bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

PyFolding: An open-source software package for graphing, analysis and simulation of thermodynamic and kinetic models of protein folding

Alan R. Lowe[1,2,3]*, Albert Perez-Riba[4], Laura S. Itzhaki[4] & Ewan R.G. Main[5]*

[1] London Centre for Nanotechnology
17-19 Gordon Street, London
WC1H 0AH, UK

[2] Structural & Molecular Biology, University College London,
Gower Street, London,
WC1E 6BT, UK

[3] Department of Biological Sciences, Birkbeck College, University of London
Malet Street, London
WC1E 7HX, UK

[4] Department of Pharmacology, University of Cambridge
Tennis Court Road, Cambridge
CB2 1PD, UK

[5] School of Biological and Chemical Sciences, Queen Mary University of London
Mile End Road, London
E1 4NS, UK

*corresponding authors (ARL: a.lowe@ucl.ac.uk, ERGM: e.main@qmul.ac.uk)

Figure 1: Illustrative Pyfolding outputs for fitting equilibrium and kinetic datasets of: (A) the two-state folding FKBP12 protein (1) and (B) the 3-state folding thermophilic AR protein (tANK) identified in the archaeon *Thermoplasma* (2). Both show three graphs, the first is the equilibrium chemical denaturation, the second is the chevron plot and the third is the residuals for the fit of the chevron plot. In (A) the fits shown are to two-state folding models (both equilibrium and kinetic). In (B) fits shown are to three-state folding models (both equilibrium and kinetic - SI Jupyter Notebook 1). For the kinetic three state-model the multiple kinetic phases of the chevron plot are fitted using two linked equations describing the slow and fast phases (SI Jupyter Notebook 4).

Figure 2: Illustrative PyFolding outputs for global fitting of GuHCl-induced equilibrium unfolding experiments of series of single-helix deletion CTPRn proteins to a heteropolymer Ising model (3). Each output shows (A) the parameters obtained with error and correlation coefficient of the fit of the data, (B) graphical representation of the topology used to fit the data, (D) the graphs of the fitted data, (E) the graph of the first derivative of the fit function for each curve and (F) graph of the denaturant dependence of each subunit used.

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

[Abstract]

Our understanding of how proteins find and adopt their functional three-dimensional structure has largely arisen through experimental studies of the denaturant- and primary sequence- dependence of protein stability and the kinetics of folding. For many years, curve fitting software packages have been heavily utilized to fit simple models to these data. Although such software packages are easy to use for simple functions, they are often expensive and provide substantial impediments to applying more complex models or for the analysis of large datasets. Moreover, over the past decade, increasingly sophisticated analytical models have been generated, but without simple tools to enable routine analysis. Consequently, users have needed to generate their own tools or otherwise find willing collaborators. Here we present *PyFolding*, a free, open source, and extensible Python framework for the analysis and modeling of experimental protein folding and thermodynamic data. To demonstrate the utility of *PyFolding*, we provide examples of complex analysis: (i) multi-phase kinetic folding data fitted to linked equations and (ii) thermodynamic equilibrium data from consensus designed repeat proteins to both homo- and heteropolymer variants of the Ising model. Example scripts to perform these and other operations are supplied with the software. Further, we show that *PyFolding* can be used in conjunction with Jupyter notebooks as an easy way to share methods and analysis for publication and amongst research teams.

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

[Introduction]

The last decade has seen a shift in the analysis of experimental protein folding and thermodynamic stability data from the fitting of individual datasets using simple models to more and more complex models employed using global optimization over multiple large datasets [examples include Refs: (3-21)]. This shift in focus has required moving from user-friendly, but expensive software packages to bespoke solutions developed in computing environments such as MATLAB and Mathematica or by using in-house solutions [examples include: (3, 6, 12, 21, 22)]. However, as these methods of analysis have become more essential, simple curve fitting software no longer provides sufficient flexibility to implement the models. Thus, there is increasingly a need for substantially more computational expertise than previously required. In this respect the protein folding field contrasts with other fields, for example x-ray crystallography, where free or inexpensive and user-friendly interfaces and analysis packages have been developed (23).

Here we present *PyFolding*, a free, open-source and extensible framework for analysing and modelling protein folding kinetics and thermodynamic stability. The software, coupled with the supplied models / Jupyter (iPython) notebooks, can be used by researchers with less programming expertise to access more complex models/analyses and share their work with others. Moreover, *PyFolding* also enables researchers to automate the time-consuming process of combinatorial calculations, fitting data to multiple models or multiple models to specific data. To demonstrate these and other functions we present a number of examples as Jupyter notebooks. This enables novice users to simply replace the data path and rerun for their systems. The Jupyter notebooks provided also show how *PyFolding* provides an easy way to share analysis for publication and amongst research teams.

[Results & Discussion]

PyFolding is distributed as a lightweight, open-source Python library through *github* and can be downloaded with instructions for installation from the authors' site[1]. *PyFolding* has several dependencies, requiring Numpy, Scipy and Matplotlib. These are now conveniently packaged in several Python frameworks, enabling easy installation of *PyFolding* even for those who have never used Python before (described in the "Setup.md" file of *PyFolding*). As part of *PyFolding*, we have

[1] https://github.com/quantumjot/PyFolding

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

provided many commonly used folding models, such as two- and three-state equilibrium folding and various equivalent kinetic variations, as standard (S.I Jupyter notebook 1-4). Functions and models themselves are open source and are thus available for inspection or modification by both reviewers and authors. Moreover, due to the open source nature, users can introduce new functionality by adding new models into the library building upon the template classes provided.

Fitting and evaluation of typical folding models within *PyFolding*: *PyFolding* uses a hierarchical representation of data internally. Proteins exist as objects that can have metadata as well as multiple sets of kinetic and thermodynamic data associated with them. Input data such as chevron plots or equilibrium denaturation curves can be supplied as comma separated value files (.CSV). Once loaded, each dataset is represented in *PyFolding* as an object, associating the data with numerous common calculations. Models are represented as functions that can be associated with the data objects you wish to fit. As such, datasets can have multiple models and *vice versa* enabling automated fitting and evaluation (S.I Jupyter notebooks 1-3). Parameter estimation for simple (non-Ising) models is performed using the Levenberg–Marquardt non-linear least-mean-squares optimization algorithm to optimize the objective function [as implemented in SciPy (24)]. The output variables (with standard error) and fit of the model to the dataset (with R^2 coefficient of determination) can be viewed within *PyFolding* and/or the fit function and parameters written out as a CSV file for plotting in your software of choice (Figure 1 & S.I Jupyter notebook 1-3). Importantly, by representing proteins as objects, containing both kinetic and equilibrium datasets, *PyFolding* enables users to perform and automate higher-level calculations such as Phi-value analysis (25, 26), which can be tedious and time-consuming to perform otherwise (S.I Jupyter notebook 3). Moreover, users can define their own calculations so that more complex data analysis can be performed. For example, Figure 1B and S.I Jupyter notebook 4 shows how multiple kinetic phases of a chevron plot (fast and slow rate constants of folding) can be fitted to two linked equations describing the slow and fast phases of a 3-state folding regime. We believe that this type of fitting is extremely difficult to achieve with the commercial curve fitting software commonly employed for analysing these data, owing to the complexity of parameter sharing amongst different models.

More "complex" fitting, evaluation and simulations using the Ising Model: Ising models are statistical thermodynamic "nearest-neighbour" models that were initially developed for ferromagnetism (27, 28). Subsequently, they have been used with great success in both biological

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

and non-biological systems to describe order-disorder transitions (12). Within the field of protein folding and design they have been used in a number of instances to model phenomena such as helix to coil transitions, beta-hairpin formation, prediction of protein folding rates/thermodynamics and with regards to the postulation of downhill folding (6, 12, 20, 29-34). Most recently two types of one-dimensional (1-D) variants have been used to probe the equilibrium and kinetic un/folding of repeat proteins (3, 12, 17, 21, 22, 35, 36). The most commonly used, and mathematically less complex, has been the 1-D homopolymer model (also called a homozipper). Here, each arrayed element of a protein is treated as an identical, equivalent independently folding unit, with interactions between units via their interfaces. Analytical partition functions describing the statistical properties of this system can be written. By globally fitting this model to, for example, chemical denaturation curves for a series of proteins that differ only by their number of identical units, the intrinsic energy of a repeated unit and the interaction energy between the folded units can be delineated. However, this simplified model cannot describe the majority of naturally occurring proteins where subunits differ in their stabilities, and varying topologies and/or non-canonical interfaces exist. In these cases, a more sophisticated and mathematically more complex heteropolymer Ising model must be used. Here the partition functions required to fit the data are dependent on the topology of interacting units and thus are unique for each analysis.

At present, there is no freely available software that can globally fit multiple folding datasets to a heteropolymer Ising model, and only a few that can adequately implement a homopolymer Ising model. Therefore, most research groups have had to develop bespoke solutions to enable analysis of their data (3, 21, 22, 35, 36). Significantly, in *PyFolding* we have implemented methods to enable users to easily fit datasets of proteins with different topologies to both the homozipper and heteropolymer Ising models. To achieve this goal *PyFolding* presents a flexible framework for defining any non-degenerate 1-D protein topology using a series of primitive protein folding "domains/modules". Users define their proteins' 1-D topology from these "domains/modules" (S.I Jupyter notebook 5-6). *PyFolding* will then automatically calculate the correct partition function for the defined topology, using the matrix formulation of the model [as previously described (12)], and globally fit the equations to the data as required (S.I Jupyter notebook 5-6). The same framework also enables users to simulate the effect of changing the topology, a feature that is of great interest to those engaged in rational protein design (S.I Jupyter notebook 7).

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

To determine a globally optimal set of parameters that minimises the difference between the experimental datasets and the simulated unfolding curves, *PyFolding* uses the stochastic differential evolution optimisation algorithm (37) implemented in SciPy (24). In practice, experimental datasets may not adequately constrain parameters during optimisation of the objective function, despite yielding an adequate curve fit to the data. It is therefore essential to estimate the parameter errors to verify the validity of any topologies used in the model. In general, estimating errors for the parameters in heteropolymer models is a complex problem, owing to the method of optimisation used. Interestingly, Barrick and coworkers used Bootstrap analysis to evaluate parameter confidence intervals (12). However, many of the published studies either do not describe how error margins were determined or simply list the error between the data and curve fit. In *PyFolding* we have provided estimates of the errors by calculating a covariance matrix of the fitted parameters from the numerical approximation of the Jacobian matrix resulting from a final least-squares minimisation of the fit. In evaluating the determinant of the Jacobian as well as the estimated errors it is possible to assess the quality of the model.

As with the simpler models, *PyFolding* can be used to visualise the global minimum output variables (with standard errors as above) and the fit of the model to the dataset (with R^2 coeff. of determination) (Figure 2 & S.I Jupyter notebook 5-6). The output can also be exported as a CSV file for plotting in your software of choice. In addition, *PyFolding* outputs a graphical representation of the topology used to fit the data and a graph of the denaturant dependence of each subunit used (Figure 2). Thus, *PyFolding* enables non-experts to create and analyse protein folding datasets with either a homopolymer or heteropolymer Ising model for any reasonable 1-D protein topology. Moreover, once the 1-D topology of your protein has been defined, *PyFolding* can also be used to simulate and thereby predict folding behavior of both the whole protein and the sub-units that it has been composed of (S.I Jupyter notebook 7). In principle, this type of approach could be extended to higher dimensional topologies, thus providing a framework to enable rational protein design.

[Conclusion]

Here we have shown that *PyFolding,* in conjunction with Jupyter notebooks, enables researchers with minimal programming expertise the ability to fit both “typical” and complex models to their thermodynamic and kinetic protein folding data. The software is free and can be used to both

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

analyse and simulate data with models/analyses that expensive commercial user-friendly options cannot. In particular, we have incorporated the ability to fit and simulate equilibrium unfolding experiments with user defined protein topologies, using a matrix formulation of the 1-D heteropolymer Ising model. This aspect of *PyFolding* will be of particular interest to groups working on protein folds composed of repetitive motifs such as Ankyrin repeats and TPRs, given that these proteins are increasingly being used as novel antibody therapeutics (38-41) and biomaterials (42-47). Further, as analysis can be performed in Jupyter notebooks, it enables novice researchers to easily use the software and for groups to share data and methods. Finally, due to *PyFolding*'s extensible framework, it could straightforwardly be extended to enable fitting and modelling of other systems or phenomena such as protein-protein and other protein- binding interactions.

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

[Acknowledgements]

We would like to thank Dr. Jonathan Phillips for his work with the heteropolymer Ising model insightful discussion, helpful comments and suggestions. LSI acknowledges the support of a Senior Fellowship from the UK Medical Research Foundation. AP was supported by a BBSRC Doctoral Training Programme scholarship and an Oliver Gatty Studentship.

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

[References]

1. Main, E. R., K. F. Fulton, and S. E. Jackson. 1999. Folding pathway of FKBP12 and characterisation of the transition state. J Mol Biol 291:429-444.
2. Low, C., U. Weininger, P. Neumann, M. Klepsch, H. Lilie, M. T. Stubbs, and J. Balbach. 2008. Structural insights into an equilibrium folding intermediate of an archaeal ankyrin repeat protein. Proc Natl Acad Sci U S A 105:3779-3784.
3. Millership, C., J. J. Phillips, and E. R. G. Main. 2016. Ising Model Reprogramming of a Repeat Protein's Equilibrium Unfolding Pathway. J Mol Biol 428:1804-1817.
4. Jackson, S. E., and A. R. Fersht. 1991. Folding of chymotrypsin inhibitor 2. 1. Evidence for a two-state transition. Biochemistry 30:10428-10435.
5. Schatzle, M., and T. Kiefhaber. 2006. Shape of the free energy barriers for protein folding probed by multiple perturbation analysis. J Mol Biol 357:655-664.
6. Naganathan, A. N., and V. Munoz. 2014. Thermodynamics of downhill folding: multi-probe analysis of PDD, a protein that folds over a marginal free energy barrier. Journal of Physical Chemistry. B 118:8982-8994.
7. Ferreiro, D. U., and P. G. Wolynes. 2008. The capillarity picture and the kinetics of one-dimensional protein folding. Proc Natl Acad Sci U S A 105:9853-9854.
8. Barrick, D., D. U. Ferreiro, and E. A. Komives. 2008. Folding landscapes of ankyrin repeat proteins: experiments meet theory. Curr Opin Struct Biol 18:27-34.
9. DeVries, I., D. U. Ferreiro, I. E. Sanchez, and E. A. Komives. 2011. Folding kinetics of the cooperatively folded subdomain of the IkappaBalpha ankyrin repeat domain. J Mol Biol 408:163-176.
10. Maxwell, K. L., D. Wildes, A. Zarrine-Afsar, M. A. De Los Rios, A. G. Brown, C. T. Friel, L. Hedberg, J. C. Horng, D. Bona, E. J. Miller, A. Vallee-Belisle, E. R. Main, F. Bemporad, L. Qiu, K. Teilum, N. D. Vu, A. M. Edwards, I. Ruczinski, F. M. Poulsen, B. B. Kragelund, S. W. Michnick, F. Chiti, Y. Bai, S. J. Hagen, L. Serrano, M. Oliveberg, D. P. Raleigh, P. Wittung-Stafshede, S. E. Radford, S. E. Jackson, T. R. Sosnick, S. Marqusee, A. R. Davidson, and K. W. Plaxco. 2005. Protein folding: defining a "standard" set of experimental conditions and a preliminary kinetic data set of two-state proteins. Protein Science 14:602-616.
11. Wensley, B. G., S. Batey, F. A. Bone, Z. M. Chan, N. R. Tumelty, A. Steward, L. G. Kwa, A. Borgia, and J. Clarke. 2010. Experimental evidence for a frustrated energy landscape in a three-helix-bundle protein family. Nature 463:685-688.
12. Aksel, T., and D. Barrick. 2009. Analysis of repeat-protein folding using nearest-neighbor statistical mechanical models. Methods in Enzymology 455:95-125.
13. Mallam, A. L., and S. E. Jackson. 2007. A comparison of the folding of two knotted proteins: YbeA and YibK. J Mol Biol 366:650-665.
14. Scott, K. A., L. G. Randles, and J. Clarke. 2004. The folding of spectrin domains II: phi-value analysis of R16. J Mol Biol 344:207-221.
15. Hutton, R. D., J. Wilkinson, M. Faccin, E. M. Sivertsson, A. Pelizzola, A. R. Lowe, P. Bruscolini, and L. S. Itzhaki. 2015. Mapping the Topography of a Protein Energy Landscape. J Am Chem Soc 137:14610-14625.
16. Tsytlonok, M., P. O. Craig, E. Sivertsson, D. Serquera, S. Perrett, R. B. Best, P. G. Wolynes, and L. S. Itzhaki. 2013. Complex energy landscape of a giant repeat protein. Structure 21:1954-1965.
17. Javadi, Y., and E. R. Main. 2009. Exploring the folding energy landscape of a series of designed consensus tetratricopeptide repeat proteins. Proc Natl Acad Sci U S A 106:17383-17388.

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

18. Lowe, A. R., and L. S. Itzhaki. 2007. Biophysical characterisation of the small ankyrin repeat protein myotrophin. J Mol Biol 365:1245-1255.
19. Xu, M., O. Beresneva, R. Rosario, and H. Roder. 2012. Microsecond folding dynamics of apomyoglobin at acidic pH. Journal of Physical Chemistry. B 116:7014-7025.
20. Garcia-Mira, M. M., M. Sadqi, N. Fischer, J. M. Sanchez-Ruiz, and V. Munoz. 2002. Experimental identification of downhill protein folding. Science 298:2191-2195.
21. Aksel, T., A. Majumdar, and D. Barrick. 2011. The contribution of entropy, enthalpy, and hydrophobic desolvation to cooperativity in repeat-protein folding. Structure 19:349-360.
22. Kajander, T., A. L. Cortajarena, E. R. Main, S. G. Mochrie, and L. Regan. 2005. A new folding paradigm for repeat proteins. Journal of the American Chemical Society 127:10188-10190.
23. Winn, M. D., C. C. Ballard, K. D. Cowtan, E. J. Dodson, P. Emsley, P. R. Evans, R. M. Keegan, E. B. Krissinel, A. G. Leslie, A. McCoy, S. J. McNicholas, G. N. Murshudov, N. S. Pannu, E. A. Potterton, H. R. Powell, R. J. Read, A. Vagin, and K. S. Wilson. 2011. Overview of the CCP4 suite and current developments. Acta Crystallogr D Biol Crystallogr 67:235-242.
24. Jones, E., T. Oliphant, P. Peterson, and others. 2001. SciPy: Open source scientific tools for Python.
25. Serrano, L., A. Matouschek, and A. R. Fersht. 1992. The folding of an enzyme. III. Structure of the transition state for unfolding of barnase analysed by a protein engineering procedure. J Mol Biol 224:805-818.
26. Fersht, A. R., A. Matouschek, and L. Serrano. 1992. The folding of an enzyme. I. Theory of protein engineering analysis of stability and pathway of protein folding. J Mol Biol 224:771-782.
27. Brush, S. G. 1967. History of the Lenz-Ising Model. Reviews of Modern Physics 39:883-893.
28. Niss, M. 2005. History of the Lenz-Ising model 1920-1950: From ferromagnetic to cooperative phenomena. Arch Hist Exact Sci 59:267-318.
29. Zimm, B. H., and J. K. Bragg. 1959. Theory of the Phase Transition between Helix and Random Coil in Polypeptide Chains. The Journal of Chemical Physics 31:526-535.
30. Munoz, V., P. A. Thompson, J. Hofrichter, and W. A. Eaton. 1997. Folding dynamics and mechanism of beta-hairpin formation. Nature 390:196-199.
31. Munoz, V., and W. A. Eaton. 1999. A simple model for calculating the kinetics of protein folding from three-dimensional structures. Proc Natl Acad Sci U S A 96:11311-11316.
32. Kubelka, J., E. R. Henry, T. Cellmer, J. Hofrichter, and W. A. Eaton. 2008. Chemical, physical, and theoretical kinetics of an ultrafast folding protein. Proc Natl Acad Sci U S A 105:18655-18662.
33. Kubelka, G. S., and J. Kubelka. 2014. Site-specific thermodynamic stability and unfolding of a de novo designed protein structural motif mapped by 13C isotopically edited IR spectroscopy. J Am Chem Soc 136:6037-6048.
34. Lai, J. K., G. S. Kubelka, and J. Kubelka. 2015. Sequence, structure, and cooperativity in folding of elementary protein structural motifs. Proc Natl Acad Sci U S A 112:9890-9895.
35. Wetzel, S. K., G. Settanni, M. Kenig, H. K. Binz, and A. Pluckthun. 2008. Folding and unfolding mechanism of highly stable full-consensus ankyrin repeat proteins. J Mol Biol 376:241-257.
36. Aksel, T., and D. Barrick. 2014. Direct observation of parallel folding pathways revealed using a symmetric repeat protein system. Biophys J 107:220-232.
37. Storn, R., and K. Price. 1997. Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11:341-359.
38. Rasool, M., A. Malik, M. Hussain, K. A. Haq, K. Butt, M. A. B. Ashraf, M. I. Naseer, M. Asif, R. Shaikh, M. Z. Mustafa, Q. Alam, G. Rasool, W. Ahmad, A. Haque, and M. A. Kamal. 2017.

bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

DARPins Bioengineering and its Theranostic Approaches: Emerging Trends in Protein Engineering. Curr Pharm Design 23:1610-1615.

39. Jost, C., and A. Pluckthun. 2014. Engineered proteins with desired specificity: DARPins, other alternative scaffolds and bispecific IgGs. Curr Op Struct Biol 27:102-112.
40. Ernst, P., and A. Pluckthun. 2017. Advances in the design and engineering of peptide-binding repeat proteins. Biol Chem 398:23-29.
41. Cortajarena, A. L., F. Yi, and L. Regan. 2008. Designed TPR modules as novel anticancer agents. ACS Chem Biol 3:161-166.
42. Sawyer, N., E. B. Speltz, and L. Regan. 2013. NextGen protein design. Biochem Soc Trans 41:1131-1136.
43. Main, E. R., J. J. Phillips, and C. Millership. 2013. Repeat protein engineering: creating functional nanostructures/biomaterials from modular building blocks. Biochem Soc Trans 41:1152-1158.
44. Grove, T. Z., L. Regan, and A. L. Cortajarena. 2013. Nanostructured functional films from engineered repeat proteins. Journal of the Royal Society, Interface 10:20130051.
45. Phillips, J. J., C. Millership, and E. R. G. Main. 2012. Fibrous Nanostructures from the Self-Assembly of Designed Repeat Protein Modules. Angew Chem Int Edit 51:13132-13135.
46. Grove, T. Z., and L. Regan. 2012. New materials from proteins and peptides. Curr Opin Struct Biol 22:451-456.
47. Grove, T. Z., J. Forster, G. Pimienta, E. Dufresne, and L. Regan. 2012. A modular approach to the design of protein-based smart gels. Biopolymers 97:508-517.

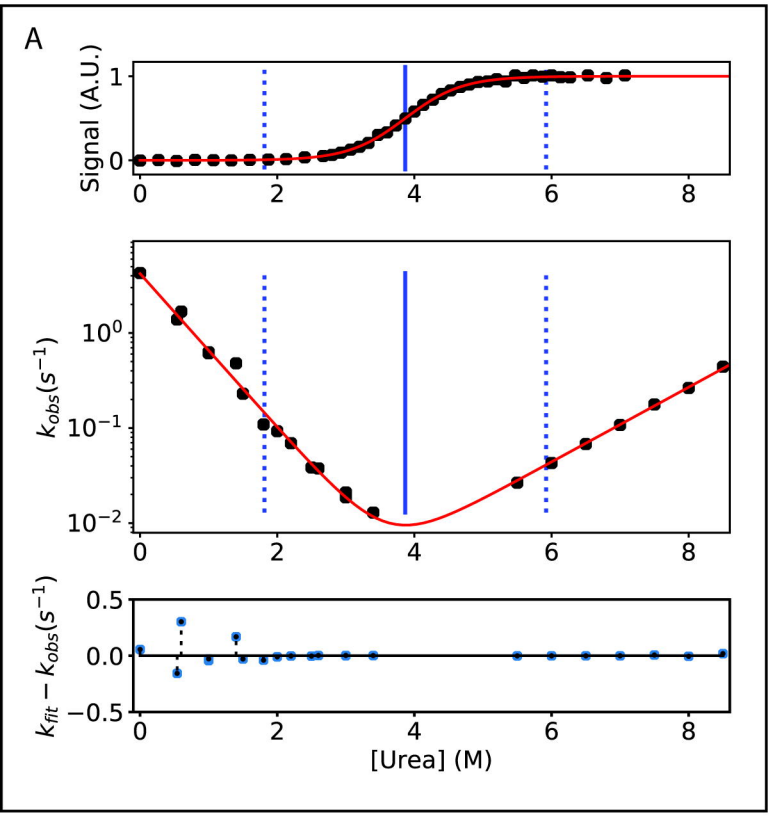
bioRxiv preprint doi: https://doi.org/10.1101/191593; this version posted September 20, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

[Figures]

Figure 1: Illustrative Pyfolding outputs for fitting equilibrium and kinetic datasets of: (A) the two-state folding FKBP12 protein (1) and (B) the 3-state folding thermophilic AR protein (tANK) identified in the archaeon *Thermoplasma* (2). Both show three graphs, the first is the equilibrium chemical denaturation, the second is the chevron plot and the third is the residuals for the fit of the chevron plot. In (A) the fits shown are to two-state folding models (both equilibrium and kinetic). In (B) fits shown are to three-state folding models (both equilibrium and kinetic - SI Jupyter Notebook 1). For the kinetic three state-model the multiple kinetic phases of the chevron plot are fitted using two linked equations describing the slow and fast phases (SI Jupyter Notebook 4).

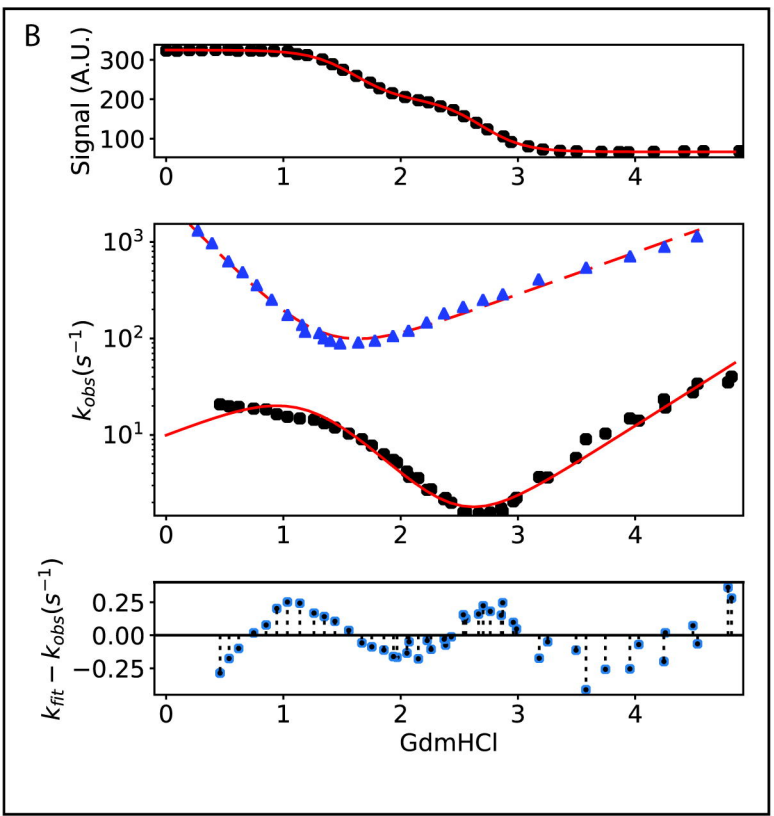
Figure 2: Illustrative PyFolding outputs for global fitting of GuHCl-induced equilibrium unfolding experiments of series of single-helix deletion CTPRn proteins to a heteropolymer Ising model (3). Each output shows (A) the parameters obtained with error and correlation coefficient of the fit of the data, (B) graphical representation of the topology used to fit the data, (D) the graphs of the fitted data, (E) the graph of the first derivative of the fit function for each curve and (F) graph of the denaturant dependence of each subunit used.

Figure 1



Equilibrium Model: TwoStateEquilibrium
m: 1.43404 ± 0.00003
d50: 3.86730 ± 0.00001
Folding midpoint: 3.87 M
R^2: 1.00

Kinetic Model: TwoStateChevron
Fit Standard Error: 0.02
kf: 4.22e+00 ± 6.25e-03
mf: 1.86e+00 ± 8.23e-04
ku: 1.92e-04 ± 1.26e-06
mu: 9.06e-01 ± 9.35e-04
R^2: 0.99



Equilibrium Model: ThreeStateEquilibrium
Y_N: 324.75027 ± 0.10551
Y_I: 197.11446 ± 0.50639
Y_D: 66.56156 ± 0.10796
DGni: 4.77711 ± 0.03063
m_ni: 2.96838 ± 0.02120
DGid: 8.33336 ± 0.06351
m_id: 3.09530 ± 0.02218
R^2: 1.00

Kinetic Model: ThreeStateFastPhaseChevron
Fit Standard Error: 0.02
kui: 2.65e+03 ± 8.52e+00
mui: 2.83e+00 ± 4.22e-03
kiu: 1.43e+01 ± 5.81e-02
miu: 9.99e-01 ± 1.30e-03
kif: 1.86e+03 ± 9.91e+00
mif: 2.74e+00 ± 2.74e-03
kfi: 1.06e-02 ± 5.98e-05
mfi: 1.77e+00 ± 1.47e-03
R^2: 0.97

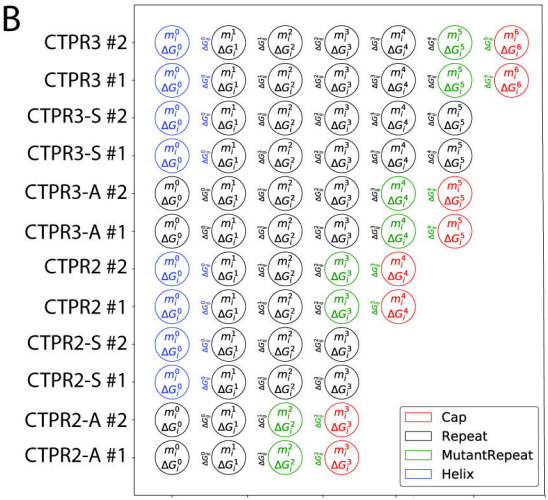
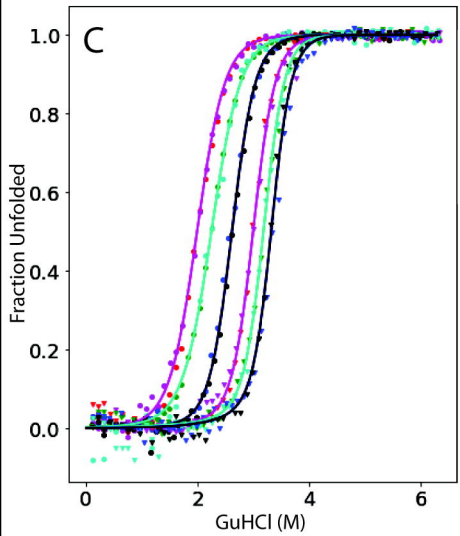
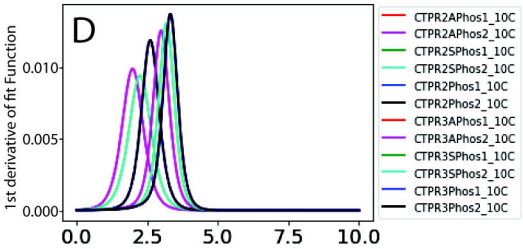
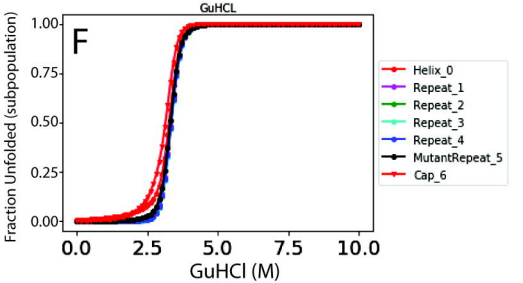
Figure 2

A **Fitting results:**

Repeat DG_i: 3.46641 ± 3.04704
Repeat DG_ij: -6.15005 ± 2.83902
Repeat m_i: -0.53090 ± 1.11790

MutantRepeat DG_i: 5.17623 ± 0.63167
MutantRepeat DG_ij: -9.96453 ± 0.31931
MutantRepeat m_i: -0.00783 ± 1.85200

Cap DG_i: 5.56471 ± 0.31994
Cap m_i: -1.25270 ± 0.67150

Helix DG_i: 6.92820 ± 1.84188
Helix DG_ij: -9.99979 ± 1.84125
Helix m_i: -0.63752 ± 7.40484

CTPR2APhos1_10C R^2: 0.99891
CTPR2APhos2_10C R^2: 0.99894
etc