater than that of i.(*135*) The resulting diagram shows each state whose size is dependent on its equilibrium probability (computed as the number of configurations in state i divided by the total number of configurations), with caps on the minimum and maximum sizes for visual clarity. The arrows are drawn relative to the flux computed, with thicker lines corresponding to higher flux. States that do not contribute to the folding-pathway (i.e. alternative states which are kinetic-traps) were omitted for visual clarity. Hierarchical clustering on the folding-transition macrostate contact-maps for the flux diagram in figure 28B was utilized in order to simplify visualization, resulting in 17 visually distinct macrostate contact-maps. Transition probabilities as well as equilibrium probabilities are computed according to the hierarchically clustered macrostate assignments. However, the conclusions drawn from the flux diagrams remain identical whether or not hierarchical clustering is used.

3. Folding-transition analysis methods

a. Folding-transition definition

As mentioned in the main text, the analysis on the folding-trajectories involved definition of the folding-transitions. The folding transitions were defined manually and correspond to each of the 14 folding and unfolding events observed in the 200 microsecond simulation. Any unfolding trajectory was reversed to resemble a folding transition. See Kellogg et al.(*138*) for the exact definitions.

b. Folding-trajectory markov state model construction

The process of markov state model construction first requires geometric definition of microstates (see main text). Because geometric clustering requires significant computational time, the Kmean refined microstate assignments obtained from clustering the full set of trajectory snap-shots was used. 100 microstates were initially defined using the full-set of configurations, but 95 microstates remained after restricting analysis to the folding-transition regions. Macrostate models were constructed as described in the section: "*MSM construction*" using a 10 ns lag-time. 20-40 macrostates was found to be optimal for describing folding, as determined by the maximum log-likelihood for varying macrostate model definitions. All subsequent folding-transition analyses used the 20 macrostate model.

c. Folding-transition markov state model analysis

We computed the contact-map representing the center of each of the macrostates, normalized such that the maximal intensity for each contact-map was 1, and found that the contact-map representing the cluster center was redundant with other macrostates in the ensemble. To summarize the information in the macrostate cluster-centers visually, we performed hierarchical clustering to combine redundant cluster-center contact-maps, resulting in 17 visually distinct contact-maps.

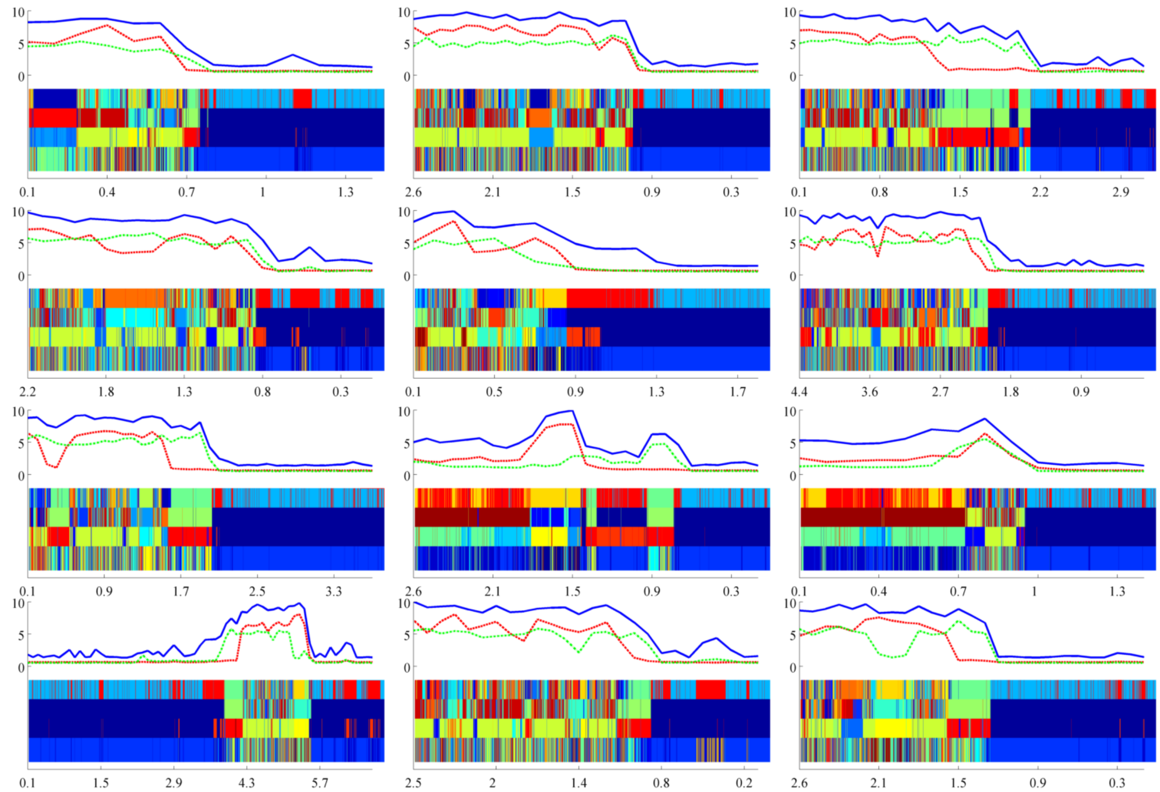
d. Folding-pathway reconstruction

Because the transitions between macrostates are visually difficult to parse, we utilized a method to reduce the noise in the macrostate trajectory transitions. First, we split the trajectory into intervals picked qualitatively. Intervals were chosen on a case-by-case basis such that transitions in the reduced trajectory accurately represented the original trajectories and did not exceed the lag-time used (10 ns or 50 configurations). Only the most frequently occurring state in each interval of the trajectory was retained. A second step was taken to eliminate short, nonproductive excursions (i.e. transitions which begin and end in the same state). If the time spent in an excursion from a metastable state is less than one percent of the entire time of the trajectory and the excursion starts and begins in the same state, then this transition was eliminated. This manner of summarizing the trajectory information, while visually useful, can lead to discrepancies between the results obtained using this method and the flux-diagram (data not shown). The resulting folding-pathways were qualitatively compared to the raw simulation data to confirm the validity of this method. This technique to reduce trajectory noise was applied to produce the folding-pathways displayed in figure 28A.

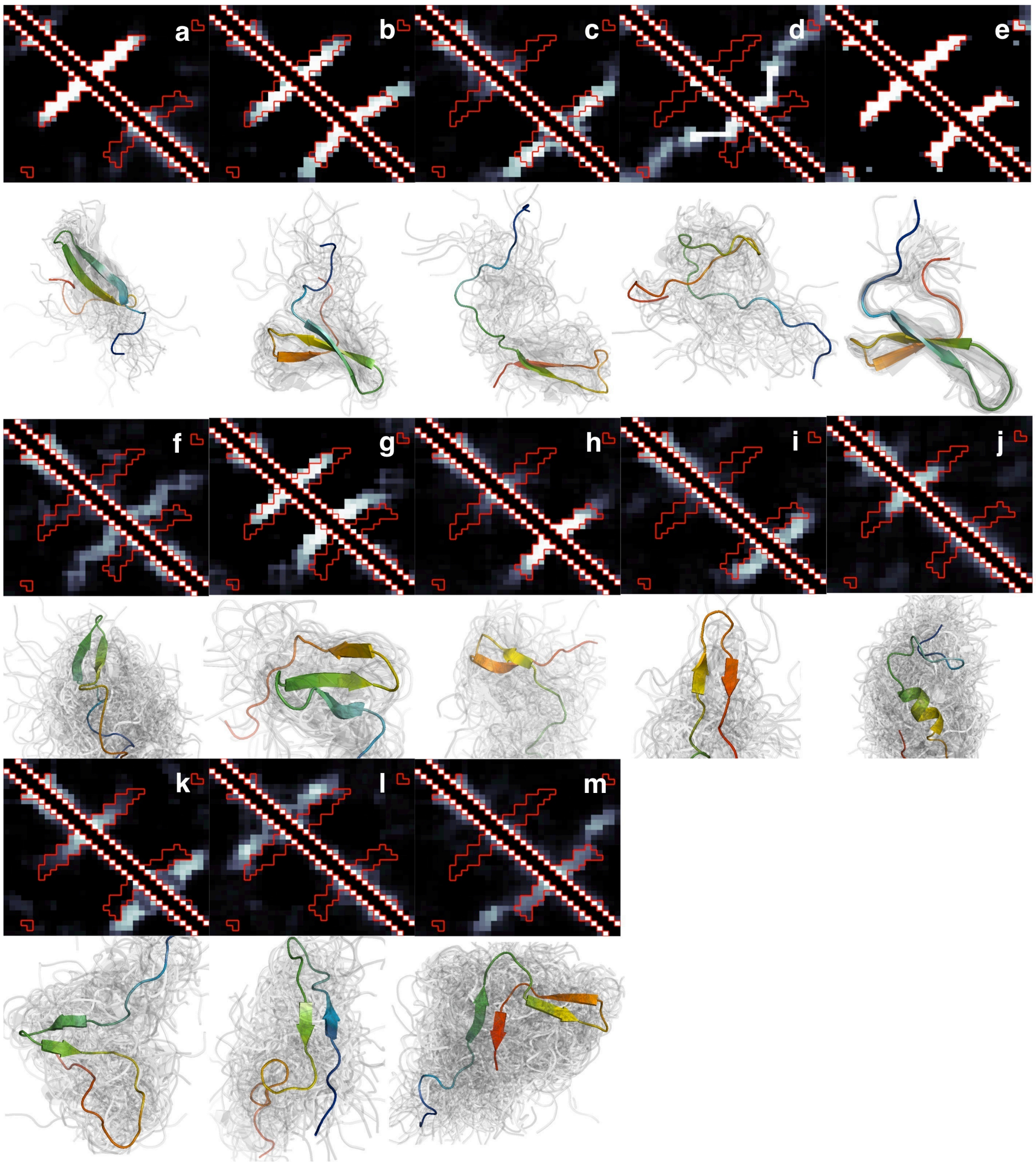
4. Hierarchical clustering for visual clarity

We performed hierarchical clustering on the representative macrostate contact-maps produced in the folding-transition analysis section to simplify results visually. Each contact-map starts out in its own cluster. Clusters of contact-maps are progressively combined based on the mean Euclidean distance between all possible contact-map pairs between two clusters. The final cluster-assignment was chosen by visual inspection, resulting in 17 clusters from the original 20. The resulting cluster-assignments are used to define macrostate count-matrices as well as representative macrostate contact-maps for all resulting analyses.

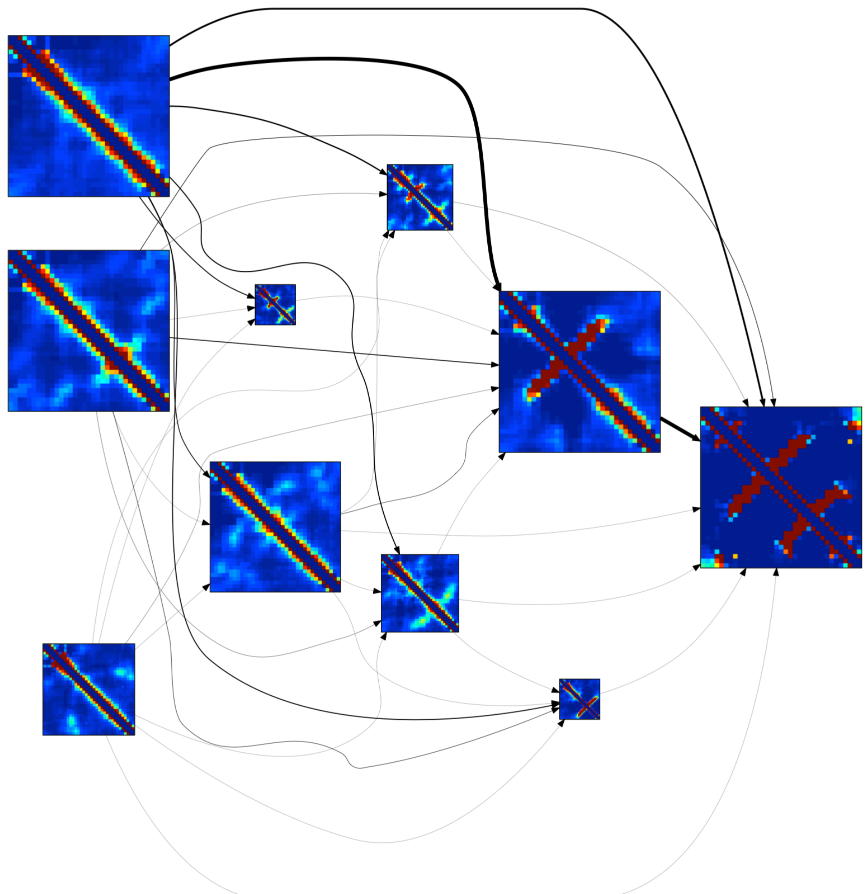
All software and examples on how to utilize them are available upon request.



Appendix Figure 1. Full set of macrostate trajectory folding-transitions. Each panel represents a folding transition (unfolding transitions are reversed to look like folding transitions). The top-portion of each panel corresponds to the rmsd to native of different parts of the structure: blue is rmsd-to-full length native, red is rmsd to hairpin 1, and green is rmsd to hairpin two (same as in figure 3 of main text). The lower portion of each panel represents the microstate/macrostate trajectory of each constructed model. The models are chosen according to the maximum log-likelihood and corresponds to the same order and models shown in figure 3 of the main text.


a
b
c
d
e
f
g
h
i
j
k
l
m

Appendix Figure 2. Structural representatives for each macrostate. The macrostate model is defined by using 100 initial microstate Kmean contact clusters followed by perron-clustering into 20 final states. The data used is based on only the folding-transitions and a lag-time of 10 ns to gain a more detailed view of the folding events. The upper portion of each panel represents the macrostate center contact map. The red outline indicates the contacts formed in the native state. The lower portion of each panel shows a representative structure belonging to the macrostate and the gray are the superimposed members of the macrostate. States which show no regular structure or duplicate features (i.e. an alternative native state missing tail contacts, for example) were omitted.



Appendix Figure 3. Network representation (flux) of macrostate model depicting ww-domain folding. Model shown is the 100 microstate Kmean contact refined model with 20 kinetically-defined macrostates (constructed from a 100 ns lag-time). Some unfolded states which do not contribute to the folding-flux are omitted for visual clarity.

direction	register	start residue	end residue
anti-parallel	26	7	9
	12	2	4
	28	8	10
	29	8	12
	30	7	13
	31	9	14
	32	10	14
	33	11	15
	34	2	4
	36	2	4
	43	17	20
	44	18	20
	45	18	21
	44	11	13
	45	12	14
	48	18	22
	49	18	23
	50	17	22
	51	20	22
	53	22	24
	55	22	26
	56	25	26

Appendix Table II. The feature information for selected regions of strand-pairs based on counts of data, as described in featurization 2-5. These are manually defined and consist of a region as well as the register that accompanies the region of interest. A strand-pair is only considered if it falls within the appropriate region (including start and end residue) as well as having the corresponding register. All selected strand-pairing features are anti-parallel, parallel strand-pairs were not populated to a significant degree.

References

1. Qian, B., Raman, S., Das, R., Bradley, P., McCoy, A. J., Read, R. J., and Baker, D. (2007) High-resolution structure prediction and the crystallographic phase problem, *Nature 450*, 259-264.
2. Kortemme, T., and Baker, D. (2002) A simple physical model for binding energy hot spots in protein-protein complexes, *Proc Natl Acad Sci U S A 99*, 14116-14121.
3. Guerois, R., Nielsen, J. E., and Serrano, L. (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations, *J Mol Biol 320*, 369-387.
4. Bordner, A. J., and Abagyan, R. A. (2004) Large-scale prediction of protein geometry and stability changes for arbitrary single point mutations, *Proteins 57*, 400-413.
5. Yin, S., Ding, F., and Dokholyan, N. V. (2007) Eris: an automated estimator of protein stability, *Nat Methods 4*, 466-467.
6. Benedix, A., Becker, C. M., de Groot, B. L., Caflisch, A., and Bockmann, R. A. (2009) Predicting free energy changes using structural ensembles, *Nat Methods 6*, 3-4.
7. Potapov, V., Cohen, M., and Schreiber, G. (2009) Assessing computational methods for predicting protein stability upon mutation: good on average but not in the details, *Protein Eng Des Sel 22*, 553-560.
8. Kumar, M. D., Bava, K. A., Gromiha, M. M., Prabakaran, P., Kitajima, K., Uedaira, H., and Sarai, A. (2006) ProTherm and ProNIT: thermodynamic databases for proteins and protein-nucleic acid interactions, *Nucleic Acids Res 34*, D204-206.
9. Kuhlman, B., and Baker, D. (2000) Native protein sequences are close to optimal for their structures, *Proc Natl Acad Sci U S A 97*, 10383-10388.
10. Das, R., and Baker, D. (2008) Macromolecular modeling with rosetta, *Annu Rev Biochem 77*, 363-382.
11. Rohl, C. A., Strauss, C. E., Misura, K. M., and Baker, D. (2004) Protein structure prediction using Rosetta, *Methods Enzymol 383*, 66-93.
12. Dantas, G., Corrent, C., Reichow, S. L., Havranek, J. J., Eletr, Z. M., Isern, N. G., Kuhlman, B., Varani, G., Merritt, E. A., and Baker, D. (2007) High-resolution structural and thermodynamic analysis of extreme stabilization of human procarboxypeptidase by computational protein design, *J Mol Biol 366*, 1209-1221.
13. Kuhlman, B., Dantas, G., Ireton, G. C., Varani, G., Stoddard, B. L., and Baker, D. (2003) Design of a novel globular protein fold with atomic-level accuracy, *Science 302*, 1364-1368.
14. Kiranyaz, S., Ince, T., Yildirim, A., and Gabbouj, M. Fractional particle swarm optimization in multidimensional search space, *IEEE Trans Syst Man Cybern B Cybern 40*, 298-319.
15. Davis, I. W., Arendall, W. B., 3rd, Richardson, D. C., and Richardson, J. S. (2006) The backrub motion: how protein backbone shrugs when a sidechain dances, *Structure 14*, 265-274.

16. Smith, C. A., and Kortemme, T. (2008) Backrub-like backbone simulation recapitulates natural protein conformational variability and improves mutant side-chain prediction, *J Mol Biol 380*, 742-756.
17. Kellogg, E. H., Leaver-Fay, A., and Baker, D. Role of conformational sampling in computing mutation-induced changes in protein structure and stability, *Proteins 79*, 830-838.
18. Jiang, L., Kuhlman, B., Kortemme, T., and Baker, D. (2005) A "solvated rotamer" approach to modeling water-mediated hydrogen bonds at protein-protein interfaces, *Proteins 58*, 893-904.
19. Gilis, D., and Rooman, M. (1997) Predicting protein stability changes upon mutation using database-derived potentials: solvent accessibility determines the importance of local versus non-local interactions along the sequence, *J Mol Biol 272*, 276-290.
20. Green, S. M., Meeker, A. K., and Shortle, D. (1992) Contributions of the polar, uncharged amino acids to the stability of staphylococcal nuclease: evidence for mutational effects on the free energy of the denatured state, *Biochemistry 31*, 5717-5728.
21. Shortle, D., Stites, W. E., and Meeker, A. K. (1990) Contributions of the large hydrophobic amino acids to the stability of staphylococcal nuclease, *Biochemistry 29*, 8033-8041.
22. Byrne, M. P., Manuel, R. L., Lowe, L. G., and Stites, W. E. (1995) Energetic contribution of side chain hydrogen bonding to the stability of staphylococcal nuclease, *Biochemistry 34*, 13949-13960.
23. Meeker, A. K., Garcia-Moreno, B., and Shortle, D. (1996) Contributions of the ionizable amino acids to the stability of staphylococcal nuclease, *Biochemistry 35*, 6443-6449.
24. Consonni, R., Santomo, L., Fusi, P., Tortora, P., and Zetta, L. (1999) A single-point mutation in the extreme heat- and pressure-resistant sso7d protein from sulfolobus solfataricus leads to a major rearrangement of the hydrophobic core, *Biochemistry 38*, 12709-12717.
25. Fulton, K. F., Jackson, S. E., and Buckle, A. M. (2003) Energetic and structural analysis of the role of tryptophan 59 in FKBP12, *Biochemistry 42*, 2364-2372.
26. Pappu, R. V., Marshall, G. R., and Ponder, J. W. (1999) A potential smoothing algorithm accurately predicts transmembrane helix packing, *Nat Struct Biol 6*, 50-55.
27. Lazaridis, T., and Karplus, M. (1999) Effective energy function for proteins in solution, *Proteins 35*, 133-152.
28. Shortle, D., and Meeker, A. K. (1989) Residual structure in large fragments of staphylococcal nuclease: effects of amino acid substitutions, *Biochemistry 28*, 936-944.
29. Korkegian, A., Black, M. E., Baker, D., and Stoddard, B. L. (2005) Computational thermostabilization of an enzyme, *Science 308*, 857-860.
30. Saunders, C. T., and Baker, D. (2002) Evaluation of structural and evolutionary contributions to deleterious mutation prediction, *J Mol Biol 322*, 891-901.
31. Cheng, G., Qian, B., Samudrala, R., and Baker, D. (2005) Improvement in protein functional site prediction by distinguishing structural and functional constraints on protein family evolution using computational design, *Nucleic Acids Res 33*, 5861-5867.
32. Baase, W. A., Liu, L., Tronrud, D. E., and Matthews, B. W. Lessons from the lysozyme of phage T4, *Protein Sci 19*, 631-641.

33. Nguyen, H., Jager, M., Moretto, A., Gruebele, M., and Kelly, J. W. (2003) Tuning the free-energy landscape of a WW domain by temperature, mutation, and truncation, *Proc Natl Acad Sci U S A 100*, 3948-3953.
34. Maisuradze, G. G., Zhou, R., Liwo, A., Xiao, Y., and Scheraga, H. A. Effects of mutation, truncation, and temperature on the folding kinetics of a WW domain, *J Mol Biol 420*, 350-365.
35. Jiang, X., Kowalski, J., and Kelly, J. W. (2001) Increasing protein stability using a rational approach combining sequence homology and structural alignment: Stabilizing the WW domain, *Protein Sci 10*, 1454-1465.
36. Jager, M., Nguyen, H., Crane, J. C., Kelly, J. W., and Gruebele, M. (2001) The folding mechanism of a beta-sheet: the WW domain, *J Mol Biol 311*, 373-393.
37. Jager, M., Zhang, Y., Bieschke, J., Nguyen, H., Dendle, M., Bowman, M. E., Noel, J. P., Gruebele, M., and Kelly, J. W. (2006) Structure-function-folding relationship in a WW domain, *Proc Natl Acad Sci U S A 103*, 10648-10653.
38. Yanagida, H., Matsuura, T., and Yomo, T. (2008) Compensatory evolution of a WW domain variant lacking the strictly conserved Trp residue, *J Mol Evol 66*, 61-71.
39. Jager, M., Dendle, M., Fuller, A. A., and Kelly, J. W. (2007) A cross-strand Trp Trp pair stabilizes the hPin1 WW domain at the expense of function, *Protein Sci 16*, 2306-2313.
40. Jager, M., Dendle, M., and Kelly, J. W. (2009) Sequence determinants of thermodynamic stability in a WW domain--an all-beta-sheet protein, *Protein Sci 18*, 1806-1813.
41. Pires, J. R., Taha-Nejad, F., Toepert, F., Ast, T., Hoffmuller, U., Schneider-Mergener, J., Kuhne, R., Macias, M. J., and Oschkinat, H. (2001) Solution structures of the YAP65 WW domain and the variant L30 K in complex with the peptides GTPPPPYTVG, N-(n-octyl)-GPPPY and PLPPY and the application of peptide libraries reveal a minimal binding epitope, *J Mol Biol 314*, 1147-1156.
42. Espanel, X., Navin, N., Kato, Y., Tanokura, M., and Sudol, M. (2003) Probing WW Domains to Uncover and Refine Determinants of Specificity in Ligand Recognition, *Cytotechnology 43*, 105-111.
43. Fowler, D. M., Araya, C. L., Fleishman, S. J., Kellogg, E. H., Stephany, J. J., Baker, D., and Fields, S. High-resolution mapping of protein sequence-function relationships, *Nat Methods 7*, 741-746.
44. Eriksson, A. E., Baase, W. A., Zhang, X. J., Heinz, D. W., Blaber, M., Baldwin, E. P., and Matthews, B. W. (1992) Response of a protein structure to cavity-creating mutations and its relation to the hydrophobic effect, *Science 255*, 178-183.
45. Skrynnikov, N. R., Dahlquist, F. W., and Kay, L. E. (2002) Reconstructing NMR spectra of "invisible" excited protein states using HSQC and HMQC experiments, *J Am Chem Soc 124*, 12352-12360.
46. Feher, V. A., Baldwin, E. P., and Dahlquist, F. W. (1996) Access of ligands to cavities within the core of a protein is rapid, *Nat Struct Biol 3*, 516-521.
47. Bouvignies, G., Vallurupalli, P., Hansen, D. F., Correia, B. E., Lange, O., Bah, A., Vernon, R. M., Dahlquist, F. W., Baker, D., and Kay, L. E. Solution structure of a minor and transiently formed state of a T4 lysozyme mutant, *Nature 477*, 111-114.
48. Shen, Y., Lange, O., Delaglio, F., Rossi, P., Aramini, J. M., Liu, G., Eletsky, A., Wu, Y., Singarapu, K. K., Lemak, A., Ignatchenko, A., Arrowsmith, C. H., Szyperski, T., Montelione, G. T., Baker, D., and Bax, A. (2008) Consistent blind protein structure generation from NMR chemical shift data, *Proc Natl Acad Sci U S A 105*, 4685-4690.

49. Vallurupalli, P., Hansen, D. F., Lundstrom, P., and Kay, L. E. (2009) CPMG relaxation dispersion NMR experiments measuring glycine 1H alpha and 13C alpha chemical shifts in the 'invisible' excited states of proteins, *J Biomol NMR 45*, 45-55.
50. Brzovic, P. S., and Klevit, R. E. (2006) Ubiquitin transfer from the E2 perspective: why is UbcH5 so promiscuous?, *Cell Cycle 5*, 2867-2873.
51. Richter, B., Gsponer, J., Varnai, P., Salvatella, X., and Vendruscolo, M. (2007) The MUMO (minimal under-restraining minimal over-restraining) method for the determination of native state ensembles of proteins, *J Biomol NMR 37*, 117-135.
52. Lindorff-Larsen, K., Best, R. B., Depristo, M. A., Dobson, C. M., and Vendruscolo, M. (2005) Simultaneous determination of protein structure and dynamics, *Nature 433*, 128-132.
53. Lange, O. F., Lakomek, N. A., Fares, C., Schroder, G. F., Walter, K. F., Becker, S., Meiler, J., Grubmuller, H., Griesinger, C., and de Groot, B. L. (2008) Recognition dynamics up to microseconds revealed from an RDC-derived ubiquitin ensemble in solution, *Science 320*, 1471-1475.
54. Gardino, A. K., Villali, J., Kivenson, A., Lei, M., Liu, C. F., Steindel, P., Eisenmesser, E. Z., Labeikovsky, W., Wolf-Watz, M., Clarkson, M. W., and Kern, D. (2009) Transient non-native hydrogen bonds promote activation of a signaling protein, *Cell 139*, 1109-1118.
55. Fraser, J. S., Clarkson, M. W., Degnan, S. C., Erion, R., Kern, D., and Alber, T. (2009) Hidden alternative structures of proline isomerase essential for catalysis, *Nature 462*, 669-673.
56. Hartl, F. U., Bracher, A., and Hayer-Hartl, M. Molecular chaperones in protein folding and proteostasis, *Nature 475*, 324-332.
57. Van Dorn, L. O., Newlove, T., Chang, S., Ingram, W. M., and Cordes, M. H. (2006) Relationship between sequence determinants of stability for two natural homologous proteins with different folds, *Biochemistry 45*, 10542-10553.
58. Newlove, T., Konieczka, J. H., and Cordes, M. H. (2004) Secondary structure switching in Cro protein evolution, *Structure 12*, 569-581.
59. Alexander, P. A., He, Y., Chen, Y., Orban, J., and Bryan, P. N. (2007) The design and characterization of two proteins with 88% sequence identity but different structure and function, *Proc Natl Acad Sci U S A 104*, 11963-11968.
60. Alexander, P. A., He, Y., Chen, Y., Orban, J., and Bryan, P. N. (2009) A minimal sequence code for switching protein structure and function, *Proc Natl Acad Sci U S A 106*, 21149-21154.
61. Tyka, M. D., Keedy, D. A., Andre, I., Dimaio, F., Song, Y., Richardson, D. C., Richardson, J. S., and Baker, D. Alternate states of proteins revealed by detailed energy landscape mapping, *J Mol Biol 405*, 607-618.
62. D.A. Case, T. A. D., T.E. Cheatham, III, C.L. Simmerling, J. Wang, R.E. Duke, R. Luo, R.C. Walker, W. Zhang, K.M. Merz, B. Roberts, S. Hayik, A. Roitberg, G. Seabra, J. Swails, A.W. Goetz, I. Kolossváry, K.F. Wong, F. Paesani, J. Vanicek, R.M. Wolf, J. Liu, X. Wu, S.R. Brozell, T. Steinbrecher, H. Gohlke, Q. Cai, X. Ye, J. Wang, M.-J. Hsieh, G. Cui, D.R. Roe, D.H. Mathews, M.G. Seetin, R. Salomon-Ferrer, C. Sagui, V. Babin, T. Luchko, S. Gusarov, A. Kovalenko, and P.A. Kollman. (2012) AMBER 12.
63. Wang, J., Wolf, R. M., Caldwell, J. W., Kollman, P. A., and Case, D. A. (2004) Development and testing of a general amber force field, *J Comput Chem 25*, 1157-1174.

64. Yang, L., Tan, C. H., Hsieh, M. J., Wang, J., Duan, Y., Cieplak, P., Caldwell, J., Kollman, P. A., and Luo, R. (2006) New-generation amber united-atom force field, *J Phys Chem B 110*, 13166-13176.
65. Brooks, B. R., Brooks, C. L., 3rd, Mackerell, A. D., Jr., Nilsson, L., Petrella, R. J., Roux, B., Won, Y., Archontis, G., Bartels, C., Boresch, S., Caflisch, A., Caves, L., Cui, Q., Dinner, A. R., Feig, M., Fischer, S., Gao, J., Hodoscek, M., Im, W., Kuczera, K., Lazaridis, T., Ma, J., Ovchinnikov, V., Paci, E., Pastor, R. W., Post, C. B., Pu, J. Z., Schaefer, M., Tidor, B., Venable, R. M., Woodcock, H. L., Wu, X., Yang, W., York, D. M., and Karplus, M. (2009) CHARMM: the biomolecular simulation program, *J Comput Chem 30*, 1545-1614.
66. Giorgio Favrin, A. I., and Fredrik Sjunnesson. (2001) Monte Carlo Update for Chain Molecules: Biased Gaussian Steps in Torsional Space, *Journal of Chemical Physics*.
67. Ulmschneider, J. P., and Jorgensen, W. L. (2004) Polypeptide folding using Monte Carlo sampling, concerted rotation, and continuum solvation, *J Am Chem Soc 126*, 1849-1857.
68. Friedland, G. D., Linares, A. J., Smith, C. A., and Kortemme, T. (2008) A simple model of backbone flexibility improves modeling of side-chain conformational variability, *J Mol Biol 380*, 757-774.
69. Patriksson, A., and van der Spoel, D. (2008) A temperature predictor for parallel tempering simulations, *Phys Chem Chem Phys 10*, 2073-2077.
70. Earl, D. J., and Deem, M. W. (2005) Parallel tempering: theory, applications, and new perspectives, *Phys Chem Chem Phys 7*, 3910-3916.
71. Kone, A., and Kofke, D. A. (2005) Selection of temperature intervals for parallel-tempering simulations, *J Chem Phys 122*, 206101.
72. Deem, D. J. E. a. M. W. (2005) Parallel Tempering: Theory, applications, and new perspectives, *Physical Chemistry Chemical Physics 7*, 3910-3916.
73. Cramer, C. J. (2004) *Essentials of Computational Chemistry: Theory and Models*, 2 ed., Wiley.
74. Bolhuis, P. G., Chandler, D., Dellago, C., and Geissler, P. L. (2002) Transition path sampling: throwing ropes over rough mountain passes, in the dark, *Annu Rev Phys Chem 53*, 291-318.
75. Wu, L., McElheny, D., Huang, R., and Keiderling, T. A. (2009) Role of tryptophan-tryptophan interactions in Trpzip beta-hairpin formation, structure, and stability, *Biochemistry 48*, 10362-10371.
76. Eidenschink, L., Kier, B. L., Huggins, K. N., and Andersen, N. H. (2009) Very short peptides with stable folds: building on the interrelationship of Trp/Trp, Trp/cation, and Trp/backbone-amide interaction geometries, *Proteins 75*, 308-322.
77. Barua, B., Lin, J. C., Williams, V. D., Kummler, P., Neidigh, J. W., and Andersen, N. H. (2008) The Trp-cage: optimizing the stability of a globular miniprotein, *Protein Eng Des Sel 21*, 171-185.
78. Kubelka, J., Chiu, T. K., Davies, D. R., Eaton, W. A., and Hofrichter, J. (2006) Sub-microsecond protein folding, *J Mol Biol 359*, 546-553.
79. Hornak, V., Abel, R., Okur, A., Strockbine, B., Roitberg, A., and Simmerling, C. (2006) Comparison of multiple Amber force fields and development of improved protein backbone parameters, *Proteins 65*, 712-725.
80. Das, R. Four small puzzles that Rosetta doesn't solve, *PLoS One 6*, e20044.

81. Ding, F., Tsao, D., Nie, H., and Dokholyan, N. V. (2008) Ab initio folding of proteins with all-atom discrete molecular dynamics, *Structure 16*, 1010-1018.
82. Periole, X., Allen, L. R., Tamiola, K., Mark, A. E., and Paci, E. (2009) Probing the free energy landscape of the FBP28WW domain using multiple techniques, *J Comput Chem 30*, 1059-1068.
83. Maisuradze, G. G., Senet, P., Czaplewski, C., Liwo, A., and Scheraga, H. A. Investigation of protein folding by coarse-grained molecular dynamics with the UNRES force field, *J Phys Chem A 114*, 4471-4485.
84. Xu, J., Huang, L., and Shakhnovich, E. I. The ensemble folding kinetics of the FBP28 WW domain revealed by an all-atom Monte Carlo simulation in a knowledge-based potential, *Proteins 79*, 1704-1714.
85. Hills, R. D., Jr., and Brooks, C. L., 3rd. (2009) Insights from coarse-grained go models for protein folding and dynamics, *Int J Mol Sci 10*, 889-905.
86. Skolnick, J., and Kolinski, A. (1991) Dynamic Monte Carlo simulations of a new lattice model of globular protein folding, structure and dynamics, *J Mol Biol 221*, 499-531.
87. Ozkan, S. B., Bahar, I., and Dill, K. A. (2001) Transition states and the meaning of Phi-values in protein folding kinetics, *Nat Struct Biol 8*, 765-769.
88. Sun, S., Thomas, P. D., and Dill, K. A. (1995) A simple protein folding algorithm using a binary code and secondary structure constraints, *Protein Eng 8*, 769-778.
89. Dill, K. A., and MacCallum, J. L. The protein-folding problem, 50 years on, *Science 338*, 1042-1046.
90. Chodera, J. D., Singhal, N., Pande, V. S., Dill, K. A., and Swope, W. C. (2007) Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics, *J Chem Phys 126*, 155101.
91. Prinz, J. H., Wu, H., Sarich, M., Keller, B., Senne, M., Held, M., Chodera, J. D., Schutte, C., and Noe, F. Markov models of molecular kinetics: generation and validation, *J Chem Phys 134*, 174105.
92. Noe, F., Schutte, C., Vanden-Eijnden, E., Reich, L., and Weikl, T. R. (2009) Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations, *Proc Natl Acad Sci U S A 106*, 19011-19016.
93. Bowman, G. R., Beauchamp, K. A., Boxer, G., and Pande, V. S. (2009) Progress and challenges in the automated construction of Markov state models for full protein systems, *J Chem Phys 131*, 124101.
94. Bowman, G. R., Voelz, V. A., and Pande, V. S. Atomistic folding simulations of the five-helix bundle protein lambda(6-85), *J Am Chem Soc 133*, 664-667.
95. Singhal, N., and Pande, V. S. (2005) Error analysis and efficient sampling in Markovian state models for molecular dynamics, *J Chem Phys 123*, 204909.
96. Hinrichs, N. S., and Pande, V. S. (2007) Calculation of the distribution of eigenvalues and eigenvectors in Markovian state models for molecular dynamics, *J Chem Phys 126*, 244101.
97. Lane, T. J., Bowman, G. R., Beauchamp, K., Voelz, V. A., and Pande, V. S. Markov state model reveals folding and functional dynamics in ultra-long MD trajectories, *J Am Chem Soc 133*, 18413-18419.
98. Buch, I., Giorgino, T., and De Fabritiis, G. Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations, *Proc Natl Acad Sci U S A 108*, 10184-10189.

99. Beauchamp, K. A., Bowman, G. R., Lane, T. J., Maibaum, L., Haque, I. S., and Pande, V. S. MSMBuilder2: Modeling Conformational Dynamics at the Picosecond to Millisecond Scale, *J Chem Theory Comput 7*, 3412-3419.
100. Shaw, D. E., Maragakis, P., Lindorff-Larsen, K., Piana, S., Dror, R. O., Eastwood, M. P., Bank, J. A., Jumper, J. M., Salmon, J. K., Shan, Y., and Wriggers, W. Atomic-level characterization of the structural dynamics of proteins, *Science 330*, 341-346.
101. Lindorff-Larsen, K., Piana, S., Dror, R. O., and Shaw, D. E. How fast-folding proteins fold, *Science 334*, 517-520.
102. Rao, F., and Karplus, M. Protein dynamics investigated by inherent structure analysis, *Proc Natl Acad Sci U S A 107*, 9152-9157.
103. Chodera, J. D., and Pande, V. S. The social network (of protein conformations), *Proc Natl Acad Sci U S A 108*, 12969-12970.
104. Ferguson, A. L., Panagiotopoulos, A. Z., Debenedetti, P. G., and Kevrekidis, I. G. Integrating diffusion maps with umbrella sampling: application to alanine dipeptide, *J Chem Phys 134*, 135103.
105. Garcia, A. E. (1992) Large-amplitude nonlinear motions in proteins, *Phys Rev Lett 68*, 2696-2699.
106. Amadei, A., Linssen, A. B., and Berendsen, H. J. (1993) Essential dynamics of proteins, *Proteins 17*, 412-425.
107. Beck, D. A., and Daggett, V. (2007) A one-dimensional reaction coordinate for identification of transition states from explicit solvent P(fold)-like calculations, *Biophys J 93*, 3382-3391.
108. Best, R. B., and Hummer, G. Coordinate-dependent diffusion in protein folding, *Proc Natl Acad Sci U S A 107*, 1088-1093.
109. Best, R. B., and Hummer, G. (2005) Reaction coordinates and rates from transition paths, *Proc Natl Acad Sci U S A 102*, 6732-6737.
110. Cho, S. S., Levy, Y., and Wolynes, P. G. (2006) P versus Q: structural reaction coordinates capture protein folding on smooth landscapes, *Proc Natl Acad Sci U S A 103*, 586-591.
111. Das, P., Moll, M., Stamati, H., Kavraki, L. E., and Clementi, C. (2006) Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction, *Proc Natl Acad Sci U S A 103*, 9885-9890.
112. Dinner, A. R., Sali, A., Smith, L. J., Dobson, C. M., and Karplus, M. (2000) Understanding protein folding via free-energy surfaces from theory and experiment, *Trends Biochem Sci 25*, 331-339.
113. Juraszek, J., and Bolhuis, P. G. (2008) Rate constant and reaction coordinate of Trp-cage folding in explicit water, *Biophys J 95*, 4246-4257.
114. Prentiss, M. C., Wales, D. J., and Wolynes, P. G. (2008) Protein structure prediction using basin-hopping, *J Chem Phys 128*, 225106.
115. Sali, A., Shakhnovich, E., and Karplus, M. (1994) How does a protein fold?, *Nature 369*, 248-251.
116. Weinkam, P., Romesberg, F. E., and Wolynes, P. G. (2009) Chemical frustration in the protein folding landscape: grand canonical ensemble simulations of cytochrome c, *Biochemistry 48*, 2394-2402.

117. Stamati, H., Clementi, C., and Kavraki, L. E. Application of nonlinear dimensionality reduction to characterize the conformational landscape of small peptides, *Proteins 78*, 223-235.
118. Ferguson A., P. A., Kevrekidis I., Debenedetti P. (2011) Nonlinear dimensionality reduction in molecular simulation: The diffusion map approach, *Chemical Physics Letters 509*, 1-11.
119. Huang, X., Yao, Y., Bowman, G. R., Sun, J., Guibas, L. J., Carlsson, G., and Pande, V. S. Constructing multi-resolution markov state models (msms) to elucidate RNA hairpin folding mechanisms, *Pac Symp Biocomput*, 228-239.
120. Bowman, G. R., Huang, X., and Pande, V. S. Network models for molecular kinetics and their initial applications to human health, *Cell Res 20*, 622-630.
121. Bowman, G. R., and Pande, V. S. Protein folded states are kinetic hubs, *Proc Natl Acad Sci U S A 107*, 10890-10895.
122. Voelz, V. A., Bowman, G. R., Beauchamp, K., and Pande, V. S. Molecular simulation of ab initio protein folding for a millisecond folder NTL9(1-39), *J Am Chem Soc 132*, 1526-1528.
123. Beauchamp, K. A., Ensign, D. L., Das, R., and Pande, V. S. Quantitative comparison of villin headpiece subdomain simulations and triplet-triplet energy transfer experiments, *Proc Natl Acad Sci U S A 108*, 12734-12739.
124. Zhuang, W., Cui, R. Z., Silva, D. A., and Huang, X. Simulating the T-jump-triggered unfolding dynamics of trpzip2 peptide and its time-resolved IR and two-dimensional IR signals using the Markov state model approach, *J Phys Chem B 115*, 5415-5424.
125. Noe, F., and Fischer, S. (2008) Transition networks for modeling the kinetics of conformational change in macromolecules, *Curr Opin Struct Biol 18*, 154-162.
126. Pande, V. S., Beauchamp, K., and Bowman, G. R. Everything you wanted to know about Markov State Models but were afraid to ask, *Methods 52*, 99-105.
127. Bacallado, S., Chodera, J. D., and Pande, V. (2009) Bayesian comparison of Markov models of molecular dynamics with detailed balance constraint, *J Chem Phys 131*, 045106.
128. Kortemme, T., Morozov, A. V., and Baker, D. (2003) An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein-protein complexes, *J Mol Biol 326*, 1239-1259.
129. Blum, B., Jordan, M. I., and Baker, D. Feature space resampling for protein conformational search, *Proteins 78*, 1583-1593.
130. Lange, O. F., and Baker, D. Resolution-adapted recombination of structural features significantly improves sampling in restraint-guided structure calculation, *Proteins 80*, 884-895.
131. Gonzalez, T. F. (1985) Clustering to Minimize the Maximum Intercluster Distance, *Theoretical Computer Science 38*, 293-306.
132. MacQueen, J. (1967) Some Methods for classification and Analysis of Multivariate Observations, *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability 1*, 281-297

.

133. Bowman, G. R., Huang, X., and Pande, V. S. (2009) Using generalized ensemble simulations and Markov state models to identify conformational states, *Methods 49*, 197-201.

134. Singhal, N., Snow, C. D., and Pande, V. S. (2004) Using path sampling to build better Markovian state models: predicting the folding rate and mechanism of a tryptophan zipper beta hairpin, *J Chem Phys 121*, 415-425.
135. Berezhkovskii, A., Hummer, G., and Szabo, A. (2009) Reactive flux and folding pathways in network models of coarse-grained protein dynamics, *J Chem Phys 130*, 205102.
136. Rao, F., and Caflisch, A. (2004) The protein folding network, *J Mol Biol 342*, 299-306.
137. Deechongkit, S., Nguyen, H., Powers, E. T., Dawson, P. E., Gruebele, M., and Kelly, J. W. (2004) Context-dependent contributions of backbone hydrogen bonding to beta-sheet folding energetics, *Nature 430*, 101-105.
138. Kellogg, E. H., Lange, O. F., and Baker, D. Evaluation and optimization of discrete state models of protein folding, *J Phys Chem B 116*, 11405-11413.
139. Dunbrack, R. L., Jr., and Cohen, F. E. (1997) Bayesian statistical analysis of protein side-chain rotamer preferences, *Protein Sci 6*, 1661-1681.
140. Kabsch, W., and Sander, C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features, *Biopolymers 22*, 2577-2637.