

**UCSF**

**UC San Francisco Electronic Theses and Dissertations**

**Title**

Increasing the complexity of computational protein modeling methodologies for functional applications in biology

**Permalink**

<https://escholarship.org/uc/item/649785wr>

**Author**

Barlow, Kyle Andrew

**Publication Date**

2017

Peer reviewed|Thesis/dissertation

Increasing the complexity of computational protein modeling  
methodologies for functional applications in biology

by

Kyle Barlow

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Biological and Medical Informatics

in the

GRADUATE DIVISION



# Acknowledgments

This work is dedicated to the following very important people:

To my wife, Malia McPherson. Nothing I've done would have been possible without your love and support.

To my parents and sister, who nurtured my scientific interest and intellectual curiosity.

My scientific work would not have been possible without the advice and support of my mentor, Tanja Kortmme, who has always been an excellent role model, and whose crystal clear communication style and instinct and ability to find interesting and important projects I will continue to strive to emulate throughout my career.

I also acknowledge and appreciate the support of my thesis committee members, Andrej Sali and Jaime Fraser, and the rest of my colleagues in the Kortemme lab. In particular, Roland Pache, Shane Conchúir, Kale Kundert, and James Lucas, who I worked closely with on many projects, and Samuel Thompson, Pooja Suresh, and James Lucas, who were integral in the development of the flex ddG method.

Finally, I must acknowledge the support and friendship of Juan Alfredo Pérez Bermejo, Graham Heimberg, and James Webber, who helped me sequence CODa.

The text of 2 is a reprint of the material as it appears in PLOS One Conchúir et al.<sup>1</sup>. The co-authors listed in this publication assisted, directed or supervised the research that forms the basis for the dissertation/thesis.

# Increasing the complexity of computational protein modeling methodologies for functional applications in biology

Kyle Barlow

While the native states of proteins usually correspond to their free energy minimum, and can often be found with experimental techniques, predicting the folded native state of a protein computationally remains a major challenge. This is partly due to the immense conformational space a single protein sequence could potentially fold into, a space that is even larger if the protein sequence is unknown, as in the case of design. In this thesis, I evaluate the performance of current state-of-the-art computational protein structure prediction and design methods (as implemented in the Rosetta macromolecular modeling software suite) on the following commonly encountered modeling problems: estimation of energetic effects of mutations (protein stability ( $\Delta\Delta G$ ) and change in protein-protein interface binding energy post-mutation); (2) protein design predictions (native sequence recovery, evolutionary profile recovery, sequence covariation recovery, and prediction of recognition specificity); and (3) protein structure prediction (loop modeling). I assemble curated benchmark data for each of these prediction problems that can be used for future evaluation of method performance

on a common data set.

As the prior state-of-the-art methods for prediction of change in protein-protein interface binding energy post-mutation were not very effective for predicting mutations to side chains other than alanine, I created a new, more general Rosetta method for prediction of these cases. This “flex ddG” method generates and utilizes ensembles of diverse protein conformational states (generated with “backrub” sampling) to predict interface  $\Delta\Delta G$  values. Flex ddG is effective for prediction of change in binding free energy post-mutation for mutations to all amino acids, including mutations to alanine, and is particularly effective (when compared to prior methods) for cases of small side chain to large side chain mutations. I show that the method succeeds in these cases due to increased sampling of diverse conformational states, as performance improves (to a threshold) as more diverse states are sampled.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Computational protein structure prediction and design . . . . .	2
1.2	Ways forward to improve computational protein modeling . . . . .	4
<b>2</b>	<b>A web resource for standardized benchmark datasets, metrics, and Rosetta protocols for macromolecular modeling and design</b>	<b>6</b>
2.1	Abstract . . . . .	6
2.2	Introduction . . . . .	7
2.3	Benchmarks (Methods) . . . . .	12
2.3.1	Tests estimating energetic effects of mutation . . . . .	12
2.3.2	Design tests . . . . .	17
2.3.3	Using evolutionary information . . . . .	17
2.3.4	Using large-scale experimental data . . . . .	23
2.3.5	Structure prediction tests . . . . .	25
2.3.6	Website description . . . . .	28
2.4	Discussion . . . . .	30
2.5	Acknowledgements . . . . .	31
2.6	Supporting Information . . . . .	32
<b>3</b>	<b>Flex ddG: Ensemble-based prediction of interface binding free energy upon mutation</b>	<b>33</b>

3.1	Introduction . . . . .	33
3.2	Methods . . . . .	35
3.3	Results and discussion . . . . .	39
3.3.1	Effect of averaging more structures . . . . .	41
3.3.2	Effect of changing backrub sampling steps . . . . .	44
3.3.3	Score analysis . . . . .	45
3.4	Conclusions . . . . .	46
<b>4</b>	<b>Conclusion</b>	<b>48</b>
<b>5</b>	<b>Appendix</b>	<b>49</b>
	<b>Bibliography</b>	<b>64</b>



# List of Tables

<b>3</b>	<b>Flex ddG: Ensemble-based prediction of interface binding free energy upon mutation</b>	
3.1	ZEMu dataset subset definition and composition . . . . .	36
3.2	Main results table . . . . .	40
<b>5</b>	<b>Appendix</b>	
5.1	Multiple mutations results . . . . .	49
5.2	SHA1 Git version of Rosetta used for benchmarking . . . . .	50
5.3	Flex ddG performance on antibodies . . . . .	51
5.4	REF results . . . . .	51

# List of Figures

<b>1</b>	<b>Introduction</b>	
1.1	Protein sequence/structure . . . . .	4
<b>2</b>	<b>A web resource for standardized benchmark datasets, metrics, and Rosetta protocols for macromolecular modeling and design</b>	
2.1	Types of benchmarks and protocols currently included in the web resource . . . . .	11
2.2	Comparison of occurrences of different amino acid residue types observed at buried positions between natural sequences and sequences designed with two different Rosetta energy functions . . . . .	21
2.3	Benchmark and protocols capture website . . . . .	29
2.4	Comparison of occurrences of different amino acid residues (by polarity) observed at buried and exposed positions . . . . .	32
<b>3</b>	<b>Flex ddG: Ensemble-based prediction of interface binding free energy upon mutation</b>	
3.1	Schematic of the flex ddG protocol method. . . . .	38
3.2	Rosetta vs. Experimental $\Delta\Delta G$ for Flex ddg and No backrub control . . . . .	42
3.3	Flex ddG performance vs. number of averaged structures for flex ddG . . . . .	43
<b>5</b>	<b>Appendix</b>	

5.1	Flex ddG performance vs. number of backrub steps . . . . .	50
5.2	Flex ddG performance vs. number of averaged structures for flex ddG . . . .	52
5.3	Flex ddG performance vs. number of averaged structures for ddG monomer .	53
5.4	Contour plot showing the effect of backrub sampling on the average wild-type complex score, for varying numbers of averaged models. . . . .	54
5.5	Mean backrub ensemble RMSD vs. backrub steps. . . . .	55
5.6	$\Delta\Delta G$ prediction error vs. ensemble RMSD . . . . .	55
5.7	Sigmoid fit Rosetta score function terms . . . . .	56
5.8	Interface $\Delta\Delta G$ prediction performance with sigmoid fit score function . . . .	57

# List of Source Code Listings

5.1 Flex ddg Rosetta Script implementation . . . . .	58
--	----

# Chapter 1

## Introduction

Proteins are essential to and found in all extant life on Earth. They function as enzymes, performing factory-like transformations of chemicals, provide structural support and function as machines that can move organisms, are integral to transportation and communication within and between cells, and are the basis of the ability of our immune systems to detect antigens. In fact, proteins are so important for the function of life that their blueprints are written in our DNA<sup>2</sup> using a near-universal code<sup>3,4</sup>, providing evidence for Darwin's theory that all life on Earth shares common ancestry<sup>5</sup>.

Understanding how proteins work can be undertaken using a variety of strategies, including genetic analysis<sup>6</sup>, but one of the most informative has been to determine the structure of proteins, as knowing what a machine looks like provides insight into its functionality. We have been able to determine the structures of proteins at a high resolution since the first published crystal structures of myoglobin and hemoglobin<sup>7,8</sup>. Still, there are many more known protein sequences in the universe of life than exist known structures in available databases of experimentally determined structures (88,032,926 unique sequences vs. 123,153 known structures, as of Aug. 12, 2017<sup>9,10</sup>).

Obtaining models of structures where no experimental data exists would help provide insight into the vast space of unknown protein functionality. In these cases, we can turn to

the power of computation to help determine unknown structures. Developing computational protocols for protein structure prediction provides a few advantages, including the fact that a method capable of accurately predicting protein structure can also be adapted to design entirely new protein functionalities. This ability to design proteins enables the creation of protein therapeutics potentially capable of treating currently untreatable diseases, as well as enzymes that could be able to make chemicals and fuels in a more environmentally friendly fashion.

The value of computational methods for protein structure prediction goes beyond their potential applications in design. By attempting to create a framework in which we can accurately represent, sample, and score and compare models of protein structures, we gain insight into the inner workings of the biophysics that underlie the functionality of proteins in all life.

## 1.1 Computational protein structure prediction and design

Protein structure prediction can operate on the “primary”, “secondary”, “tertiary”, or “quaternary” levels (??). While modern protein secondary structure prediction methods can achieve relatively high accuracy (of about 80%<sup>11</sup>), protein structure prediction of more detailed models remains a major challenge.

The difficulty of computational protein structure prediction can be thought of as three complementary challenges: the “sampling” problem, the “scoring” problem, and the question of “representation”.

**Representation** Choosing a manner in which to represent protein structures within the overall computational framework that fits the desired application is an oft-overlooked, but essential step in modeling. For example, software that attempted to internally represent

proteins as a collection of subatomic components such as protons, neutrons, or even quarks, would be a far more detailed representation than is required for simulating biologically relevant processes such as protein-protein interactions. Computational time would be wasted simulating the interactions of subatomic particles if the desired output can be obtained by simulating the system at a coarser level of detail. On the other hand, representing proteins as individual spheres that interact and bounce off each other like billiard balls would be too simplistic for many applications, and would not be able to simulate at the level of detail required for predicting the specificity and strength of binding interactions. In the end, a balance must be struck, and full-atom representations of protein structures are now commonly used in modeling.

**Scoring** Scoring refers to the ability of a prediction/design method to successfully rank and sort generated models of proteins (in whichever representation they are generated) in terms of stability or other desired biophysical attributes. Again, as with representation, an appropriately detailed score function should be chosen to fit the problem at hand. Score functions used in protein modeling and design have taken a variety of approaches, including modeling atomic interactions with solvent (water) both explicitly<sup>12,13</sup> or implicitly<sup>14</sup>, repulsive and attractive terms<sup>15</sup>, and knowledge-based terms calibrated based on the propensity of protein substructures to occupy various states<sup>16</sup>. If a score function is to be used in protein design, it must be fast enough to evaluate the many potential combinations and permutations of amino acids that could come together to form the entire modeled protein.

**Sampling** Many, many structural models can represent a protein sequence for even the simplest representations of protein structure<sup>17,18</sup>. The sampling problem refers to the difficulty of generating protein structural models out of this near-infinite universe of potential states. As only a finite number of models can be scored, decoy models must be generated efficiently. Sampling methods that have proven to be effective include modifying a known protein structure for new activity<sup>19,20</sup>, designing a protein backbone to fit a desired

topology<sup>21</sup>, defining “moves” that are likely to jump between stable structures<sup>22–24</sup>, and a divide-and-conquer approach that breaks proteins into “fragments” that can be sampled individually<sup>25</sup>.

## 1.2 Ways forward to improve computational protein modeling

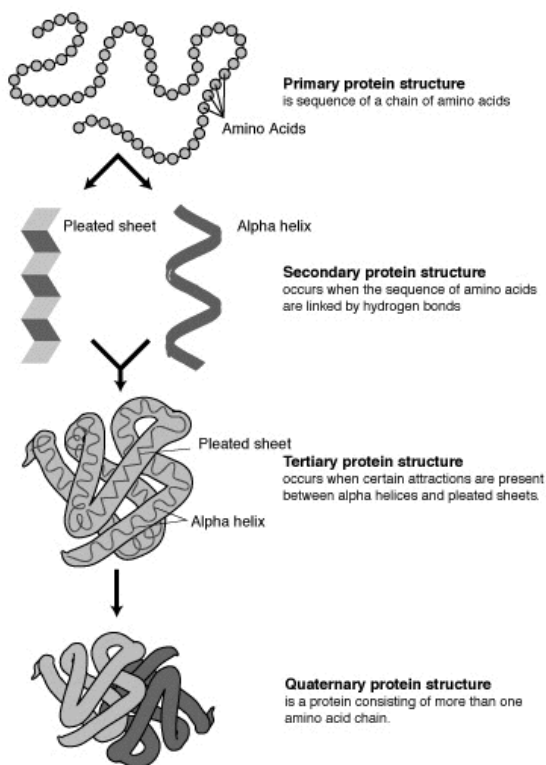


Figure 1.1: Knowledge of the primary amino acid sequence is a common input for computational structure prediction programs, which then can produce possible models of the protein’s secondary, tertiary, and quaternary structure. Figure produced by: U.S. National Institutes of Health (public domain).

Although proteins tend to “fold” into energy minima centered on the globally most stable conformation<sup>26</sup>, life takes place at non-frozen, biological temperatures. In cells, proteins are dynamic and sample an ensemble of conformations centered around free-energy minima<sup>27</sup>. Protein modeling and design software might therefore obtain improved predictions from representing proteins more complexly, allowing for conformational flexibility.

In my graduate research, I set out to develop and test sampling and representation methods that allowed for more realistic models of proteins, enabling design of new protein functions and predictions of protein structures that could better explain the inner functionality of biological systems. I have developed these methods within the

Rosetta macromolecular modeling suite, which is developed by researchers around the world,



allowing access to improved score function and sampling methods as they are developed by others.

Since the problem of scoring is inextricably linked to the problem of sampling, I needed to rigorously test the performance of new score functions on rationally created benchmark datasets, providing a foundation of known performance to build up to these more complex representations (??). I then developed a method capable of predicting the change in strength of protein-protein interactions after mutation that utilizes ensemble-based representations of protein structure (??). As I, in collaboration with my cohort of iPQB graduate students, have already shown the ability of Rosetta to provide insight into the mechanisms of how changes in strength in protein-protein interactions affect fitness in yeast<sup>28,29</sup>, I hope and expect that these developments in computational protein modeling methodology will continue to prove useful in the future for biological applications.

# Chapter 2

## A web resource for standardized benchmark datasets, metrics, and Rosetta protocols for macromolecular modeling and design

### 2.1 Abstract

The development and validation of computational macromolecular modeling and design methods depend on suitable benchmark datasets and informative metrics for comparing protocols. In addition, if a method is intended to be adopted broadly in diverse biological applications, there needs to be information on appropriate parameters for each protocol, as well as metrics describing the expected accuracy compared to experimental data. In certain disciplines, there exist established benchmarks and public resources where experts in a particular methodology are encouraged to supply their most efficient implementation of each particular benchmark. We aim to provide such a resource for protocols in macromolecular modeling and design. We present a freely accessible web resource

(<https://kortemmelab.ucsf.edu/benchmarks>) to guide the development of protocols for protein modeling and design. The site provides benchmark datasets and metrics to compare the performance of a variety of modeling protocols using different computational sampling methods and energy functions, providing a “best practice” set of parameters for each method. Each benchmark has an associated downloadable benchmark capture archive containing the input files, analysis scripts, and tutorials for running the benchmark. The captures may be run with any suitable modeling method; we supply command lines for running the benchmarks using the Rosetta software suite. We have compiled initial benchmarks for the resource spanning three key areas: prediction of energetic effects of mutations, protein design, and protein structure prediction, each with associated state-of-the-art modeling protocols. With the help of the wider macromolecular modeling community, we hope to expand the variety of benchmarks included on the website and continue to evaluate new iterations of current methods as they become available.

## 2.2 Introduction

Structure-based modeling and design of biological macromolecules have become rich areas of computational research and method development<sup>13,30-33</sup>. The accuracy of these modeling protocols on diverse applications can be assessed via use of increasingly available, high quality curated experimental datasets<sup>9,34-37</sup>. Demonstration of the utility of a new prediction or design method requires, at the very least, a proof-of-concept case that exhibits initial success. Further widespread adoption of the method requires more extensive validation: demonstrated success and careful evaluation of key limitations on multiple, diverse, test cases. This general utility can be shown through the use of a suitable benchmark set.

Even though the compilation of these benchmarks is often essential to the creation of novel computational methods, the successful application of a method can often overshadow the critical role of benchmarking during its development. Furthermore, the associated pub-

lication of a new method may not contain a description of the dataset or statistical analysis in a format that is readily usable for developers of alternate methods, creating additional obstacles for a direct comparison. Organizations such as CASP<sup>38</sup> and CAPRI<sup>39</sup> create blind prediction tests for problems in protein structure prediction, protein-protein docking, and other applications, but many questions in the field of macromolecular modeling and design could also benefit from canonical benchmarks such as those that exist for protein-protein docking<sup>37-40</sup>. To facilitate rapid, iterative development, it is convenient to make benchmarks available for retrospective testing (although it is essential to pay attention to issues of overfitting to a particular target problem, even for large and diverse datasets).

Even in cases where an effective benchmark has been defined and the efficacy of a modeling protocol has been measured and published, it may be difficult to reproduce similar results post-publication as the method evolves. Protocols in large, complex software suites, such as Rosetta, are highly dependent on core functionality. For example, a sampling algorithm may yield varying results as changes are made to its accompanying score function. Regular benchmarking to track changes in performance is desirable both when core functionality is altered and when the specific protocol has been modified directly. To determine what constitutes the best practice, a user needs access to current benchmarking results, or at the very least, clear instructions on how to benchmark against the latest version of the protocol.

Here we present a web resource (<https://kortemmelab.ucsf.edu/benchmarks>) to address some of the aforementioned difficulties associated with informative benchmarking. We define the following criteria for a benchmark set in this resource: First, the scientific question or modeling problem posed by the benchmark must be clearly defined. Second, the input dataset should contain numerous, varied test cases that cover a broad range of possible inputs a user might use in a protocol. Success is easier to find when only a small subset of potential test cases is employed; a more general set indicates a correspondingly more generally useful method, and ameliorates issues with over-fitting a method to perform well on a specific test case. To be suitable for comparison against predictions, this input data

set should be made up of experimentally validated data (we will refer to predicted data as “predictions” and experimentally determined data as “experiments”). Third, instructions on how to run each computational method should be provided with enough detail and clarity such that researchers other than the developers of a given method are able to use the resource. Finally, each benchmark set should be accompanied both by an appropriate set of defined metrics to quantify how successfully the method addresses the modeling problem and by a set of analysis tools which, given input in a defined format, computes these metrics.

We have used these guidelines to collect benchmark sets for commonly encountered problems in the following three areas (??): (1) estimation of energetic effects of mutations (protein stability ( $\Delta\Delta G$ ) and computational alanine scanning); (2) protein design predictions (native sequence recovery, evolutionary profile recovery, sequence covariation recovery, and prediction of recognition specificity); and (3) protein structure prediction (loop modeling). We also present corresponding state-of-the-art Rosetta protocols, parameters and command lines applicable to each problem. Each benchmark capture can be downloaded from the web resource either as a self-contained zip file/bundle or as a version-controlled repository. Each bundle contains the input data and documentation describing the given modeling problem, explains how the accompanying methods solve that problem, lists the metrics we use to measure success, includes the Rosetta protocol, and provides analysis scripts to generate these metrics from output data. In the sequel, we describe the technical details of the website we have created for open access and dissemination of benchmarking results.

(A)  $\Delta\Delta G$  / Alanine scanning – predicting the energetic effect of point mutations on folding or binding. REU: Rosetta energy units. The dashed line represents the best linear fit model ( $y = 0.93x + 0.43$ ). (B) Native sequence recovery – measuring the similarity between designed and native sequences for a given structure. Boxplots compare fixed to flexible backbone design performance, and designed residues that are identical to the native sequence are highlighted in yellow. (C) Sequence profile recovery – measuring the similarity between designed and natural sequence profiles of protein families. Boxplots compare

fixed to flexible backbone design performance in recovering the natural sequence profile. (D) Amino acid covariation – predicting pairs of naturally covarying residues in protein families. Boxplots compare fixed to flexible backbone design performance, and covarying pairs in the multiple sequence alignment are highlighted in green and magenta. (E) Recognition specificity – predicting the tolerated sequence space in a protein-protein interface. The sequence logos<sup>41</sup> visualize the similarities and differences between the predicted and experimentally determined sequence profiles. (F) Loop reconstruction – predicting the backbone conformation of loops in protein structures. Here the scatterplot shows a minimum in the Rosetta energy landscape for the given loop, with the five lowest energy models shown in yellow and the one closest to the experimentally determined (native) structure highlighted in red.

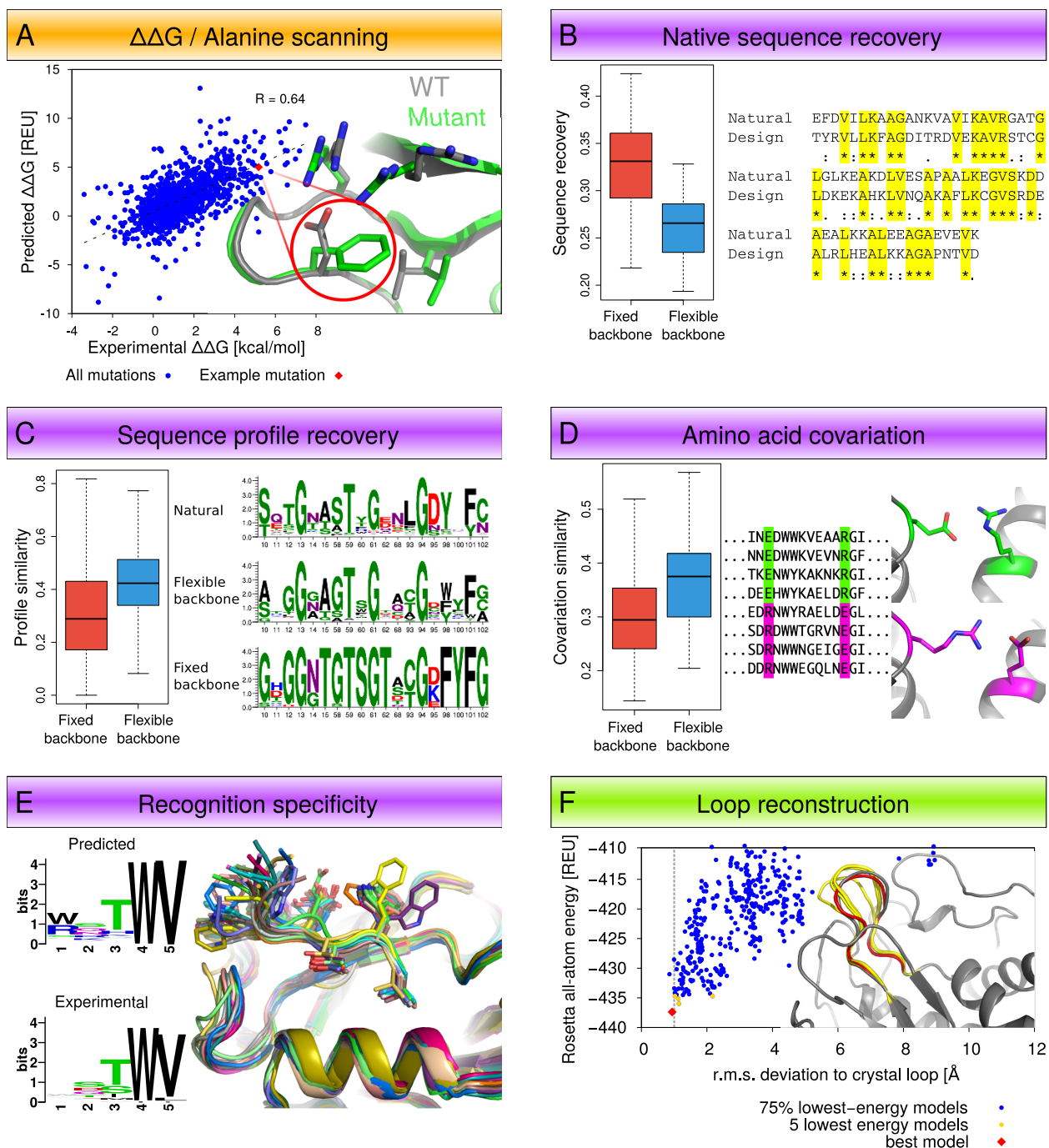


Figure 2.1: Types of benchmarks and protocols currently included in the web resource. Tests estimating energetic effects of mutation (orange, A), design tests (purple, B-E) and structure prediction tests (green, F).

## 2.3 Benchmarks (Methods)

The web resource currently contains benchmark captures, Rosetta protocols, and performance information for five different benchmarks, which we have grouped below in three different areas (??): (1) tests estimating energetic effects of mutations, (2) design tests, and (3) structure prediction tests. Each subsection describes, for each benchmark, its purpose (the modeling problem addressed), the benchmark dataset, a Rosetta protocol addressing the modeling problem, metrics of success, key results, and notes on limitations and caveats.

### 2.3.1 Tests estimating energetic effects of mutation

#### Protein stability ( $\Delta\Delta G$ )

**Purpose of this test** The purpose of this benchmark is to predict the change in stability ( $\Delta\Delta G$ ) of a monomeric protein caused by single point mutations (??A). The predicted stability change is given as the difference in predicted energy between the modeled wild-type and mutant structures. The benchmark compares the predicted energy differences against experimentally measured  $\Delta\Delta G$  values in kcal/mol.

**Benchmark dataset** In previous studies, protein stability prediction methods have been benchmarked against multiple curated datasets: a set of 1030 mutants collected by Guerois et al.<sup>42</sup>; a set of 2156 mutants collected by Potapov et al.<sup>43</sup>; a set of 1210 mutants collected by Kellogg et al.<sup>44</sup>; and a set of 582 mutants collected by Benedix et al.<sup>45</sup>. The records in these datasets mainly originate from the ProTherm database<sup>34</sup> - a large, manually curated collection of thermostability data from the literature - and are mostly single point mutations. Our benchmark capture collects the Guerois, Potapov, and Kellogg datasets together and adds a fourth dataset of 2971 point mutants from ProTherm. This last dataset is lightly curated; it contains most of the single point mutations available in ProTherm excepting records where: (i) there is no corresponding structure determined via X-ray crystallography



with a resolution of at least  $2.5\text{\AA}$ ; (ii) there are multiple experimental  $\Delta\Delta G$  values for an individual mutation that differ by more than 2.5 kcal/mol in the experimental values; or (iii) the mutated protein is a transmembrane protein.

ProTherm contains details of the publications from which the thermodynamic data originated. These explicit references were omitted in the previously published datasets mentioned above but we have determined the source of the mutations for each record in the benchmark capture and reformatted the datasets into a standardized format. This refactoring has allowed us to determine the overlap between the datasets in terms of mutations and experimental assays to a large degree. These refactored datasets are included in the benchmark capture.

**Rosetta protocol** The benchmark capture currently includes scripts that can be used to run the best-performing protocol described by Kellogg et al. as protocol 16 (see row 16 in Table 1 in reference<sup>44</sup>). This protocol combines a soft-repulsive potential for conformational sampling of side-chains with a standard hard-repulsive potential for minimization to achieve higher prediction accuracy, following the observation that predictive methods are more accurate when the resolution of the force field is matched to the granularity of the sampling method. There are two steps in the protocol. First, the input structure is minimized. Next, fifty pairs of wild-type and mutant structural models are generated using the sampling strategy described above. The  $\Delta\Delta G$  value is calculated as the difference between the three best-scoring wild-type structural models and the three best-scoring mutant structural models as measured in Rosetta energy units (REU).

**Performance metrics** Three metrics are used for measuring the accuracy of the computational methods, each with a separate focus.

Pearson’s correlation coefficient measures the linear correlation between experimentally determined  $\Delta\Delta G$  values and their corresponding computationally predicted values. The coefficient is invariant to the scale of the predicted values.

The mean absolute error (MAE) is defined as the mean of the absolute differences between experimental and predicted  $\Delta\Delta G$  values. MAE is sensitive to the scale of the predicted values and is an important metric for protein design; high error reduces confidence in the predicted stability of individual cases.

Finally, the stability classification accuracy or fraction correct metric measures whether a mutation is correctly predicted to be (de)stabilizing or neutral, for a given definition of what constitutes a neutral mutation. Depending on this definition, it is possible to get a relatively high value for this metric with a set of random predicted values. Therefore this metric, while a useful metric for reporting whether a method can correctly classify the stability of a mutant, should be considered alongside the correlation and MAE.

**Key results** It has been previously reported that a recent Rosetta score function (Talaris) improves the performance of the Rosetta  $\Delta\Delta G$  protocol on the Kellogg dataset compared to the older score function, termed Score12<sup>46</sup>. We have tested the protocol on the three other curated datasets and found that Talaris improves the correlation with comparable MAE values for these datasets as well, compared to Score12. However, the performance measured by the same metric differs significantly between the different datasets, suggesting that the datasets represent different levels of prediction difficulty. These data are presented on the website.

**Notes** (i) We have made some modifications to the datasets from the original publications, such as updating deprecated PDB identifiers and correcting PDB IDs, PDB residue IDs, and  $\Delta\Delta G$  values based on cross-referencing to the respective publications. We now attribute each record of a dataset with publications from which the  $\Delta\Delta G$  values originate. This information was not present in some of the published datasets. (ii) Neutral experimental  $\Delta\Delta G$  values are defined as values within +/-1 kcal/mol, as used by Kellogg et al. We define neutral predicted  $\Delta\Delta G$  values as values within +/-1 score unit which differs from their definition (see<sup>44</sup> supporting information; neutral predicted is defined to be in the range [-3, 1.1]).

## Alanine scanning

**Purpose of this test** A frequent application of modeling methods is the prediction of energetically important interactions (“hotspots”) in protein-protein interfaces. By systematically mutating protein interface residues to alanine (“alanine scanning”) and measuring the effect on binding, Wells and coworkers<sup>47</sup> showed that not all residues with interface contacts, but only a smaller subset of ‘hotspot’ residues contribute significantly to the binding free energy of human growth hormone to its receptor. Subsequent studies suggested that such hotspots may be a general characteristic of many protein-protein interfaces<sup>48-50</sup>. This benchmark tests the ability of computational alanine scanning protocols to recapitulate the results of measurements of changes in binding affinity ( $\Delta\Delta G$  values) produced by experimental alanine scanning. A computational protocol performing well on this test set can then be used for additional applications, for instance, as a design tool to disrupt protein-protein interactions by mutations or through targeting small molecules to hotspots, or to analyze the effect of disease mutations.

**Benchmark dataset** The protocol has been benchmarked on a previously published set of the energetic effects of 233 mutations to alanine in 19 different protein-protein interfaces with known crystal structures<sup>51</sup>.

**Rosetta protocol** We have re-implemented a previously published alanine scanning protocol<sup>51,52</sup> in the current version of Rosetta to determine the current performance of this method. Unlike the generalized  $\Delta\Delta G$  protocol described above, which performs side chain optimization and side chain and backbone minimization over the entire protein structure, the alanine scanning protocol does not model perturbation of the backbone or side chains other than the side chain of the residue replaced with alanine. The  $\Delta\Delta G$  of binding upon mutation to alanine is calculated using the following equation, in which Rosetta total energy

is used to estimate the  $\Delta G$  of folding of each of the six terms:

$$\begin{aligned}\Delta\Delta G_{bind} &= \Delta G_{bind}^{MUT} - \Delta G_{bind}^{WT} \\ &= (\Delta G_{complex}^{MUT} - \Delta G_{partnerA}^{MUT} - \Delta G_{partnerB}^{MUT}) \\ &\quad - (\Delta G_{complex}^{WT} - \Delta G_{partnerA}^{WT} - \Delta G_{partnerB}^{WT})\end{aligned}\tag{2.1}$$

Alanine scanning uses a version of Rosetta’s Talaris energy function with modified weights intended for scoring mutations to alanine within interfaces, where the score term representing repulsive electrostatic interactions is down-weighted.

The previously published protocol<sup>51,52</sup> is available via the Robetta webserver at <http://robetta.bakerlab.org>, which has provided more than 20,000 predictions to date. The implementation described here will allow users to run predictions off-line and on large datasets, and implement and test modifications to the protocol.

**Performance metrics** Performance can be measured using the same metrics as in the generalized case of the  $\Delta\Delta G$  protocol described above, including the Pearson’s correlation of predicted  $\Delta\Delta G$  values to experimental  $\Delta\Delta G$  values, mean absolute error (MAE), and fraction correct (see previous section for descriptions of these metrics).

**Key results** Alanine scanning performance has not shown improvement when used with modern Rosetta score functions and aggressive side chain/backbone minimization methods; performance of the protocol described here is comparable to that shown in earlier publications<sup>51,52</sup> and available on the Robetta server.

**Notes/Limitations** (i) As the alanine scanning protocol does not perturb the protein backbone or side chains (other than the mutant residue), this protocol is not suitable for use on mutations outside of the interface. A mutation outside of the interface will not change the predicted interaction energy without the use of a more intensive sampling protocol. (ii)

As the backbone structure of the wild-type crystal structure is assumed to be a close approximation of the backbone structure of the mutant, this protocol is not useful in situations where this assumption does not hold. This includes testing of many simultaneous mutations that may result in larger structural rearrangements.

### 2.3.2 Design tests

Protein design methods are difficult to test rigorously because an ideal benchmark set would contain both successful and unsuccessful designs, however, the number of cases where both have been characterized functionally and structurally is small and not yet diverse enough. Until the amount of available data of this nature greatly increases, other datasets, in particular the diversity of sequences present in naturally evolved protein families or selected in large-scale experimental screens, can provide informative benchmarks that have been used in the past to assess and compare design methods<sup>53,54</sup>. In the following sections, we first focus on design tests using evolutionary information, and then describe a benchmark testing prediction of protein recognition specificity using data from comprehensive phage display experiments. In each case, we compare designed and evolutionary or experimentally selected sequences using metrics comparing not individual sequences (as the number of possible sequences is large and hence the chance of an exact match at all sequence positions extremely small), but instead predicted and observed amino acid distributions.

### 2.3.3 Using evolutionary information

**Purpose of this test** Evolutionary pressures on protein structure and function have shaped the amino acid sequences of today’s naturally occurring proteins<sup>55</sup>. Consequently, the sequences of natural proteins are nearly optimal for their structures<sup>56</sup>. Natural protein sequences therefore provide valuable information for evaluating the accuracy of computational protein design in predicting sequences consistent with a given protein structure and function. We expect that an ideal computational protein design method should be able to

recapitulate properties of naturally occurring proteins, including amino acid sequence preferences (“sequence profiles”) and patterns of amino acid covariation. In particular the latter tests whether computational protein design methods are capable of recapitulating the precise details of specific residue-residue interactions in proteins.

**Benchmark dataset** To evaluate to what extent protein design methods can recapitulate properties of naturally evolved proteins, we first characterized amino acid sequence profiles and amino acid covariation in 40 diverse protein domain families. Protein domains for this benchmark were selected from Pfam<sup>57</sup> based on the following criteria: (i) there is at least one crystal structure of the domain available from the PDB; (ii) there were at least 500 sequences of the domain family available from Pfam; and (iii) the domain had 150 or fewer amino acids. We selected 40 structurally diverse domains that satisfied these criteria. Sequence profiles were calculated by determining the amino acid distribution at each position and amino acid covariation was calculated for all pairs of amino acids using a mutual information based metric<sup>58</sup>.

**Rosetta protocol** We designed 500 sequences for each domain using a variety of protein design methods that used the same energy function but differed in how they modeled protein backbone flexibility. As a baseline, we performed fixed backbone protein design, which does not allow the backbone to be moved. Flexible backbone design simulations were performed multiple times using different temperatures and different types of backbone moves to assess how the magnitude and mechanism of backbone variation affects the recapitulation of natural sequence properties. The different types of backbone moves included Backrub, Kinematic Closure (KIC), small phi/psi moves and all atom minimization (Relax). We also tested fixed backbone design using a soft-repulsive energy function. Additional details on the different methods are described in<sup>53</sup> and Rosetta command lines are provided on the web resource, along with a performance comparison.

**Notes/Limitations** This benchmark makes the assumption that naturally occurring proteins are optimized for stability given their particular three-dimensional structures used as input. However, there certainly exist cases where proteins trade stability for function, such as hydrophobic patches that act as protein-protein binding interfaces or charged residues in the protein core used for catalyzing chemical reactions. We therefore expect to observe some differences between naturally occurring sequences and sequences predicted by an accurate protein design method (even if it were perfect). The benchmark assumes that methods that predict more “native-like” sequences overall are more accurate and thus more useful for experimental design applications<sup>56</sup>. In these applications, functional constraints, such as binding and catalysis, are usually explicitly represented by including functional binding partners or specifying certain key catalytic groups and their conformations.

### **Native sequence recovery**

**Performance metrics** Native sequence recovery<sup>56</sup> measures the ability of computational protein design to predict the amino acid sequence of a protein given its backbone conformation (??B). This is simply calculated as the percent identity between the native sequence and a designed sequence.

**Key results** We found that adding a small degree of backbone flexibility prior to design increased sequence recovery on average, however, further increasing the amount of backbone flexibility led to worse sequence recovery scores. A possible explanation for this decrease in recovery is that allowing more backbone flexibility resulted in sequences with a greater diversity in their amino acid sequences and consequently greater divergence from the native sequence. To confirm this, we calculated sequence entropy for the designed sequences and found that structural variation is positively correlated with sequence diversity. These results highlight a caveat with using native sequence recovery as a test of protein design accuracy, which is that protein sequences can be very different from each other but still be consistent

with the same protein fold<sup>59</sup>, and it is this sequence divergence that can be utilized to evolve existing proteins for new functions.

## Sequence profile recovery of protein families

**Performance metrics** Sequence profiles represent the distribution of amino acids at each position in a multiple sequence alignment of a protein family (??C). To compare natural and designed sequence profiles, we computed the divergence between the amino acid distributions at corresponding positions in the natural and designed sequences, as described in<sup>59</sup>. Briefly, profile similarity is the product of two scores: (i) the estimated probability that two amino acid distributions represent the same source distribution; and (ii) the a priori probability of the source distribution. It is defined as:

$$Profile\ Similarity(p, q) = \frac{1}{2}(1 - D^{JS}[p||q])(1 + D^{JS}[r||P_0]) \quad (2.2)$$

where  $p$  and  $q$  are amino acid probability distributions at corresponding positions in natural and designed sequences,  $r$  is the average of  $p$  and  $q$ ,  $P_0$  is the background distribution, and  $D^{JS}$  is the Jensen-Shannon divergence. Using this metric, positions in designed sequences receive high profile similarity scores if both: (i) their amino acid distribution is similar to the amino acid distribution at the corresponding position in the natural alignment; and (ii) their amino acid distribution is different than the background amino acid distribution.

**Key results** We observed that backbone flexibility improved our ability to recapitulate sequence profiles of naturally occurring protein families relative to fixed backbone design, and that there exists an optimal magnitude of backbone flexibility (using Rosetta  $kT = 0.9$  in “backrub” simulations,<sup>53</sup>) given that low or high temperature simulations performed worse than medium temperature simulations. This analysis also revealed an important pathology in the designed sequences, which showed an unrealistically high percentage of designed buried polar residues when compared to the natural sequences. To overcome this problem, we



repeated the benchmark using a newer Rosetta energy function, Talaris<sup>46</sup> that has stricter definitions for hydrogen-bonding geometries. We found that this decreased the percentage of buried polar residues, including serine, threonine and histidine, although it remained higher than in the natural sequences (?? and ??). These results can be quite sensitive to the reference energies in the applied energy function (which allow energetic evaluation of mutations). Existing automated tools<sup>60</sup> that reweight reference energy terms can be used to develop alternative energy functions. Future improvements to sampling and scoring will be required to further reduce the percentage of buried polar groups to levels found in naturally occurring proteins.

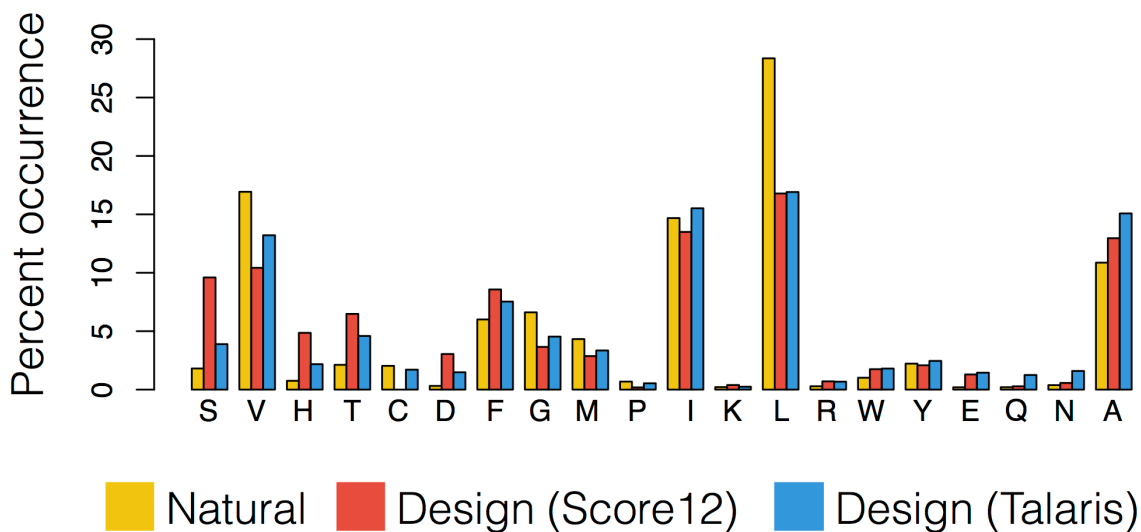


Figure 2.2: Comparison of occurrences of different amino acid residue types observed at buried positions between natural sequences and sequences designed with two different Rosetta energy functions. Barplot showing the percent occurrence of each type of amino acid found at buried positions in natural and designed sequences across 40 diverse protein families. Buried positions are defined as positions with greater than 14 neighboring positions, where neighboring positions have  $C\text{-}\beta$  atoms within  $8\text{\AA}$  of the  $C\text{-}\beta$  atom of the residue of interest. The X-axis is sorted by the magnitude of improvement of the Talaris energy function relative to the previous Score12 energy function with respect to the similarity to the natural percent occurrences.

## Amino acid covariation

**Performance metrics** To evaluate how well a given protein design method could recapitulate natural amino acid covariation, we designed 500 sequences for each protein domain in the benchmark and calculated the covariation between all pairs of positions in the designed sequences (??D). Covariation is calculated based on a mutual-information based metric described in<sup>58</sup>. The mutual information (MI) between each pair of columns in a multiple sequence alignment,  $i$  and  $j$ , was calculated as the difference between individual entropies ( $H_i$ ,  $H_j$ ) and the joint entropy ( $H_{i,j}$ ):

$$MI_{i,j} = H_i + H_j - H_{i,j} \quad (2.3)$$

The background mutual information due to random noise and shared ancestry is then subtracted to obtain the product corrected mutual information (MIp)<sup>61</sup>:

$$MIp_{i,j} = MI_{i,j} - \frac{\overline{MI}_i \times \overline{MI}_j}{\overline{MI}} \quad (2.4)$$

where  $\overline{MI}_j$  is the mean MI of position  $i$  with all other positions and  $\overline{MI}$  is the overall mean value. Next this value is converted to two Z-scores, one for each column, which are then multiplied together:

$$Z_{i \times y} = \frac{MIp_{i,j} - \overline{MIp}_i}{\sigma(MIp_i)} \times \frac{MIp_{i,j} - \overline{MIp}_j}{\sigma(MIp_j)} \quad (2.5)$$

The final covariation score, called Zpx, is calculated as the square root of the absolute value of  $Z_{i \times y}$ . (If  $Z_{i \times y}$  is negative, then Zpx is multiplied by -1.) This normalization was previously shown to reduce sensitivity to potential misaligned regions in multiple sequence alignments, which otherwise result in artificially high MI scores<sup>58</sup>. Similarity between natural and designed covariation was calculated as the percent overlap between the highly covarying pairs in the natural sequences and in the designed sequences. We considered pairs

with covariation scores greater than two standard deviations from the mean to be highly covarying<sup>53</sup>.

**Key results** We used this metric for quantifying the similarity of natural and designed covariation in order to compare different flexible backbone protein design methods that varied in either the magnitude or mechanism of backbone flexibility. As with sequence profile recovery, we observed that backbone flexibility improved our ability to recapitulate naturally occurring amino acid covariation relative to fixed backbone design, and that there exists an optimal magnitude of backbone flexibility (in the range of  $kT = 0.6$  to  $kT = 0.9$  in Rosetta simulations). We also found that flexible backbone design methods which incorporate backbone flexibility via iteratively applying local backbone moves (e.g. Backrub<sup>62</sup> or Kinematic Closure<sup>23</sup>) performed better than Rosetta methods that globally alter the backbone of the entire protein (e.g. Relax or AbInitioRelax)<sup>53</sup>.

### 2.3.4 Using large-scale experimental data

#### Recognition specificity

**Purpose of this test** “Sequence tolerance” refers to the concept that a certain profile of allowed residues can accommodate the evolved structure and function of a protein (??E). The computational sequence tolerance protocol attempts to predict the allowed sequence profile in protein-protein interfaces. The predictions are tested for their ability to recapitulate the sequence specificity preferences of protein recognition domains that have been determined by comprehensive phage display experiments. In contrast to the comparison to sequences of evolutionary families in the previous section, the experimentally determined profiles were selected primarily based on the same criterion (most stable binding) as in the design simulations.

**Benchmark dataset** The experimental data used for comparison in this benchmark set come from phage display specificity profiles for naturally occurring PDZ domains<sup>63</sup>, as well as phage display profiles for peptide interactions with synthetic variants of the Erbin PDZ domain<sup>63,64</sup>, comprising over 8000 peptide sequences tested against 169 natural and synthetic PDZ domains total.

**Rosetta protocol** There are two main computational steps: (i) the Rosetta Backrub application<sup>62</sup> uses Monte Carlo sampling starting from a single input structure to create an ensemble of near-native conformations; (ii) the sequence tolerance application<sup>65,66</sup> then uses a genetic algorithm to sample and score a large number of sequences for each member of the ensemble. An input file defines the sequence positions to be designed, and interactions within and between different parts of the structure can be individually reweighted, depending on the desired objective.

**Performance metrics** The analysis scripts use Boltzmann weighting to generate a predicted position weight matrix (PWM) for the specified sequence positions. This predicted PWM can be compared to known sequence profiles via these metrics described in the previous sequence tolerance publications<sup>65,66</sup>: (i) AAD, average absolute deviation, defined as:

$$\frac{1}{N} \sum_{i=1}^N |E_i - P_i| \quad (2.6)$$

and (ii) Frobenius distance, defined as:

$$\sqrt{\sum_{i=1}^N (E_i - P_i)^2} \quad (2.7)$$

where E is the vector of experimentally determined amino acid frequencies and P is the corresponding vector of predictions. (iii) AUC, or area under the receiver operator characteristic curve, measures the ability of the predictions to match the experimental values on a known scale, where 0.5 indicates random predictions and 1.0 is perfect. (iv) “Rank top”

measures the predicted rank of the most frequent experimentally determined amino acid.

**Key results** Recognition specificity performance with Talaris is comparable to the originally published performance of the protocol with Score12<sup>65,66</sup>.

**Notes/Limitations** (i) Although the sequence tolerance protocol is capable of generating backbone flexibility, which improves performance, it still relies on known input starting structures. Mutations can be made to these starting structures in order to predict the recognition specificity of experimentally characterized mutated proteins, but the additional mutation step might reduce the overall performance of the protocol. (ii) The backrub phase of the protocol must be run at a reasonable temperature (see protocol capture) to generate an appropriately matching amount of backbone flexibility in the sequence tolerance step. (iii) Due to limitations in the sequence space sampled by the genetic algorithm, it is not recommended to try and sample more than about 4-6 design positions simultaneously. (iv) Sequence profiles produced by this method may accurately predict the most frequently observed amino acid at a design position without containing enough total variation at that same position. (v) The performance metrics described above ignore potential co-variation in predicted or experimentally selected sequences.

### 2.3.5 Structure prediction tests

#### Loop reconstruction

**Purpose of this test** Being able to correctly model loop conformations (??F) is crucial because of their functional importance in many proteins, such as in forming the complementarity-determining regions in antibodies or in controlling substrate access and product release in enzyme active sites. However, since many loops in protein structures are flexible, loop modeling is computationally hard, because the many backbone degrees of freedom (depending on the length of the loop) result in a vast conformational search space. The purpose of this

test is to reconstruct known native loop conformations, as observed in crystal structures, in non-redundant benchmark sets of different loop lengths.

**Benchmark dataset** The Rosetta loop modeling benchmark<sup>23,63-67</sup> tests the ability of a protocol to reconstruct the backbone conformation of 12-residue loop segments in protein structures. The benchmark set consists of 45 non-redundant protein segments without regular secondary structure, curated from two previously described datasets<sup>68-73</sup>. In each case, the given segment is deleted from the protein structure and then reconstructed de novo, given a fixed backbone environment for the rest of the protein. All segment side chains and those within 10Å of the segment are modeled based on a side chain rotamer library<sup>74</sup> that does not include the native side chain conformations. The long loops benchmark<sup>67</sup> analogously tests whether protocols are able to reconstruct loop segments of 14-17 residues. This benchmark set consists of 27 non-redundant long loops, extracted and manually curated from the dataset described in<sup>75</sup>, by requiring at most five residues within 6Å of symmetry mates in the crystal lattice to minimize the potential impact of crystal contacts on loop conformations. De novo loop reconstruction and side chain optimization are performed as described above for the standard loop modeling benchmark.

**Rosetta protocol** Several protocols have previously been developed to reconstruct or predict the backbone conformation of loops in protein structures. The CCD protocol in Rosetta<sup>69</sup> uses insertion of fragments from proteins of known structure to sample the loop backbone degrees of freedom, followed by torsion angle adjustments via cyclic coordinate descent (CCD) to close the resulting chain break<sup>76</sup>. The kinematic closure (KIC) protocol<sup>23</sup> samples all but six loop backbone degrees of freedom probabilistically from Ramachandran space. These remaining three pairs of  $\phi/\psi$  torsion angles are then solved analytically through kinematic closure to close the chain break<sup>77</sup>. Next-generation KIC (NGK)<sup>67</sup> adds four additional sampling strategies to the standard KIC protocol: (i) the selection of pairs of  $\phi/\psi$  torsions from neighbor-dependent Ramachandran distributions; (ii) sampling of  $\omega$  degrees

of freedom; as well as annealing methods that gradually ramp the weights of (iii) the repulsive terms; and (iv) the Ramachandran terms of the Rosetta energy function to overcome energy barriers. All three loop modeling protocols use Monte-Carlo simulated annealing for rotamer-based side-chain optimization (“repacking”) of the loop residues and those within 10 Å of the loop, followed by gradient-based minimization.

**Performance metrics** With each loop modeling protocol, hundreds of models are generated per benchmark case. Each model is then superposed onto the native structure (excluding the reconstructed loop), followed by calculating the loop backbone heavy-atom root mean square deviation (RMSD) of the model to the native loop conformation. The overall benchmark performance of each protocol is then evaluated using two different metrics across the entire benchmark set: (i) the median loop backbone RMSD of the lowest-energy model to the native structure (or median lowest loop backbone RMSD of the 5 lowest-energy models, which is less susceptible to stochastic fluctuations<sup>60</sup>); and (ii) the median percentage of models generated that have a loop backbone RMSD below 1Å (sub-angstrom predictions).

**Key results** With the Rosetta Score12 energy function (the standard before the switch to the Talaris2013 energy function in revision 55274), only the KIC<sup>23</sup> and NGK<sup>67</sup> protocols successfully sampled sub-angstrom loop conformations in many cases, achieving a median RMSD across the entire 12-residue loop benchmark set of 1Å. NGK significantly outperformed standard KIC in the sampling of sub-Å loop conformations, with NGK reaching a median of 16.3% sub-Å models compared to 4.3% for standard KIC<sup>67</sup>. Since the advent of the Talaris2013 score function<sup>46-60</sup>, the CCD protocol now also achieves a median RMSD < 1Å on the 12-residue loop dataset. The median percentage of models with sub-Å RMSD is still significantly higher for NGK (13.4%) than for standard KIC (6.4%) and CCD (1.8%). For the more difficult sampling problem in the long loops benchmark, the sub-Å sampling performance of NGK improved from Score12 (0.53%) to Talaris2013 (1.0%).

**Notes/Limitations** (i) Flexible loops are often better described by a conformational ensemble rather than a single conformation, and some simulations indeed reveal several clusters of different low-energy conformations. (ii) Crystal contacts can influence loop conformations, and the absence of those contacts during modeling can result in predictions differing from the crystallographic conformation. (iii) There are similar considerations for the presence of water molecules, ions or other small molecules, which might influence loop conformations. (iv) For the KIC and NGK protocols, the start and end points of loops are assumed fixed during the simulations; this simplifies the sampling problem in the context of “native” loop endpoints (i.e. taken from a crystal structure), but complicates the situation when the conformation of loop endpoints may not be known exactly, e.g. when building loops in homology models. In these cases, protocols that sample the positions of the endpoints or apply KIC moves over several overlapping regions may be more suitable. (v) Modeling long loops is difficult for current protocols, due to the large conformational search space, which is apparent from the considerably lower fraction of sub-angstrom models in the 14-17 residues loop benchmark. (vi) The KIC and NGK protocols do not preserve protein secondary structure, due to probabilistic sampling of  $\phi/\psi$  torsions from Ramachandran space. Additional sampling constraints could be included to preserve secondary structure.

### 2.3.6 Website description

The benchmark captures are collected and presented online at <https://kortemmelab.ucsf.edu/benchmarks>. The purposes of the website are to: (i) describe specific and well-defined problems in computational modeling; (ii) describe and provide benchmarks which can be used to measure the success of methods designed to address these problems; (iii) publish the performance of methods using parameters provided by experienced users; and (iv) act as a unified portal for downloading the benchmark captures.

On the main benchmark page (??A), we describe each benchmark - its purpose, application, and the currently considered datasets - and publish results of benchmark runs so that



users can quickly gauge the performance of different methods. Relevant command lines are provided to promote best practice for each method when using the Rosetta software suite.

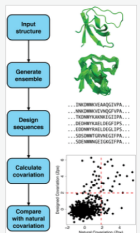
### Sequence profile recovery and amino acid covariation ↗

Amino acid covariation, where the identities of amino acids at different sequence positions are correlated, is a hallmark of naturally occurring proteins. This benchmark quantifies the extent to which computational protein design methods can recapitulate naturally occurring amino acid covariation.

To compare amino acid covariation in natural and predicted designed protein sequences, we selected a dataset of 40 protein domains that were diverse with respect to their secondary structure composition and fold class. We then quantified natural amino acid covariation for each domain by creating a multiple sequence alignment for the domain, followed by computing covariation between every pair of columns in the multiple sequence alignment by using a mutual information based method. Pairs of amino acid positions with a covariation score that is two standard deviations above the mean or greater were considered to be highly covarying pairs.

We predicted designed protein sequences for each of the 40 domains using RosettaDesign (see publication below). We first used the standard RosettaDesign fixed backbone protocol, which takes a crystal structure as input and runs Monte Carlo simulated annealing, to predict 500 designed sequences for each domain structure. We then quantified amino acid covariation in the designed sequences and compared it to natural amino acid covariation for each domain.

To investigate the effect of the magnitude of backbone flexibility in the design protocol, we generated conformational ensembles with a variety of protocols, including Backrub, Kinematic Closure ("KIC"), Small Phi-Psi moves ("Sma"), and Relax. We designed sequences for each ensemble and quantified similarity to natural covariation for each set of sequences and observed a significant increase in covariation similarity for flexible backbone simulations relative to the fixed backbone simulation. The benchmark also examines several other sequence characteristics, including sequence recovery, sequence profile similarity, sequence entropy.



**Metrics**

Revision	Design Method	Dataset	Dataset size	Covariation Similarity	Sequence Profile Similarity	Native Sequence Recovery	Sequence Entropy
N/A	Native sequences	Olikainen	40 domains	100%	N/A	N/A	0.68
Rev. 39284	Fixed backbone (fixbb)	Olikainen	40 domains	29.4%	0.32	33.1	0.23
Rev. 39284	Backrub, kT = 0.9	Olikainen	40 domains	37.5% <sup>1</sup>	0.44	26.6	0.67
Rev. 39284	Small phi-psi moves, kT = 1.2	Olikainen	40 domains	33.9%	0.38	25.1	0.56
Rev. 39284	Soft	Olikainen	40 domains	23.0%	0.30	33.4	0.13
Rev. 40935	KIC, kT = 1.2	Olikainen	40 domains	36.9%	0.43	33.0	0.60
Rev. 40935	Relax	Olikainen	40 domains	31.2%	0.37	45.0	0.28

<sup>1</sup> Values in bold text denote the best result for the corresponding metric (column).

**Relevant command line flags**

Revision	Design Method	Binary name	Additional flags
Rev. 39284	Fixed backbone (fixbb)	fixbb	--write ALLAA_res --ex1 --ex2 --extract_cutoff 0 --no_his_his_pairE --minimize_sidechains
Rev. 39284	Backrub, kT = 0.9	backrub	--write NATAA_res --ex1 --ex2 --extract_cutoff 0 --backrubmc_kt 0.9 --backrub_inital_pack
Rev. 39284	Small phi-psi moves, kT = 1.2	backrub	--write NATAA_res --ex1 --ex2 --extract_cutoff 0 --backrubmc_kt 1.2 --backrub_inital_pack --backrub_sm_prob 1.0
Rev. 39284	Soft	fixbb	--write ALLAA_res --ex1 --ex2 --extract_cutoff 0 --no_his_his_pairE --minimize_sidechains --score_weights soft_rep_design.wts
Rev. 40935	KIC, kT = 1.2	loopmodel	--loops refine_refine_kic --loops kic_max_seglen 12 --loops outer_cycles 1 --loops refine_init_temp 12 --loops refine_final_temp 12 --loops optimize_only_kic_region_sidechains_after_move --ex1 --ex2 --extract_cutoff 0 --loops kic_recover_last --loops max_inner_cycles 250 --loops repack_period 250 --loops outer_cycles 20
Rev. 40935	Relax	relax	--ex1 --ex2 --extract_cutoff 0

**Protocol documentation**

Fixed backbone design (fixbb)  
Backrub  
Kinematic closure (KIC)

**Publications**

Olikainen, N., Kortemme, T., Computational Protein Design Quantifies Structural Constraints on Amino Acid Covariation, 2013. PLoS Comput Biol 9(11):e1003313. doi: 10.1371/journal.pcbi.1003313

### Sequence profile recovery and amino acid covariation Download

Last updated: 2015-02-11

Evolutionary pressures on protein structure and function have shaped the amino acid sequences of today's naturally occurring proteins. Consequently, the sequences of natural proteins are nearly optimal for their structures. Natural protein sequences therefore provide valuable information for evaluating the accuracy of computational protein design. The purpose of this benchmark is to evaluate the extent to which protein design can recapitulate properties of naturally occurring proteins, including amino acid sequence preferences ("sequence profiles") and patterns of amino acid covariation.

This benchmark includes:

- a set of 40 diverse protein domains with representative crystal structures and sequence alignments
- command line arguments for running fixed backbone and flexible backbone design methods in Rosetta
- analysis scripts that compare sequence profiles and patterns of amino acid covariation between natural and designed sequences

This protocol capture is based on a benchmark developed by Olikainen & Kortemme and referenced below.

**Licensing**

The contents of the repository where possible are licensed under the MIT License. The license only applies to files which either: i) include the license statement; or ii) which are explicitly listed in some file in the repository as being covered by the license. All other files may be covered under a separate license. The LICENSE file in the root of this repository is present only for the convenience of the user to indicate the license which covers any novel content presented herein.

**Downloading the benchmark**

The benchmark is hosted on GitHub. The most recent version can be checked out using the `git` command-line tool:

```
git clone https://github.com/Kortemme-Lab/covariation.git
```

**Directories in this archive**

This archive contains the following directories:

- `input`: contains the input files for the benchmark. Input files specific to a particular protocol are in a subdirectory named after the protocol. The input files are described in more detail in `input/README.rst`.
- `output`: these directories are empty by default. This is the default output location for protocols if they are run on the local machine.
- `output/sample`: contains sample output data that can be used to test the analysis script.
- `analysis`: contains the analysis scripts used to analyze the output of a prediction run. All protocols are expected to produce output that will work with the analysis scripts.
- `protocols`: contains the scripts needed to run a job. The scripts for a protocol are provided in a specific subdirectory.
- `app`: contains scripts that can be used to run the entire benchmark using specific cluster architectures. For practical reasons, a limited number of cluster systems are supported. Please feel free to provide scripts which run the benchmark for your particular cluster system.

**Protocols**

This repository contains one protocol which can be used to run the benchmark. We welcome the inclusion of more protocols. Please contact [support@kortemmelab.ucsf.edu](mailto:support@kortemmelab.ucsf.edu) if you wish to contribute towards the repository.

Each protocol is accompanied by specific documentation in its protocol directory.

**Protocol 1: Fixed backbone design**

Software suite: Rosetta  
Protocol directory: protocols/fixbb

**Protocol 2: Flexible backbone design using backrub ensembles**

Software suite: Rosetta  
Protocol directory: protocols/backrub

**References**

Computational protein design quantifies structural constraints on amino acid covariation. 2013. Olikainen N, Kortemme T. PLoS Comput Biol 9(11):e1003313. doi: 10.1371/journal.pcbi.1003313. Epub 2013 Nov 14.

**Analysis**

The same set of analysis scripts is used by all protocols. Conceptually, the analysis scripts should be a black box that is separated from the output of each protocol by an interface.

The analysis scripts calculate sequence profile similarity and covariation similarity metrics which can be used to evaluate the results of the design simulations. The scripts are described in more detail in `analysis/README.rst`.

Protocol capture

Figure 2.3: Benchmark and protocols capture website. Left: The website presents an overview of each benchmark and publishes the performance of different methods using a set of standardized metrics. Parameters important to the protocol performance are also provided. Right: Each benchmark capture is stored in a documented version-controlled archive. The most recent version can be downloaded directly from the website.

Each capture has been compiled as a version-controlled, publicly-accessible, open-source archive (currently hosted on GitHub), containing both the associated benchmark datasets and scripts to analyze benchmark output in a specific format. Execution scripts to run the benchmark using at least one computational method are provided. Both the analysis and

execution scripts are documented in detail within the capture, and this documentation can be viewed online on GitHub. For convenience, these captures are available for download directly from the web resource (??B).

As our intention is to provide a dynamic resource, it may be appropriate to refine or expand certain datasets as new data become available in the future. Version control allows us to update the contents while allowing users to track changes in the datasets or analysis metrics. Major changes to repositories will be tagged and referred to in the website text. Following the philosophy of the computer language benchmark projects<sup>78,79</sup>, the parameters used for each method should reflect the best practice. For this reason, they should be ideally contributed by a developer or experienced user, and we encourage users to submit their methods, parameters, and results for inclusion on the website.

We have aimed to provide rich, user-friendly datasets. For example, the protein stability datasets are provided in both JSON and CSV formats. The former is readily integrated with multiple programming languages and web frameworks whereas the latter is human-readable and easily imported into spreadsheet applications. In both of these datasets, each record is now associated with experimental values taken from the literature - which we were able to determine using the rich source of data provided by the ProTherm database<sup>34</sup> - so that outliers in the predicted set can be investigated using the original experimental data.

## 2.4 Discussion

We have presented our implementation of a benchmarking and protocol capture web resource which currently describes five diverse benchmarks and their expected performance when tested using a known best-practice methods from the Rosetta software suite. The web site functions as an openly accessible, online, and version-controlled collection of a variety of benchmarks and macromolecular modeling and design protocols, providing a summary of the evolution of the protocols and indicating their expected performance on the associated

benchmarks.

The web resource was motivated by previous work<sup>78,79</sup> which has fostered, and continues to foster, competition and innovation in computer language development through the open communication of standardized benchmarks which allow for direct and fair comparison between competing computer languages. In those projects, knowledge of the performance of each language for the particular problem and the open communication of the most efficient code is important for both software developers when choosing which language to use for a particular project and for the language developers so that they can identify parts of the language for optimization. By providing both curated diverse datasets for benchmarking and analysis scripts to generate a set of appropriate metrics, we hope that we can help developers to evaluate new methods in informative ways, which is critically needed for continued progress in many areas of structure-based modeling and design.

## 2.5 Acknowledgements

We would like to thank the general Rosetta Commons community for continuing, helpful development, support, and discussion. We thank Liz Kellogg, Rebecca Alford, and Julia Koehler Leman for support with developing the  $\Delta\Delta G$  benchmark, and Amelie Stein for contributions to loop modeling.

## 2.6 Supporting Information

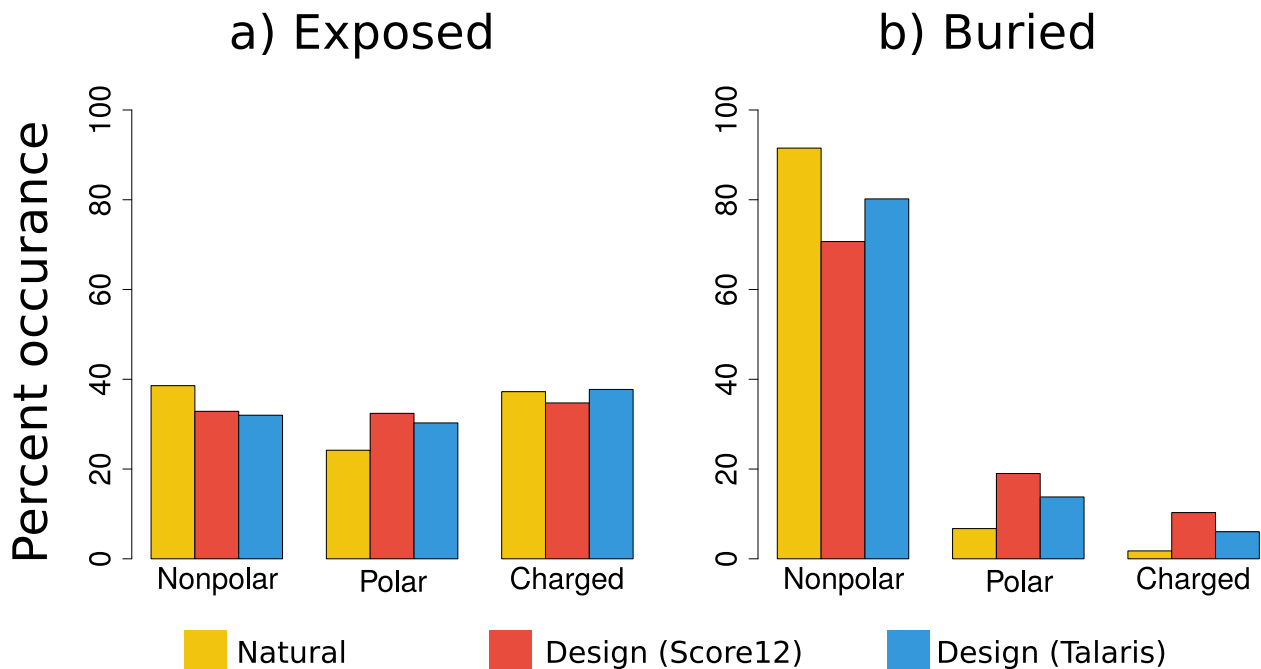


Figure 2.4: **Comparison of occurrences of different amino acid residues (by polarity) observed at buried and exposed positions.** Each barplot shows the percent occurrence of each polarity category of amino acid found in natural and designed sequences across 40 diverse protein families. Yellow bars show the percent occurrence in natural sequences, red the percent occurrence in sequences designed using Rosetta’s Score12 energy function, and blue the percent occurrence when designing with Rosetta’s Talaris energy function. Neighboring positions are defined as any position with a C- $\beta$  atom within 8Å of the position being investigated. For the purposes of this figure, nonpolar amino acids are defined as: CGAVLIMFWP, polar: STYNQ, charged: HRKDE. (A) Exposed positions are defined as positions with between 0 and 8 neighboring positions. (B) Buried positions are defined as positions with greater than 14 neighboring positions. Using the Talaris energy function reduces the percentage of charged residues placed in buried positions by 43% (from 10.5% to 6.2%), bringing the predictions closer to the native sequence properties.

# Chapter 3

## Flex ddG: Ensemble-based prediction of interface binding free energy upon mutation

### 3.1 Introduction

Protein-protein interactions underlie many biological processes, including signal transduction and antibody-antigen recognition. In fact, mutations at protein-protein interfaces are over-represented within disease-causing mutations<sup>80</sup>, indicating the central importance of these interactions to biology and implications to human health. A computational method capable of predicting mutations that strengthen or weaken known protein-protein interactions would not only serve as a useful experimental tool to improve our understanding of biology, but would also enhance our ability to create protein drugs with new modes of actions, and additionally enhance engineering applications such as design of protein-based sensors and materials.

Several prior methods have attempted to predict changes in binding free energies using different approaches to scoring and sampling, including weighted energy functions that

seek to describe physical interactions underlying protein-protein interactions<sup>42,81</sup>, statistical and contact potentials<sup>82-84</sup>, a combination of these approaches<sup>85</sup>, graph-based representations<sup>86</sup>, and methods that attempt to sample backbone structure space locally around mutations<sup>87</sup>.

We set out to develop and assess methods for prediction of change in binding free energy after mutation (interface  $\Delta\Delta G$ ) within the Rosetta macromolecular modeling suite. Rosetta is freely available for academic usage, allowing future combination of these predictions with Rosetta’s powerful protein design capabilities, which has proven successful in a variety of applications<sup>88</sup>. Prior projects have applied Rosetta predictions to dissect determinants of binding specificity and promiscuity<sup>89,90</sup> enhance protein-protein binding affinities<sup>91,92</sup>, and to design modified<sup>52,93</sup> and new interactions<sup>94,95</sup>, but no prior benchmarking effort has studied the performance of predicting changes in binding free energy in Rosetta on a large, diverse benchmark data set, in part because such a dataset has only become available more recently. The current state-of-the-art Rosetta  $\Delta\Delta G$  method, `ddg_monomer`<sup>44</sup>, has proven effective at predicting changes in stability in monomeric proteins after mutation, but had not yet been tested at predicting change of binding free energies in protein-protein complexes. Prior “computational alanine scanning”  $\Delta\Delta G$  methods were benchmarked on mutations in protein-protein interfaces, focusing on mutations to alanine<sup>1,51,52</sup>. The original alanine scanning method sampled only side chain degrees of freedom, which is a fair approximation for mutations to alanine (which are not expected to cause large backbone perturbations<sup>96</sup>), but a less probable assumption for mutations to larger side chains which might require some degree of backbone rearrangement to accommodate the change. Adaptation of the alanine scanning method to recent score function and sampling method developments in Rosetta has not shown improvement in benchmarking<sup>1</sup>, indicating a need to more thoroughly develop and test a method that attempts to more aggressively sample conformational space.

We sought to create a method that would take into account the natural conformational flexibility of proteins by representing structures as an ensemble of individual full-atom mod-

els, generating sufficient microstate-like models to effectively explore the biologically relevant and accessible portions of conformational space close to the native structure. Ensemble representations have proven their effectiveness to predict change in protein stabilities after mutation<sup>45</sup> and to improve  $\Delta G_{binding}$  calculations between kinases and their inhibitors<sup>97</sup>.

We chose to sample conformational diversity through use of the “backrub” protocol implemented in Rosetta. The backrub method samples using local, coupled, side chain and backbone motions, similar to those observed in high-resolution crystal structures<sup>22</sup>. Backrub ensembles appear to recapitulate properties of proteins that have been experimentally determined, such as side chain NMR order parameters<sup>24</sup>, sequence profiles at protein-protein interfaces<sup>98</sup>, sequence profiles of protein-peptide binding specificity<sup>65,66</sup>, and can sample the conformational variability between protein homologs<sup>99</sup>. Backrub has also proved effective in design applications, such as for the redesign of protein-protein interfaces<sup>93</sup> and recapitulation of mutations that alter ligand-binding specificity<sup>100</sup>. When Davey and Chica compared backrub ensembles to ensembles generated via molecular dynamics simulations or PertMin<sup>101</sup>, backrub ensembles were shown in certain cases to be the only generated ensembles with a higher diversity (as measured with RMSD) from each other than from the original input crystal structure, indicating that backrub is uniquely suited to produce diverse ensembles that stay in the local conformational space of the input structure.<sup>101</sup> We hypothesized that this property of backrub ensembles would translate into these ensembles serving as an effective structural representation to use when predicting interface  $\Delta\Delta G$  values.

## 3.2 Methods

Developing and assessing the accuracy of a new method to predict changes in binding free energy after mutation requires a large and diverse benchmark set covering single mutations to all amino acid types, multiple mutations, and mutations across a variety of protein-protein interfaces. To facilitate comparisons to other methods and to avoid biases specific to our

n	Name
1240	Complete dataset
748	Single mutation to alanine
273	Multiple mutations
130	Small-To-Large Mutation(s)

Table 3.1: ZEMu dataset subset definition and composition

approach, we chose to use an existing benchmark dataset created by Dourado and Flores<sup>87</sup> during the creation of their ZEMu (Zone Equilibration of Mutants) method. The ZEMu dataset was curated from the larger SKEMPI database<sup>36</sup> and was filtered to avoid a bias towards complexes in which a single position is repeatedly mutated, experimental data that is not peer-reviewed, redundancy, mutations outside of interfaces, mutations involved in crystal contacts, and experimental  $\Delta\Delta G$  values for which wild-type and mutant conditions (such as pH) varied. Confidence in the “known” experimental  $\Delta\Delta G$  values is important, as it has been shown that the experimental methodology used can have a strong effect on the performance of predictors of changes in binding free energy<sup>102</sup>. The ZEMu dataset was also curated to include a wide range of both stabilizing and destabilizing mutants, small side chain to large side chain mutations, single and multiple mutations, and a diversity of complexes (??).

After a review of the literature from which the known experimental  $\Delta\Delta G$  values originated, we removed one data point from the 1254 point ZEMu set that we could not match to the originally published reported affinity value. We also removed 5 mutations we determined to be duplicates, along with 8 mutations that were reverse mutations of other data points, leaving us with a test set of 1240 mutations. We defined which complexes that contained at least one antibody binding partner by comparison of PDB identifiers with SAbDab<sup>103</sup>.

Our protocol, called “flex ddG”, is implemented within the RosettaScripts scripting interface to the Rosetta macromolecular modeling software suite<sup>104</sup>, which makes the protocol easily adaptable to future improvements and energy function development. We utilized Rosetta’s Talaris<sup>60,74,105</sup> energy function. Version numbers of tested software are available



in ??.

Flex ddG method steps are outlined in ?? (full Rosetta Scripts XML available in ??). Step 1: The protocol begins with an initial minimization (on backbone  $\phi/\psi$  and side chain  $\chi$  torsional degrees of freedom, using the “lbfgs\_armijo\_nonmonotone” minimization algorithm option) of the input model using the Rosetta energy function. This (and later) minimizations are performed with constraints that harmonically restrain pairwise atom distance to their values in the input crystal structure. Minimization is run until convergence into a local energy well. Step 2: Starting from the minimized input structure, the backrub method in Rosetta is used to create an ensemble of models. In brief, each backrub move is undertaken on a three-residue stretch of protein, chosen randomly from the set of residues within 8 Å of any mutant position. All atoms in the three-residue stretch of structure are rotated locally about an axis defined as the vector between the endpoint c- $\alpha$  atoms. Bond angles involving the endpoint atoms are then minimized. Backrub is run at a temperature of 1.2, for up to 60,000 backrub Monte Carlo trials. 50 output structures are generated. Step 3A: For each of the 50 structure models in the ensemble (output by backrub), the Rosetta “packer” is used to optimize side chain conformations for the wild-type sequence using discrete rotameric conformations<sup>74</sup>. The packer is run with no extra options to the multi-cool annealer<sup>106</sup>. Step 3B: Independently and in parallel to step 3A, side chain conformations for the mutant sequence are optimized on all 50 models. Step 4A: Each of the 50 wild-type structures is minimized, again adding pairwise atom-atom constraints to the input structure. Minimization is run with the same parameters as in step 1; the coordinate constraints used in this step are taken from the coordinates of the Step 3A structure. Step 4B: As step 3B but for each of the 50 mutant models. Step 5A: Each of the 50 minimized wild-type structures are scored in complex, and the individual complex components are scored individually. The scores of the split, unbound complex components are obtained simply by splitting the complex halves away from each other. No further minimization or side chain optimization is performed on the unbound states before scoring. Step 5B: In the same fashion as Step 5A, each of the 50 minimized

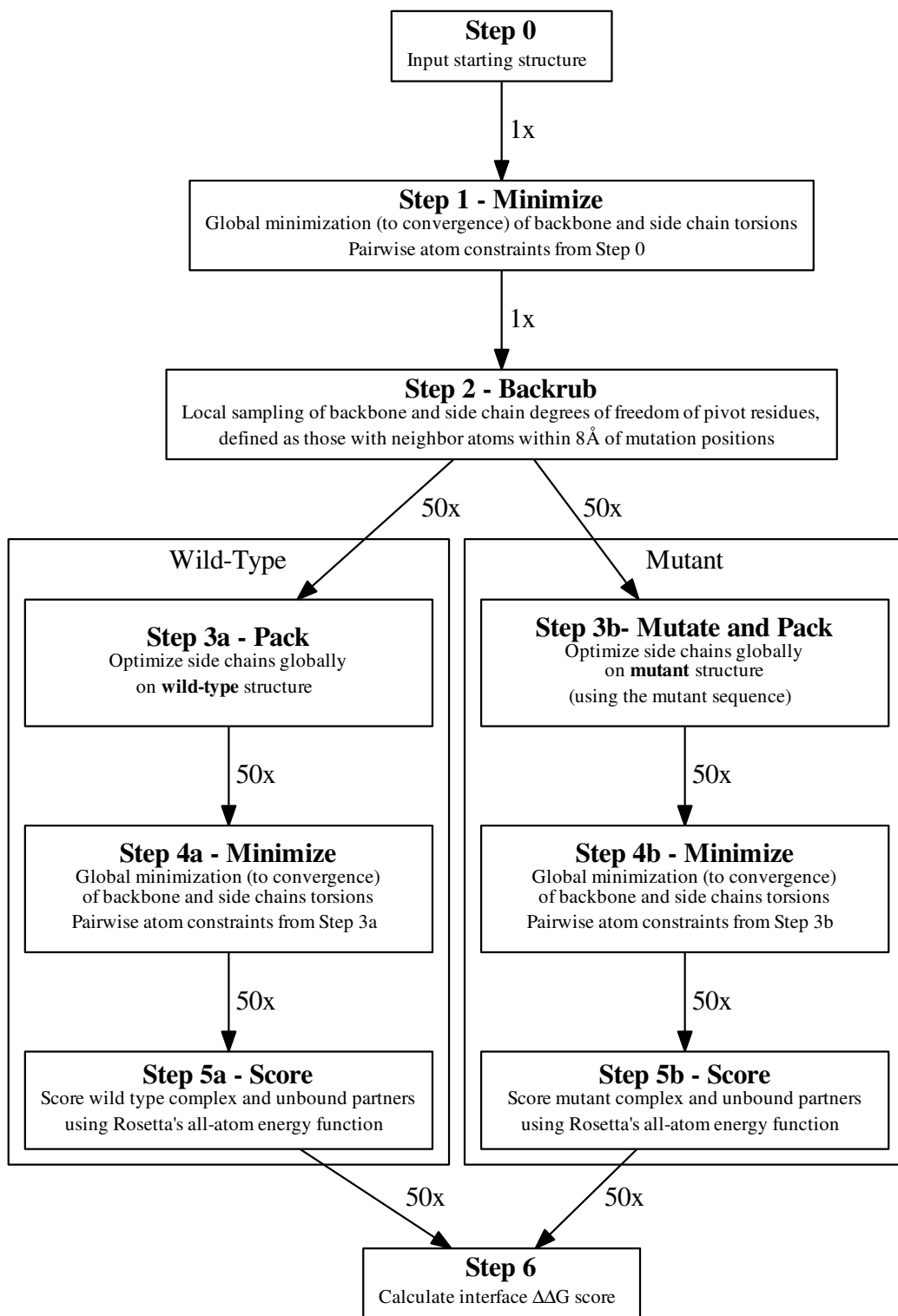


Figure 3.1: Schematic of the flex ddG protocol method.

mutant structures are scored in complex, and the individual complex components are scored individually. Step 6: The interface ddG score is produced via Eq. ??:

$$\begin{aligned} \Delta\Delta G_{bind} &= \Delta G_{bind}^{MUT} - \Delta G_{bind}^{WT} \\ &= (\Delta G_{complex}^{MUT} - \Delta G_{partnerA}^{MUT} - \Delta G_{partnerB}^{MUT}) \\ &\quad - (\Delta G_{complex}^{WT} - \Delta G_{partnerA}^{WT} - \Delta G_{partnerB}^{WT}) \end{aligned} \tag{3.1}$$

We evaluate performance of the protocol by comparing predicted ddG scores to known experimental values, using Pearson’s correlation R, Mean Absolute Error (MAE), and Fraction Correct (FC). Fraction Correct is defined as the number of mutations categorized as stabilizing, neutral, or destabilizing correctly, divided by the total number of mutations in the benchmark set. Stabilizing mutations are defined as those with a  $\Delta\Delta G \leq -1.0$  kcal/mol, neutral as those with  $-1.0$  kcal/mol  $< \Delta\Delta G < 1.0$  kcal/mol, and destabilizing as those with  $\Delta\Delta G \geq 1.0$  kcal/mol.

MAE (Mean Absolute Error) is defined in Eq. ?? as:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i| = \frac{1}{n} \sum_{i=1}^n |e_i| \tag{3.2}$$

where  $y_i$  are the predicted  $\Delta\Delta G$  values,  $x_i$  are the known, experimentally determined values, and  $e_i$  is the prediction error.

### 3.3 Results and discussion

The overall performance of the protocol is summarized in ?. We compare 4 prediction methods: (a) our flex ddG backrub ensemble method, (b) the prior state-of-the-art Rosetta methodology, `ddg_monomer`<sup>44</sup>, (c) a control version of our flex ddG protocol which omits the backrub ensemble generation step, leaving only the minimization and packing steps, and (d) the ZEMu (zone equilibration of mutants) method<sup>87</sup>.

Mutation Category	Prediction Method	N	R	MAE	FC
Complete dataset	flex ddG	1240	<b>0.63</b>	<b>0.93</b>	<b>0.76</b>
	ddG monomer (hard-rep)		0.51	1.04	0.70
	no backrub control		0.57	1.00	0.74
	ZEMu paper		0.61	0.96	0.75
Small-To-Large Mutation(s)	flex ddG	130	<b>0.64</b>	<b>0.87</b>	<b>0.75</b>
	ddG monomer (hard-rep)		0.31	1.10	0.65
	no backrub control		0.42	1.01	0.70
	ZEMu paper		0.48	1.03	0.64
Single mutation to alanine	flex ddG	748	<b>0.50</b>	<b>0.72</b>	<b>0.75</b>
	ddG monomer (hard-rep)		0.36	0.81	0.70
	no backrub control		0.44	0.78	<b>0.75</b>
	ZEMu paper		0.45	0.76	<b>0.75</b>
Multiple mutations	flex ddG	273	0.62	1.51	0.77
	ddG monomer (hard-rep)		0.50	1.69	0.66
	no backrub control		0.58	1.60	0.71
	ZEMu paper		<b>0.64</b>	<b>1.46</b>	<b>0.78</b>

Table 3.2: Main results table. R = Pearson’s R. MAE = Mean Absolute Error. FC = Fraction Correct. flex ddG steps = 35000.

Our flex ddG method outperforms the comparison methods on the complete dataset in each of the correlation, MAE, and fraction correct metrics. On the small-to-large subset of mutations where we expect to see the largest performance gains from using a backbone ensemble method, we see a substantial improvement in performance as compared to the alternative methods. Performance of the flex ddG on the subset of single mutations to alanine is also competitive or outperforms the alternative methods. As we do not expect single mutations to alanine to require intensive backbone sampling, our method’s effectiveness on this subset shows that it is fairly robust to the mutation type. This observation could be explained by the fact that we undertake backrub sampling prior to making the mutation to sample underlying, relevant flexibility of the input crystal structure instead of distorting the local structure around a mutation to resolve a clash or poor interaction with a mutant side chain. Finally, our method shows improved performance compared to the control method and ddg\_monomer on the subset of multiple mutations, but for this set does not match the performance of the ZEMu method. This could indicate that further refinement to the backrub

sampling parameters are required in the case of multiple mutations, since as there are more mutation sites, there will be more surrounding backrub pivot residue sites. However, flex ddG outperforms ZEMu on multiple mutations if none of the mutations are to alanine (??).

The underlying scatterplots for the flex ddG and control methods on the complete dataset and small-to-large subsets are shown in ??.

### 3.3.1 Effect of averaging more structures

In order to measure the degree to which averaging  $\Delta\Delta G$  values across an ensemble of models improves performance, we evaluated the performance of flex ddG as we average across an increasing number of structures (from 1 to 50). ?? shows the effect on performance as predictions from an increasing number of wild type and mutant structures are averaged. The structures used are first sorted by the score of the corresponding repacked and minimized wild type model, such that producing a  $\Delta\Delta G$  with 1 model will only use the lowest (best) scoring model, 2 models will use the 2 lowest scoring models, and so forth. ??(a) shows performance on the complete dataset. As more structures, of increasingly high wild type complex score, are averaged, correlation with known experimental values increases. Conversely, performance for the no backrub control method (shown in light blue) decreases as more structures are averaged. This result indicates that sampling with backrub adds information that improves  $\Delta\Delta G$  calculation, despite the additional structures being averaged having higher scores ???. These higher-scoring models which would traditionally be thought to be less likely to represent the folded global free-energy minimum, and therefore less predictive for  $\Delta\Delta G$  calculation.

Performance of the ddg\_monomer method is also improved as more output structures are averaged (??). This is somewhat unexpected, as our no-backrub control method is conceptually similar to the ddg\_monomer method. The difference may arise from the fact that ddg\_monomer ramps the repulsive term of the energy function during minimization, which is likely to improve results by sampling conformational space more broadly than

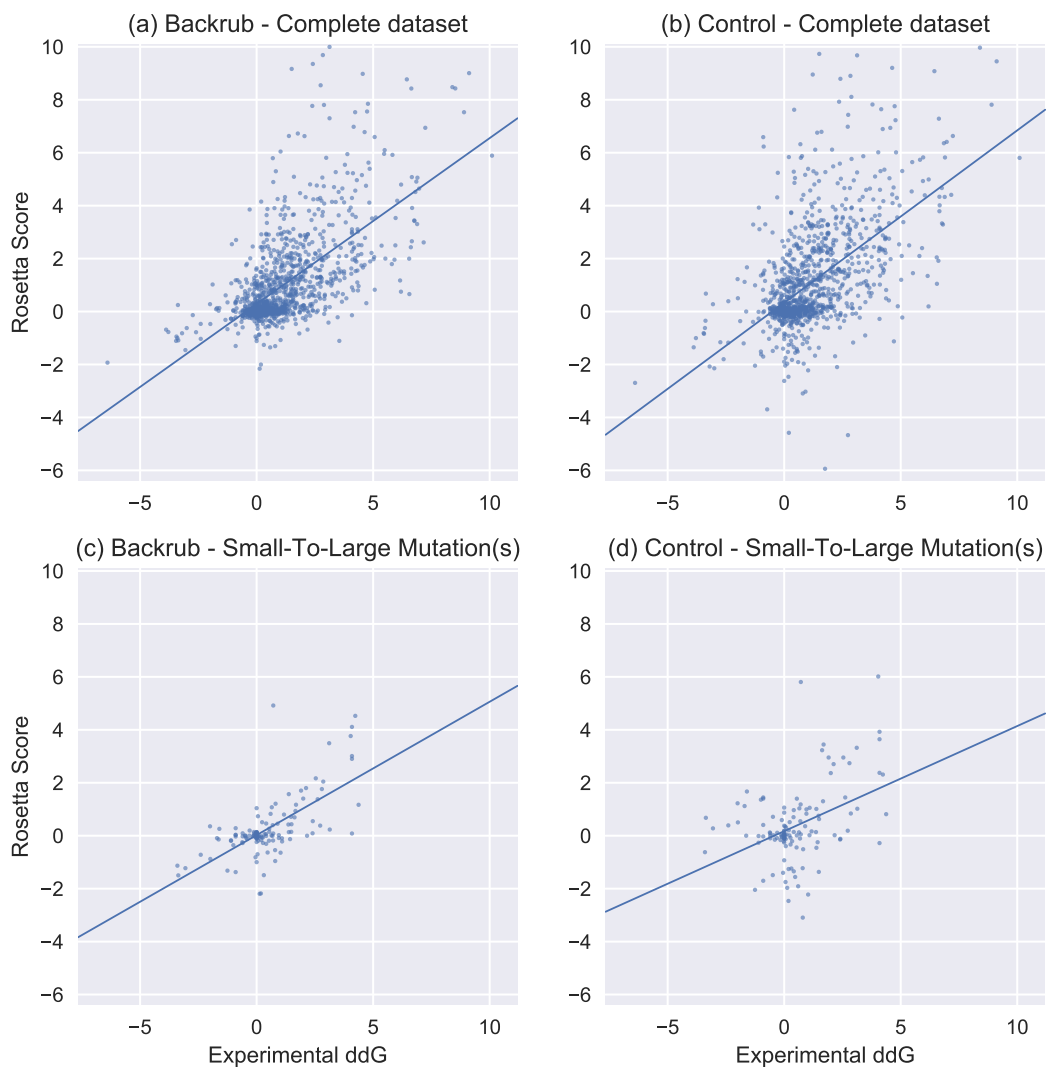


Figure 3.2: Experimentally determined  $\Delta\Delta G$  values (y-axis) vs. Rosetta predictions. (a) Flex ddg method (32500 backrub steps); Complete dataset mutation set (n=1240). (b) No backrub control; Complete dataset mutation set (n=1240). (c) Flex ddg method (32500 backrub steps); Small-to-large mutation(s) mutation set (n=130). (d) No backrub control; Small-to-large mutation(s) mutation set (n=130).

### Number of Structures Performance (structures sorted by minimized wild-type complex energy)

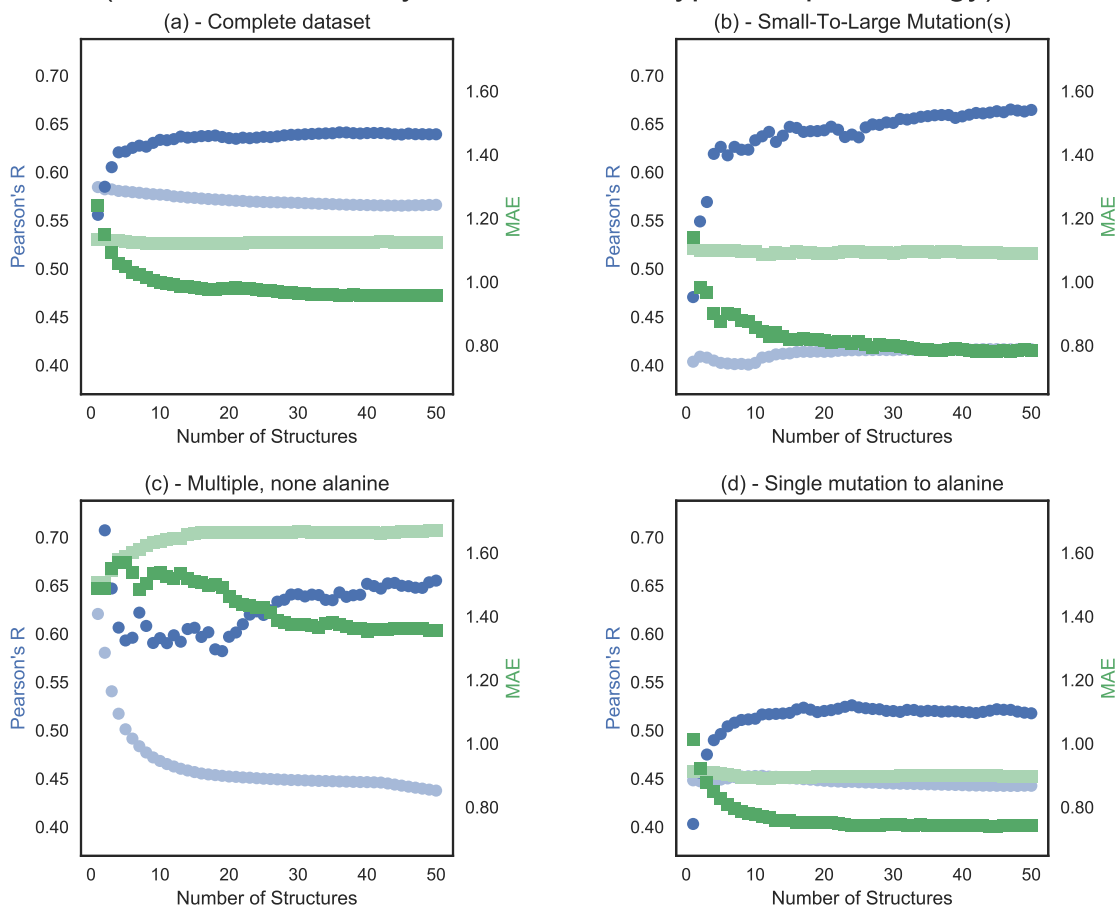


Figure 3.3: Correlation (Pearson’s R) and MAE (Mean Absolute Error) vs. number of averaged structures, on the complete ZEMu set, and subsets. Dark color lines represent “flex ddG” run. Light color lines represent “no backrub control” run. (a) Complete dataset ( $n = 1240$ , backrub step = 32500) (b) Small-to-large mutation(s) ( $n = 130$ , backrub step = 37500) (c) Multiple, none alanine ( $n = 45$ , backrub step = 40000) (d) Single mutation to alanine ( $n = 748$ , backrub step = 50000)

minimization with a fully weighted repulsive term.

Our observation that correlation with experimental  $\Delta\Delta G$  values improves as more structures are averaged is seen to an even greater degree ??(b) for the subset of small-to-large mutations. However, the subset of multiple mutations (where none are mutations to alanine) shown in ??(c) does not see monotonically increasing performance as more structures are averaged, indicating that more parameterization of the backrub method might be necessary for multiple mutations.

Averaging across increasing structures also improves correlation ??(d) for the subset of single mutations to alanines, a subset where it is not expected that increased sampling is necessary, indicating that increased sampling, in the very least, is not harmful for this case.

From a practical standpoint, simply generating 20-30 structures should constitute sufficient sampling for most use cases, as the performance when selecting the best scoring 20 out of 50 models is not significantly improved over the results in ?? (where there is no sorting of structures by score).

### 3.3.2 Effect of changing backrub sampling steps

Sampling can also be controlled by changing the length of the backrub simulation, as measured in the number of Monte Carlo sampling steps. ?? shows the effect on performance of increasing backrub simulation length (while averaging all 50 structures at each output step).  $\Delta\Delta G$  scores are calculated every 2,500 backrub steps (shown in circles for correlation and in squares for MAE). A “X” marks the performance with zero backrub steps (control minimization and side chain packing only method).

As we observed when averaging over more structures, increased performance is also seen in both correlation with experimental data and MAE as backrub simulation length increases for the subsets of small-to-large mutations (panel b) and multiple mutations, none to alanine (panel c). Performance reaches a maximum at 30,000 backrub steps, after which it levels off. Performance improves for the first step of 2500 backrub steps on the complete dataset and



for single mutations to alanine, but remains relatively flat afterwards.

The increased performance seen is not simply a result of scores decreasing as the simulation progresses, as the score of the minimized wild type complex does not decrease uniformly across the sampled ensemble as the simulation progresses (??). ?? shows that pairwise backrub ensemble RMSDs continue to increase throughout the backrub simulation for all subsets, indicating that diminishing returns at 30,000+ steps is not a result of failure to sample new states, but rather might indicate that no additional sampling is needed to capture the degree of local change in structure that occurs post-mutation in this benchmark set. ?? also shows that the starting pairwise ensemble RMSDs vary for each different subset, reflecting the fact that different subsets are composed of different wild-type complexes with different inherent flexibility (as sampled by backrub). This different inherent flexibility should be kept in mind when comparing results across subsets.

Unlike when increasing the number of averaged structures, we see continual improved performance with additional sampling (from longer backrub simulations) on the subset of multiple mutations (not to alanine). This indicates that sampling cannot be simply “increased” by tuning either the length of the backrub simulation or the number of models generated.

### 3.3.3 Score analysis

As the sampling/scoring problems of protein modeling are inextricably linked, it is often the case that improving one enables further improvements on the other. For example, increased sampling can exposed score function problems. We sought to analyze underlying errors of the Rosetta score function (when applied to interface  $\Delta\Delta G$ ) by reweighting the generally parameterized energy function for this specific application. Additionally, analyzing a reweighted energy function on a score term by score term basis could provide insight into which terms might benefit from future developments.

We chose to reweight the energy function using a non-linear reweighting scheme similar to

Generalized Additive Models (GAMs)<sup>107</sup>. In this reweighting method, we use Monte Carlo sampling to fit either a linear transformation or a sigmoid function to the individual distributions of score terms, with the objective function of reducing the absolute error between our predictions and known experimental values over the entire dataset.

As we do not modify our models of the unbound state, any effects on stability of the complex partners will cancel out, as the  $\Delta G$  of folding score of the unbound partners is subtracted from the total score of the complex (??). Not modeling conformational change in the unbound models might be effective because to modeling any such fluctuation might produce more noise than signal when the scores for the bound and unbound states are subtracted. This is supported by the prior observation that the mobility of amino acids at dimeric interfaces is generally lower than for other amino acids at the protein surface exposed to solvent<sup>108</sup>.

The terms in the Rosetta Talaris energy function that cancel out to zero are: `yhh_planarity`, `pro_close`, `hbond_sr_bb`, `ref`, `fa_dun`, `fa_intra_rep`, `omega`, `p_aa_pp`, and `rama`. Seven score terms are left; combined they become the final interface  $\Delta\Delta G$  score: `fa_sol`, `hbond_sc`, `hbond_bb_sc`, `fa_rep`, `fa_elec`, `hbond_lr_bb` and `fa_atr`.

The fit sigmoid and linear functions are shown in ???. The effect on the distribution of predictions is shown in ???.

## 3.4 Conclusions

We have shown that our new “flex ddG” method for estimating change in binding affinity after mutation in protein-protein interfaces is more effective than previous methods on a large, curated benchmark dataset. Particular improvement in performance is seen on the subset of small-to-large mutations, indicating that modeling backbone flexibility does improve performance in the case where backbone rearrangements are expected to be more common.

We have also shown more accurate predictions can be obtained by averaging the scores across a generated ensemble of backrub microstates, and that the number of required states is relatively low (20-30). Prior methods that attempted to produce  $\Delta\Delta G$  predictions by averaging an ensemble of models required on the order of thousands of structures<sup>45</sup>, indicating that backrub sampling can efficiently sample the local conformational landscape around a wild-type structure that is relevant for interface  $\Delta\Delta G$  prediction.

By creating a method that uses backrub to sample conformational space more broadly than minimization alone can sample, while still staying close in conformational space to the known wild-type input structure, we have also generated data that should prove useful for future energy function improvements. In particular, performance with Rosetta's newest REF energy function<sup>109</sup> is currently not better in our method than performance with the prior Talaris energy function (??), indicating that the backrub sampling parameters might require further benchmarking and adaptation to the REF energy function. Our error analysis via GAM-like reweighting also indicates potential score function improvement could be obtained via non-linear score term reweighting, and that examination of the weights we obtained for interface  $\Delta\Delta G$  prediction might provide insight into why the current energy function fails on certain cases in our dataset. Further improvements might also be obtained by more explicitly including the effects of entropy, including the potential to use our ensembles to calculate change in conformational entropy after mutation.

# Chapter 4

## Conclusion

In my thesis research, I have evaluated the performance of current state-of-the-art computational protein structure prediction and design methods that estimate the energetic effects of mutations, design protein sequences, and predict the structure of protein loops. The curated benchmark data that I assembled to evaluate each of these methods will also be of use for future benchmarking of Rosetta and non-Rosetta protocols.

I have advanced the ability of the Rosetta computational protein structure prediction and design software to more accurately represent proteins with conformational flexibility using ensemble-based approaches. This ensemble-based sampling approach has enabled improved performance in calculations of change in binding free energy post-mutation, particularly for cases of small-to-large mutations that were difficult to model with previous methods.

I look forward to the continued application of these new methods to the advancement of our knowledge of biology. I hope that my contribution to science will now join the greater body of work generated by countless others around the world, enabling science to continue to advance the health and well-being of humanity.

# Chapter 5

## Appendix

Mutation Category	Prediction Method	N	R	MAE	FC
Multiple mutations	flex ddG	273	0.62	1.51	0.77
	ZEMu paper		<b>0.64</b>	<b>1.46</b>	<b>0.78</b>
Multiple, all alanine	flex ddG	191	0.47	1.55	<b>0.85</b>
	ZEMu paper		<b>0.55</b>	<b>1.44</b>	<b>0.85</b>
Multiple, none alanine	flex ddG	45	<b>0.67</b>	<b>1.57</b>	<b>0.53</b>
	ZEMu paper		0.53	1.79	0.51
Mutation(s) to alanine	flex ddG	939	0.61	<b>0.89</b>	<b>0.77</b>
	ZEMu paper		<b>0.62</b>	0.90	<b>0.77</b>

Table 5.1: Multiple mutations results. R = Pearson’s R. MAE = Mean Absolute Error. FC = Fraction Correct. flex ddG steps = 35000.

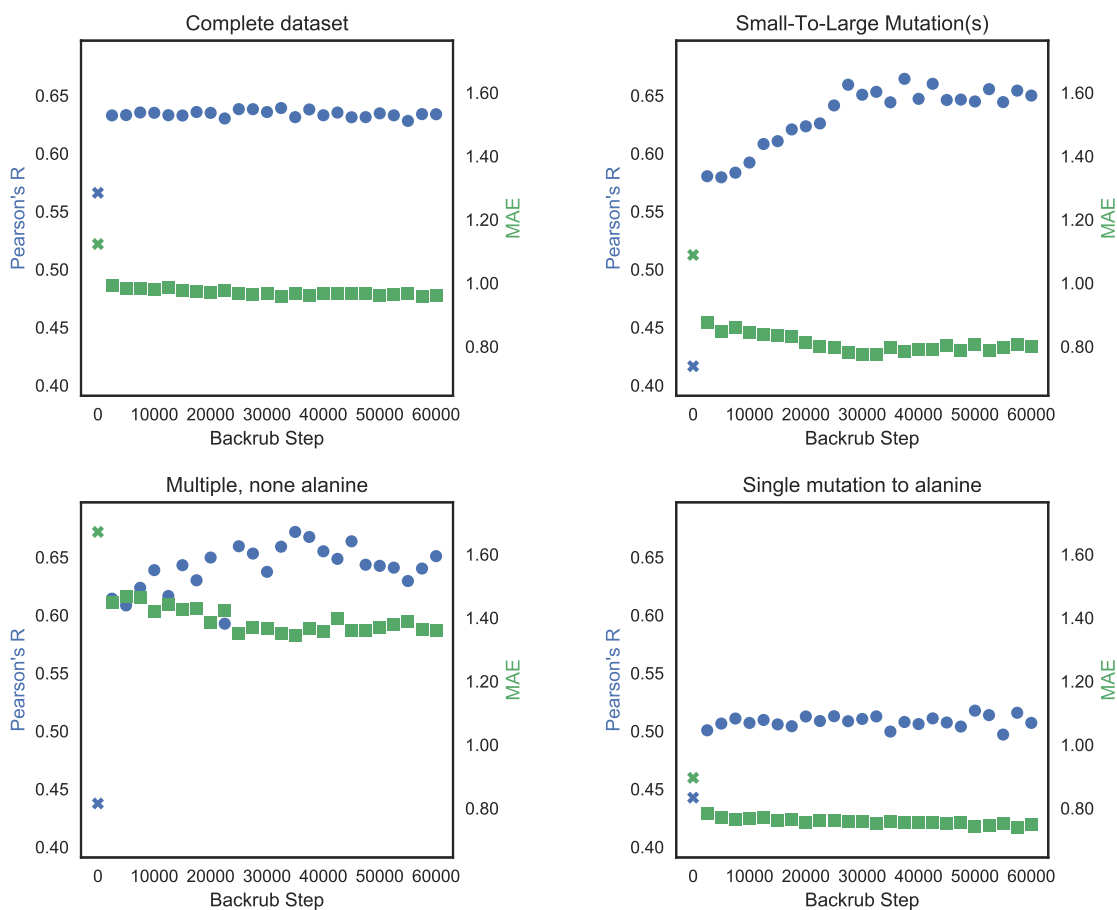


Figure 5.1: Correlation (Pearson’s R) and MAE (Mean Absolute Error) vs. number of backrub steps, on the complete ZEMu set, and subsets. “X” markers represent performance of control run (zero backrub steps). (a) Complete dataset (n=1240) (b) Small-to-large mutation(s) (n=130) (c) Multiple, none alanine (n=45) (d) Single mutation to alanine (n=748)

Git SHA1	Protocol
69aa5266f0d5	flex ddG
0c91ecd5bde5	no backrub control
3b2aa5cc3798	ddG monomer

Table 5.2: SHA1 Git version of Rosetta used for benchmarking

Mutation Category	Prediction Method	N	R	MAE	FC
Complete dataset	flex ddG		<b>0.64</b>	<b>0.92</b>	0.76
	ddG monomer (hard-rep)	1240	0.62	0.94	<b>0.77</b>
	ZEMu paper		0.61	0.96	0.75
Antibodies	flex ddG		<b>0.62</b>	<b>0.88</b>	0.75
	ddG monomer (hard-rep)	355	0.58	0.90	<b>0.77</b>
	ZEMu paper		0.54	0.96	0.74

Table 5.3: Performance of the Rosetta flex ddG method on the subset of complexes containing an antibody binding partner. R = Pearson’s R. MAE = Mean Absolute Error. FC = Fraction Correct. flex ddG steps = 32500.

Mutation Category	Prediction Method	N	R	MAE	FC
Complete dataset	flex ddG	1240	<b>0.63</b>	<b>0.93</b>	<b>0.76</b>
	flex ddG (REF energy)		<b>0.63</b>	<b>0.93</b>	<b>0.76</b>
Small-To-Large Mutation(s)	flex ddG	130	<b>0.64</b>	<b>0.87</b>	<b>0.75</b>
	flex ddG (REF energy)		0.57	0.92	0.72
Single mutation to alanine	flex ddG	748	<b>0.50</b>	<b>0.72</b>	0.75
	flex ddG (REF energy)		0.49	0.73	<b>0.76</b>
Multiple mutations	flex ddG	273	<b>0.62</b>	<b>1.51</b>	<b>0.77</b>
	flex ddG (REF energy)		0.59	1.57	0.75
Res. $\rho = 1.5$ Ang.	flex ddG	52	0.46	0.85	0.75
	flex ddG (REF energy)		<b>0.65</b>	<b>0.74</b>	<b>0.77</b>
Res. $\rho = 2.5$ Ang.	flex ddG	457	<b>0.50</b>	<b>0.74</b>	0.74
	flex ddG (REF energy)		0.48	0.75	<b>0.76</b>

Table 5.4: REF results. R = Pearson’s R. MAE = Mean Absolute Error. FC = Fraction Correct. flex ddG steps = 35000. flex ddG (REF energy) steps = 35000.

## Number of Structures Performance (no sorting of structures)

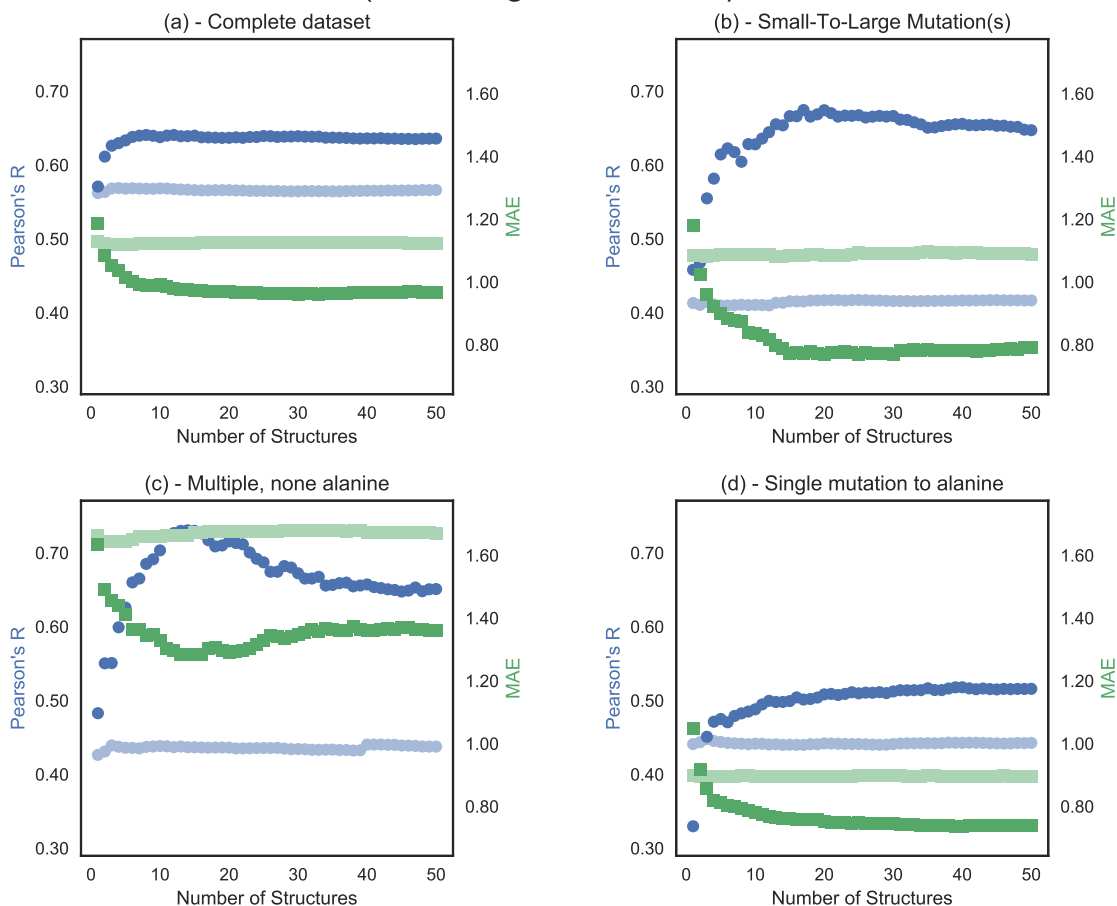


Figure 5.2: Correlation (Pearson's R) and MAE (Mean Absolute Error) vs. number of averaged structures, on the complete ZEMu set, and subsets. Dark color lines represent "flex ddG" run. Light color lines represent "no backrub control" run. (a) Complete dataset ( $n = 1240$ , backrub step = 30000) (b) Small-to-large mutation(s) ( $n = 130$ , backrub step = 40000) (c) Multiple, none alanine ( $n = 45$ , backrub step = 60000) (d) Single mutation to alanine ( $n = 748$ , backrub step = 57500)



### Number of Structures Performance (structures sorted by minimized wild-type complex energy)

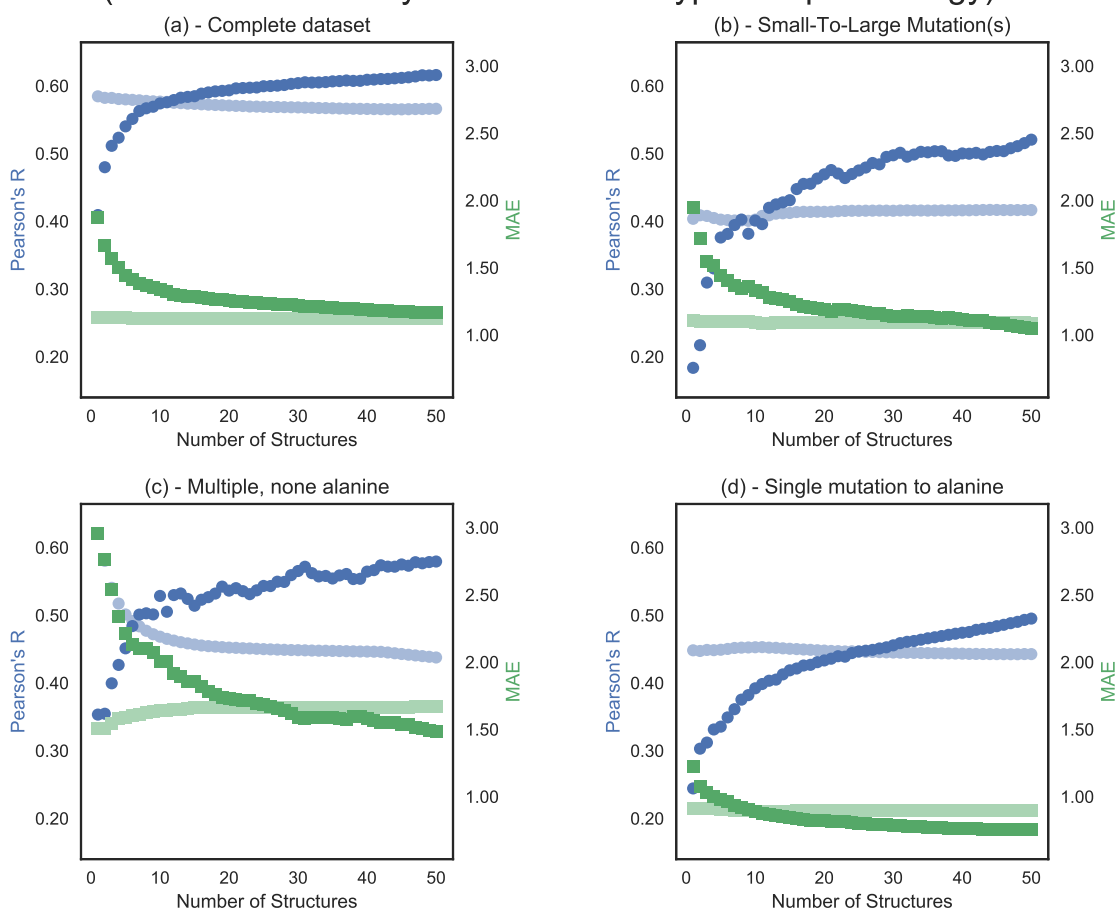


Figure 5.3: Correlation (Pearson’s R) and MAE (Mean Absolute Error) vs. number of averaged structures, on the complete ZEMu set, and subsets. Dark color lines represent “ddG monomer” run. Light color lines represent “no backrub control” run. (a) Complete dataset (n = 1240) (b) Small-to-large mutation(s) (n = 130) (c) Multiple, none alanine (n = 45) (d) Single mutation to alanine (n = 748)

\_1.2-60000\_rscript\_validated-t14 - Step vs. nstruct vs. WildTypeComplex mean total score (n=1240)

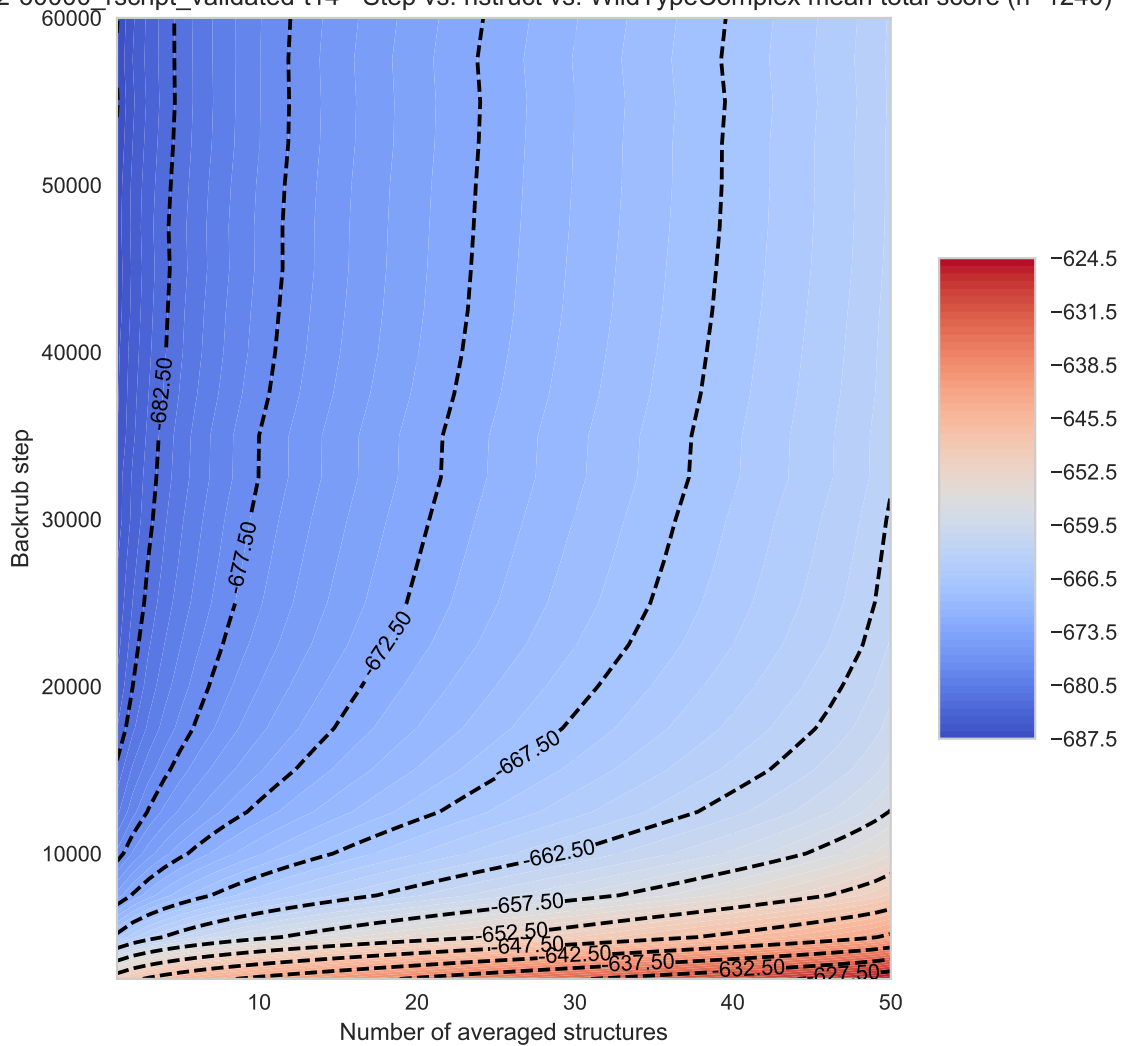


Figure 5.4: Contour plot showing the effect of backrub sampling on the average wild-type complex score, for varying numbers of averaged models.

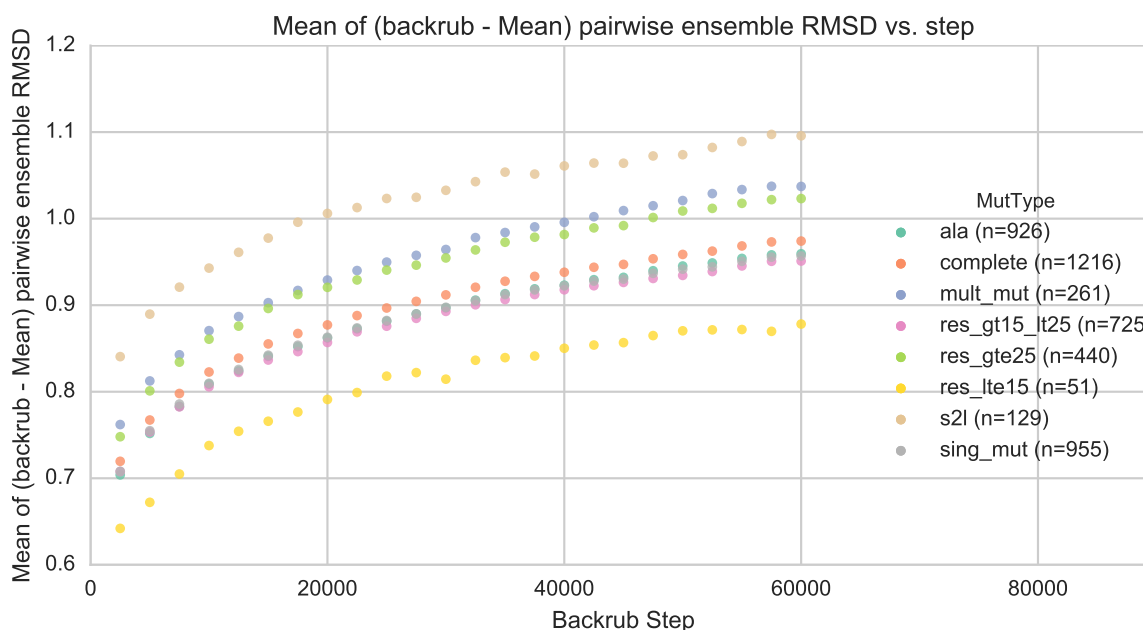


Figure 5.5: Mean backrub ensemble RMSD vs. backrub steps.

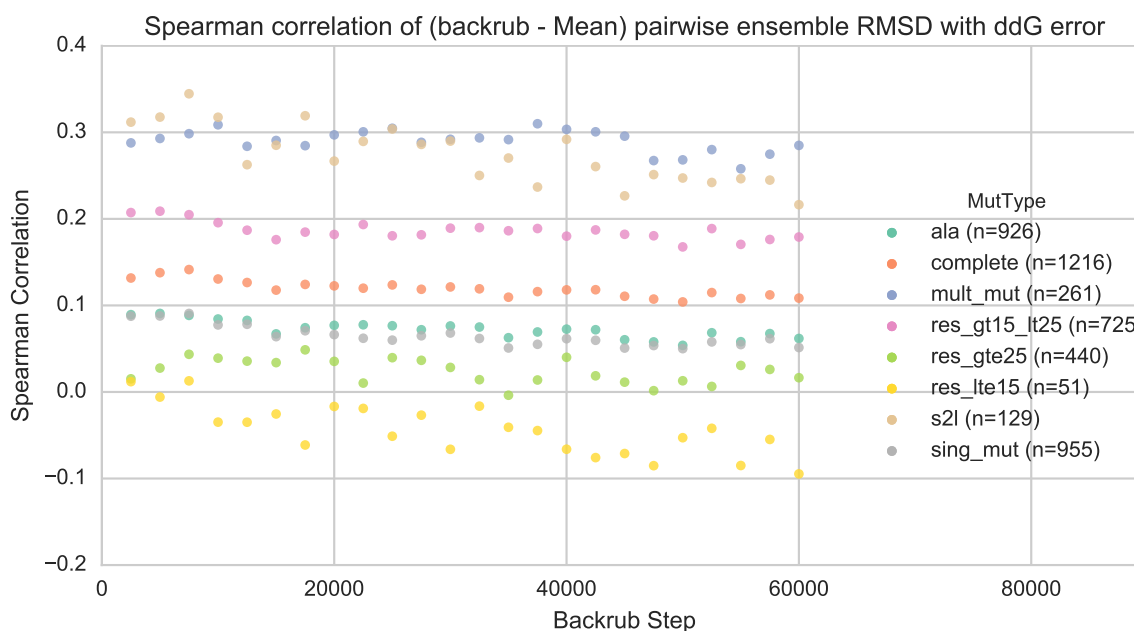


Figure 5.6: Scatter plot showing the average Spearman correlation of ddG prediction error v. mean pairwise backrub ensemble RMSD, v. backrub steps. As mean backrub ensemble RMSD increases (??), we don't see any significant change in correlation between mean ensemble RMSD and ddG error. This demonstrates that mean pairwise backrub ensemble RMSD is not an effective metric to measure the degree to which we have sampled "enough".

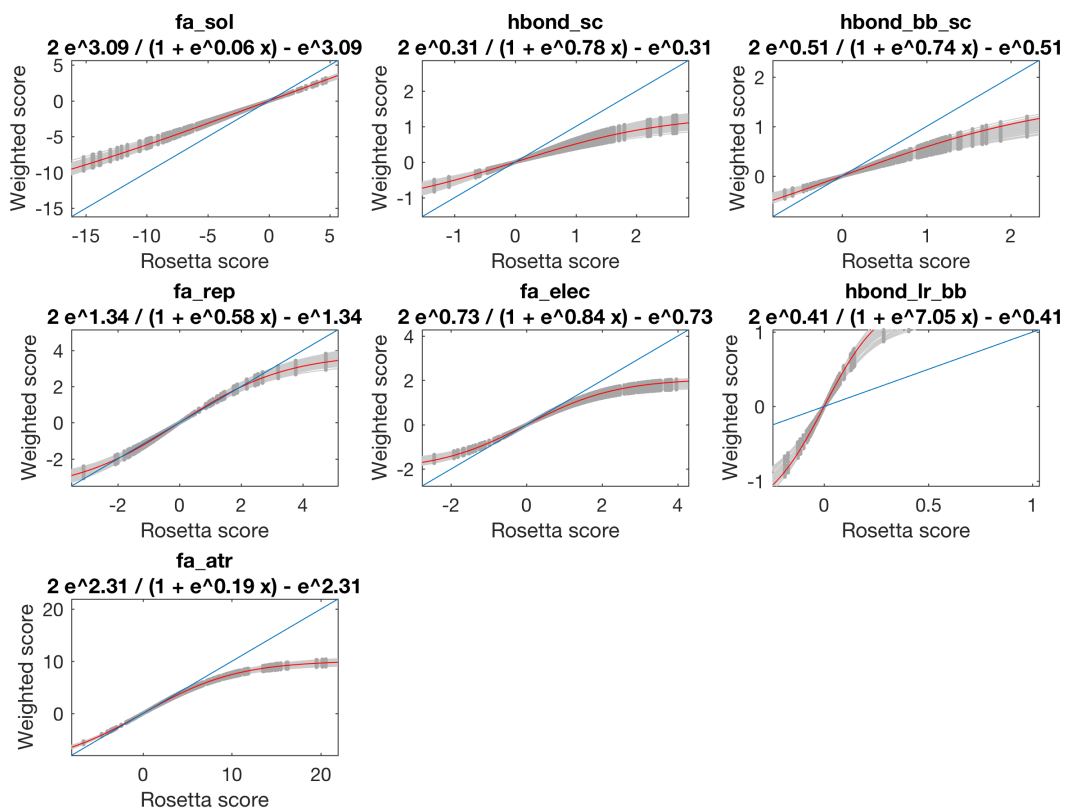


Figure 5.7: Rosetta interface  $\Delta\Delta G$  score function terms fit on interface  $\Delta\Delta G$  predictions via a sigmoid fit Generalized Additive Model. (Figure courtesy Markus Heinonen)

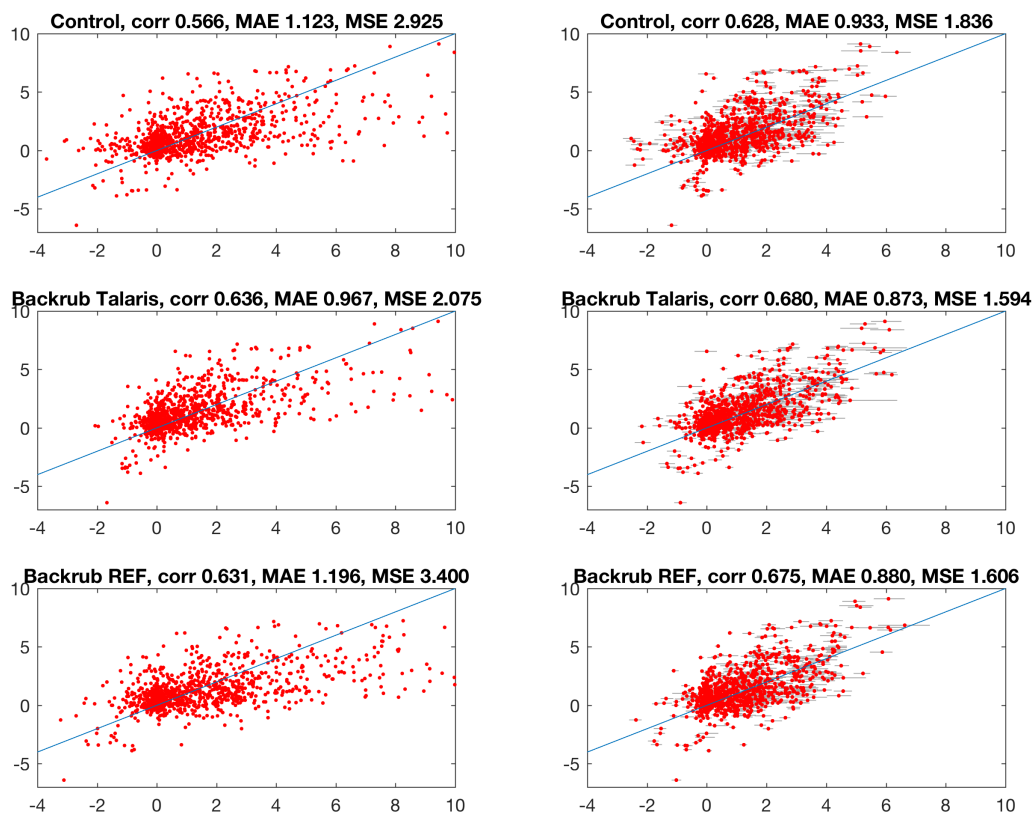


Figure 5.8: Left: standard, non-fitted predictions vs. experimental  $\Delta\Delta G$  values. Right: Fit predictions vs. experimental data. Top: Control (no backrub) predictions. Middle: Backrub/talaris. Bottom: Backrub/REF. (Figure courtesy Markus Heinonen)

Listing 5.1: Flex ddg Rosetta Script implementation

```

1 <ROSETTASCRIPTS>
2   <SCOREFXNS>
3     <ScoreFunction name="fa_talaris2013" weights="talaris2013"/>
4     <ScoreFunction name="fa_talaris2014" weights="talaris2014"/>
5     <ScoreFunction name="fa_talaris2014_cst" weights="talaris2014">
6       <Reweight scoretype="atom_pair_constraint" weight="1.0"/>
7       <Set fa_max_dis="9.0"/>
8     </ScoreFunction>
9 </SCOREFXNS>
10
11 <!-- #### All residues must be set to be NATAA packable at top of
12     ↪ resfile ### -->
13 <TASKOPERATIONS>
14   <ReadResfile name="res_mutate" filename="%mutate_resfile_relpath%"/>
15 </TASKOPERATIONS>
16 <RESIDUE_SELECTORS>
17   <Task name="resselector" fixed="0" packable="0" designable="1"
18     ↪ task_operations="res_mutate"/>
19   <Neighborhood name="bubble" selector="resselector" distance="8.0"/>
20   <ResidueName name="isgly" residue_name3="GLY"/>
21   <Not name="notgly" selector="isgly"/>
22   <And name="bubble_notgly" selectors="bubble,notgly"/>
23   <PrimarySequenceNeighborhood name="bubble_notgly_adjacent"
24     ↪ selector="bubble_notgly" lower="1" upper="1"/>

```

```

23     <StoredResidueSubset name="restore_neighbor_shell"
      ↪ subset_name="neighbor_shell"/>
24     <Not name="everythingelse" selector="restore_neighbor_shell"/>
25 </RESIDUE_SELECTORS>
26 <TASKOPERATIONS>
27     <OperateOnResidueSubset name="repackonly"
      ↪ selector="restore_neighbor_shell">
28         <RestrictToRepackingRLT/>
29     </OperateOnResidueSubset>
30     <OperateOnResidueSubset name="norepack" selector="everythingelse">
31         <PreventRepackingRLT/>
32     </OperateOnResidueSubset>
33     <UseMultiCoolAnnealer name="multicool" states="6"/>
34     <ExtraChiCutoff name="extrachizero" extrachi_cutoff="0"/>
35     <InitializeFromCommandline name="commandline_init"/>
36     <RestrictToRepacking name="restrict_to_repacking"/>
37 </TASKOPERATIONS>
38
39 <FILTERS>
40 </FILTERS>
41
42 <MOVERS>
43     <StoreResidueSubset name="neighbor_shell_storer"
      ↪ subset_name="neighbor_shell"
      ↪ residue_selector="bubble_notgly_adjacent" />
44

```

```

45 <AddConstraintsToCurrentConformationMover name="addcst"
    ↪ use_distance_cst="1" coord_dev="0.5" min_seq_sep="0"
    ↪ max_distance="9" CA_only="1" bound_width="0.0" cst_weight="0.0"/>
46 <ClearConstraintsMover name="clearcst"/>
47 <MinMover name="minimize" scorefxn="fa_talaris2014_cst" chi="1" bb="1"
    ↪ type="lbfgs_armijo_nonmonotone" tolerance="0.000001" max_iter="5000"
    ↪ abs_score_convergence_threshold="1.0"/>
48
49 <PackRotamersMover name="repack" scorefxn="fa_talaris2014"
    ↪ task_operations="commandline_init,repackonly,norepack,multicool"/>
50 <PackRotamersMover name="mutate" scorefxn="fa_talaris2014"
    ↪ task_operations="commandline_init,res_mutate,norepack,multicool"/>
51
52 <ReportToDB name="dbreport" batch_description="interface_ddG"
    ↪ database_name="ddG.db3">
53   <PdbDataFeatures/>
54   <ScoreTypeFeatures/>
55   <ScoreFunctionFeatures scorefxn="fa_talaris2013"/>
56   <StructureScoresFeatures scorefxn="fa_talaris2013"/>
57   <ResidueFeatures/>
58   <ResidueScoresFeatures scorefxn="fa_talaris2013"/>
59   <PoseConformationFeatures/>
60   <ResidueConformationFeatures/>
61   <ProteinResidueConformationFeatures/>
62   <PairFeatures/>
63   <AtomAtomPairFeatures/>
64   <ProteinBondGeometryFeatures/>

```



```

65     <ProteinBackboneTorsionAngleFeatures/>
66     <RotamerFeatures/>
67 </ReportToDB>
68
69 <SavePoseMover name="save_wt_bound_pose" restore_pose="0"
    ↪ reference_name="wt_bound_pose"/>
70 <SavePoseMover name="save_backrub_pose" restore_pose="0"
    ↪ reference_name="backrubpdb"/>
71 <SavePoseMover name="restore_backrub_pose" restore_pose="1"
    ↪ reference_name="backrubpdb"/>
72
73 <BackrubProtocol name="backrub" mc_kt="1.2" ntrials="20000"
    ↪ pivot_residue_selector="restore_neighbor_shell"
    ↪ task_operations="restrict_to_repacking,commandline_init,extrachizero"
    ↪ recover_low="0"/>
74
75 <InterfaceDdGMover name="int_ddG_mover"
    ↪ wt_ref_savepose_mover="save_wt_bound_pose" db_reporter="dbreport"
    ↪ scorefxn="fa_talaris2013"/>
76
77 <ScoreMover name="apply_score" scorefxn="fa_talaris2014_cst"
    ↪ verbose="0"/>
78
79 </MOVERS>
80 <APPLY_TO_POSE>
81 </APPLY_TO_POSE>
82 <PROTOCOLS>

```

```
83 <Add mover_name="addcst"/>
84 <Add mover_name="apply_score"/> <!-- Necessary to initialize neighbor
   ↪ graph -->
85 <Add mover_name="neighbor_shell_storer"/>
86
87 <Add mover_name="minimize"/>
88 <Add mover_name="clearcst"/>
89
90 <Add mover_name="backrub"/>
91 <Add mover_name="save_backrub_pose"/>
92
93 <Add mover_name="repack"/>
94
95 <Add mover_name="addcst"/>
96 <Add mover_name="minimize"/>
97 <Add mover_name="clearcst"/>
98
99 <Add mover_name="save_wt_bound_pose"/>
100 <Add mover_name="restore_backrub_pose"/>
101
102 <Add mover_name="mutate"/>
103
104 <Add mover_name="addcst"/>
105 <Add mover_name="minimize"/>
106 <Add mover_name="clearcst"/>
107
108 <Add mover_name="int_ddG_mover"/>
```

```
109 </PROTOCOLS>
110 <OUTPUT />
111 </ROSETTASCRIPTS>
```

# Bibliography

- [1] Shane Ó Conchúir, Kyle A. Barlow, Roland A. Pache, Noah Ollikainen, Kale Kundert, Matthew J. O’Meara, Colin A. Smith, and Tanja Kortemme. A Web Resource for Standardized Benchmark Datasets, Metrics, and Rosetta Protocols for Macromolecular Modeling and Design. *PLOS ONE*, 10(9):e0130433, September 2015. ISSN 1932-6203. doi: 10.1371/journal.pone.0130433. URL <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0130433>. iii, 34
- [2] Oswald T. Avery, Colin M. MacLeod, and Maclyn McCarty. Studies on the Chemical Nature of the Substance Inducing Transformation of Pneumococcal Types: Induction of Transformation by a Desoxyribonucleic Acid Fraction Isolated from Pneumococcus Type Iii. *Journal of Experimental Medicine*, 79(2):137–158, February 1944. ISSN 0022-1007, 1540-9538. doi: 10.1084/jem.79.2.137. URL <http://jem.rupress.org/content/79/2/137>. 1
- [3] F. H. C. Crick. The origin of the genetic code. *Journal of Molecular Biology*, 38(3): 367–379, December 1968. ISSN 0022-2836. doi: 10.1016/0022-2836(68)90392-6. URL <http://www.sciencedirect.com/science/article/pii/0022283668903926>. 1
- [4] R. T. Hinegardner and J. Engelberg. RATIONALE FOR A UNIVERSAL GENETIC CODE. *Science (New York, N.Y.)*, 142(3595):1083–1085, November 1963. ISSN 0036-8075. 1
- [5] CR Darwin. *On the Origin of Species*. London: John Murray, 1859. 1

- [6] Eric S. Lander, Lauren M. Linton, Bruce Birren, Chad Nusbaum, Michael C. Zody, Jennifer Baldwin, Keri Devon, Ken Dewar, Michael Doyle, William FitzHugh, Roel Funke, Diane Gage, Katrina Harris, Andrew Heaford, John Howland, Lisa Kann, Jessica Lehoczky, Rosie LeVine, Paul McEwan, Kevin McKernan, James Meldrim, Jill P. Mesirov, Cher Miranda, William Morris, Jerome Naylor, Christina Raymond, Mark Rosetti, Ralph Santos, Andrew Sheridan, Carrie Sougnez, Nicole Stange-Thomann, Nikola Stojanovic, Aravind Subramanian, Dudley Wyman, Jane Rogers, John Sulston, Rachael Ainscough, Stephan Beck, David Bentley, John Burton, Christopher Clee, Nigel Carter, Alan Coulson, Rebecca Deadman, Panos Deloukas, Andrew Dunham, Ian Dunham, Richard Durbin, Lisa French, Darren Grafham, Simon Gregory, Tim Hubbard, Sean Humphray, Adrienne Hunt, Matthew Jones, Christine Lloyd, Amanda McMurray, Lucy Matthews, Simon Mercer, Sarah Milne, James C. Mullikin, Andrew Mungall, Robert Plumb, Mark Ross, Ratna Shownkeen, Sarah Sims, Robert H. Waterston, Richard K. Wilson, LaDeana W. Hillier, John D. McPherson, Marco A. Marra, Elaine R. Mardis, Lucinda A. Fulton, Asif T. Chinwalla, Kymberlie H. Pepin, Warren R. Gish, Stephanie L. Chissoe, Michael C. Wendl, Kim D. Delehaunty, Tracie L. Miner, Andrew Delehaunty, Jason B. Kramer, Lisa L. Cook, Robert S. Fulton, Douglas L. Johnson, Patrick J. Minx, Sandra W. Clifton, Trevor Hawkins, Elbert Branscomb, Paul Predki, Paul Richardson, Sarah Wenning, Tom Slezak, Norman Doggett, Jan-Fang Cheng, Anne Olsen, Susan Lucas, Christopher Elkin, Edward Uberbacher, Marvin Frazier, Richard A. Gibbs, Donna M. Muzny, Steven E. Scherer, John B. Bouck, Erica J. Sodergren, Kim C. Worley, Catherine M. Rives, James H. Gorrell, Michael L. Metzker, Susan L. Naylor, Raju S. Kucherlapati, David L. Nelson, George M. Weinstock, Yoshiyuki Sakaki, Asao Fujiyama, Masahira Hattori, Tetsushi Yada, Atsushi Toyoda, Takehiko Itoh, Chiharu Kawagoe, Hidemi Watanabe, Yasushi Totoki, Todd Taylor, Jean Weissenbach, Roland Heilig, William Saurin, Francois Artiguenave, Philippe Brottier, Thomas Bruls, Eric Pelletier, Cather-

ine Robert, Patrick Wincker, André Rosenthal, Matthias Platzer, Gerald Nyakatura, Stefan Taudien, Andreas Rump, Douglas R. Smith, Lynn Doucette-Stamm, Marc Rubenfield, Keith Weinstock, Hong Mei Lee, JoAnn Dubois, Huanming Yang, Jun Yu, Jian Wang, Guyang Huang, Jun Gu, Leroy Hood, Lee Rowen, Anup Madan, Shizen Qin, Ronald W. Davis, Nancy A. Federspiel, A. Pia Abola, Michael J. Proctor, Bruce A. Roe, Feng Chen, Huaqin Pan, Juliane Ramser, Hans Lehrach, Richard Reinhardt, W. Richard McCombie, Melissa de la Bastide, Neilay Dedhia, Helmut Blöcker, Klaus Hornischer, Gabriele Nordsiek, Richa Agarwala, L. Aravind, Jeffrey A. Bailey, Alex Bateman, Serafim Batzoglou, Ewan Birney, Peer Bork, Daniel G. Brown, Christopher B. Burge, Lorenzo Cerutti, Hsiu-Chuan Chen, Deanna Church, Michele Clamp, Richard R. Copley, Tobias Doerks, Sean R. Eddy, Evan E. Eichler, Terrence S. Furey, James Galagan, James G. R. Gilbert, Cyrus Harmon, Yoshihide Hayashizaki, David Haussler, Henning Hermjakob, Karsten Hokamp, Wonhee Jang, L. Steven Johnson, Thomas A. Jones, Simon Kasif, Arek Kasprzyk, Scot Kennedy, W. James Kent, Paul Kitts, Eugene V. Koonin, Ian Korf, David Kulp, Doron Lancet, Todd M. Lowe, Aoife McLysaght, Tarjei Mikkelsen, John V. Moran, Nicola Mulder, Victor J. Pollara, Chris P. Ponting, Greg Schuler, Jörg Schultz, Guy Slater, Arian F. A. Smit, Elia Stupka, Joseph Szustakowki, Danielle Thierry-Mieg, Jean Thierry-Mieg, Lukas Wagner, John Wallis, Raymond Wheeler, Alan Williams, Yuri I. Wolf, Kenneth H. Wolfe, Shiaw-Pyng Yang, Ru-Fang Yeh, Francis Collins, Mark S. Guyer, Jane Peterson, Adam Felsenfeld, Kris A. Wetterstrand, Richard M. Myers, Jeremy Schmutz, Mark Dickson, Jane Grimwood, David R. Cox, Maynard V. Olson, Rajinder Kaul, Christopher Raymond, Nobuyoshi Shimizu, Kazuhiko Kawasaki, Shinsei Minoshima, Glen A. Evans, Maria Athanasiou, Roger Schultz, Aristides Patrinos, and Michael J. Morgan. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, February 2001. ISSN 0028-0836. doi: 10.1038/35057062. URL <https://www.nature.com/nature/journal/v409/n6822/full/409860a0.html>. 1

- [7] J. C. Kendrew, G. Bodo, H. M. Dintzis, R. G. Parrish, H. Wyckoff, and D. C. Phillips. A Three-Dimensional Model of the Myoglobin Molecule Obtained by X-Ray Analysis. *Nature*, 181(4610):662–666, March 1958. ISSN 0028-0836. doi: 10.1038/181662a0. URL <https://www.nature.com/nature/journal/v181/n4610/abs/181662a0.html>. 1
- [8] M. F. Perutz, M. G. Rossmann, Ann F. Cullis, Hilary Muirhead, Georg Will, and A. C. T. North. Structure of Hæmoglobin: A Three-Dimensional Fourier Synthesis at 5.5- $\text{\AA}$  Resolution, Obtained by X-Ray Analysis. *Nature*, 185(4711):416–422, February 1960. ISSN 0028-0836. doi: 10.1038/185416a0. URL <https://www.nature.com/nature/journal/v185/n4711/abs/185416a0.html>. 1
- [9] Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, January 2000. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/28.1.235. URL <http://nar.oxfordjournals.org/content/28/1/235>. 1, 7
- [10] UniProt: the universal protein knowledgebase. *Nucleic Acids Research*, 45(D1):D158–D169, January 2017. ISSN 0305-1048. doi: 10.1093/nar/gkw1099. URL <https://academic.oup.com/nar/article/45/D1/D158/2605721/UniProt-the-universal-protein-knowledgebase>. 1
- [11] Walter Pirovano and Jaap Heringa. Protein Secondary Structure Prediction. In *Data Mining Techniques for the Life Sciences*, Methods in Molecular Biology, pages 327–348. Humana Press, 2010. ISBN 978-1-60327-240-7 978-1-60327-241-4. URL [https://link.springer.com/protocol/10.1007/978-1-60327-241-4\\_19](https://link.springer.com/protocol/10.1007/978-1-60327-241-4_19). DOI: 10.1007/978-1-60327-241-4\_19. 2
- [12] Yong Duan, Chun Wu, Shibasish Chowdhury, Mathew C. Lee, Guoming Xiong, Wei Zhang, Rong Yang, Piotr Cieplak, Ray Luo, Taisung Lee, James Caldwell, Junmei

- Wang, and Peter Kollman. A point-charge force field for molecular mechanics simulations of proteins based on condensed-phase quantum mechanical calculations. *Journal of Computational Chemistry*, 24(16):1999–2012, December 2003. ISSN 1096-987X. doi: 10.1002/jcc.10349. URL <http://onlinelibrary.wiley.com/doi/10.1002/jcc.10349/abstract>. 3
- [13] B. R. Brooks, C. L. Brooks, A. D. Mackerell, L. Nilsson, R. J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caffisch, L. Caves, Q. Cui, A. R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoseck, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R. W. Pastor, C. B. Post, J. Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D. M. York, and M. Karplus. CHARMM: The biomolecular simulation program. *Journal of Computational Chemistry*, 30(10):1545–1614, July 2009. ISSN 1096-987X. doi: 10.1002/jcc.21287. URL <http://onlinelibrary.wiley.com/doi/10.1002/jcc.21287/abstract>. 3, 7
- [14] Themis Lazaridis and Martin Karplus. Effective energy function for proteins in solution. *Proteins: Structure, Function, and Bioinformatics*, 35(2):133–152, May 1999. ISSN 1097-0134. doi: 10.1002/(SICI)1097-0134(19990501)35:2<133::AID-PROT1>3.0.CO;2-N. URL [http://onlinelibrary.wiley.com/doi/10.1002/\(SICI\)1097-0134\(19990501\)35:2<133::AID-PROT1>3.0.CO;2-N/abstract](http://onlinelibrary.wiley.com/doi/10.1002/(SICI)1097-0134(19990501)35:2<133::AID-PROT1>3.0.CO;2-N/abstract). 3
- [15] J. E. Lennard-Jones. On the Determination of Molecular Fields. II. From the Equation of State of a Gas. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 106(738):463–477, October 1924. ISSN 1364-5021, 1471-2946. doi: 10.1098/rspa.1924.0082. URL <http://rspa.royalsocietypublishing.org/content/106/738/463>. 3
- [16] Carol A. Rohl, Charlie E.M. Strauss, Kira M.S. Misura, and David Baker. Protein Structure Prediction Using Rosetta. In *Numerical Computer Methods, Part D*, volume



- Volume 383, pages 66–93. Academic Press, 2004. ISBN 0076-6879. URL <http://www.sciencedirect.com/science/article/pii/S0076687904830040>. 3
- [17] Cyrus Levinthal. How to Fold Graciously. pages 22–24. University of Illinois Press, 1969. 3
- [18] M. Karplus. The Levinthal paradox: yesterday and today. *Folding & Design*, 2(4): S69–75, 1997. ISSN 1359-0278. 3
- [19] Lin Jiang, Eric A. Althoff, Fernando R. Clemente, Lindsey Doyle, Daniela Röthlisberger, Alexandre Zanghellini, Jasmine L. Gallaher, Jamie L. Betker, Fujie Tanaka, Carlos F. Barbas, Donald Hilvert, Kendall N. Houk, Barry L. Stoddard, and David Baker. De Novo Computational Design of Retro-Aldol Enzymes. *Science*, 319(5868):1387–1391, March 2008. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1152692. URL <http://www.sciencemag.org/content/319/5868/1387>. 3
- [20] Justin B. Siegel, Alexandre Zanghellini, Helena M. Lovick, Gert Kiss, Abigail R. Lambert, Jennifer L. St.Clair, Jasmine L. Gallaher, Donald Hilvert, Michael H. Gelb, Barry L. Stoddard, Kendall N. Houk, Forrest E. Michael, and David Baker. Computational Design of an Enzyme Catalyst for a Stereoselective Bimolecular Diels-Alder Reaction. *Science*, 329(5989):309–313, July 2010. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1190239. URL <http://science.sciencemag.org/content/329/5989/309>. 3
- [21] Brian Kuhlman, Gautam Dantas, Gregory C. Ireton, Gabriele Varani, Barry L. Stoddard, and David Baker. Design of a Novel Globular Protein Fold with Atomic-Level Accuracy. *Science*, 302(5649):1364–1368, November 2003. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1089427. URL <http://www.sciencemag.org/content/302/5649/1364>. 4
- [22] Ian W. Davis, W. Bryan Arendall, David C. Richardson, and Jane S. Richardson. The

- Backrub Motion: How Protein Backbone Shrugs When a Sidechain Dances. *Structure*, 14(2):265–274, February 2006. ISSN 0969-2126. doi: 10.1016/j.str.2005.10.007. URL <http://www.sciencedirect.com/science/article/pii/S0969212606000402>. 4, 35
- [23] Daniel J Mandell, Evangelos A Coutsiias, and Tanja Kortemme. Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling. *Nature Methods*, 6(8):551–552, 2009. ISSN 1548-7091. doi: 10.1038/nmeth0809-551. URL <http://www.nature.com/nmeth/journal/v6/n8/full/nmeth0809-551.html>. 23, 26, 27
- [24] Gregory D. Friedland, Anthony J. Linares, Colin A. Smith, and Tanja Kortemme. A Simple Model of Backbone Flexibility Improves Modeling of Side-chain Conformational Variability. *Journal of Molecular Biology*, 380(4):757–774, July 2008. ISSN 0022-2836. doi: 10.1016/j.jmb.2008.05.006. URL <http://www.sciencedirect.com/science/article/pii/S0022283608005597>. 4, 35
- [25] Kim T. Simons, Charles Kooperberg, Enoch Huang, and David Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions<sup>11</sup>edited by F. E. Cohen. *Journal of Molecular Biology*, 268(1):209–225, April 1997. ISSN 0022-2836. doi: 10.1006/jmbi.1997.0959. URL <http://www.sciencedirect.com/science/article/pii/S0022283697909591>. 4
- [26] Ken A. Dill and Hue Sun Chan. From Levinthal to pathways to funnels. *Nature Structural & Molecular Biology*, 4(1):10–19, January 1997. doi: 10.1038/nsb0197-10. URL <https://www.nature.com/nsmb/journal/v4/n1/abs/nsb0197-10.html>. 4
- [27] Katherine Henzler-Wildman and Dorothee Kern. Dynamic personalities of proteins. *Nature*, 450(7172):964–972, December 2007. ISSN 0028-0836. doi: 10.1038/

nature06522. URL <https://www.nature.com/nature/journal/v450/n7172/full/nature06522.html>. 4

- [28] David Mavor, Kyle Barlow, Samuel Thompson, Benjamin A. Barad, Alain R. Bonny, Clinton L. Cario, Garrett Gaskins, Zairan Liu, Laura Deming, Seth D. Axen, Elena Caceres, Weilin Chen, Adolfo Cuesta, Rachel Gate, Evan M. Green, Kaitlin R. Hulce, Weiyue Ji, Lillian R. Kenner, Bruk Mensa, Leanna S. Morinishi, Steven M. Moss, Marco Mravic, Ryan K. Muir, Stefan Niekamp, Chimno I. Nnadi, Eugene Palovcak, Erin M. Poss, Tyler D. Ross, Eugenia C. Salcedo, Stephanie See, Meena Subramaniam, Allison W. Wong, Jennifer Li, Kurt S. Thorn, Shane Thomas Ó Conchúir, Benjamin P. Roscoe, Eric D. Chow, Joseph L. DeRisi, Tanja Kortemme, Daniel NA Bolon, and James S. Fraser. Determination of ubiquitin fitness landscapes under different chemical stresses in a classroom setting. *eLife*, 5:e15802, April 2016. ISSN 2050-084X. doi: 10.7554/eLife.15802. URL <http://elifesciences.org/content/5/e15802v1>. 5
- [29] David Mavor, Kyle Barlow, Daniel Asarnow, Yuliya Birman, Derek Britain, Weilin Chen, Evan M. Green, Lillian R. Kenner, Bruk Mensa, Leanna S. Morinishi, Charlotte A. Nelson, Erin M. Poss, Pooja Suresh, Ruilin Tian, Taylor Arhar, Beatrice E. Ary, David P. Bauer, Ian D. Bergman, Rachel M. Brunetti, Cynthia M. Chio, Shizhong A. Dai, Miles S. Dickinson, Susanna Elledge, Nathan L. Hendel, Cole V. M. Helsell, Emily Kang, Nadja Kern, Matvei S. Khoroshkin, Lisa L. Kirkemo, Greyson R. Lewis, Kevin Lou, Wesley M. Marin, Alison M. Maxwell, Peter F. McTigue, Douglas Meyers-Turnbull, Tamas L. Nagy, Andrew M. Natale, Keely Oltion, Sergei Pourmal, Gabriel K. Reder, Nicholas J. Rettko, Peter J. Rohweder, Daniel M. C. Schwarz, Sophia K. Tan, Paul V. Thomas, Ryan W. Tibble, Mary K. Tsai, Jason P. Town, Fatima S. Ugur, Douglas R. Wassarman, Alexander M. Wolff, Taia S. Wu, Derek Bogdanoff, Jennifer Li, Kurt S. Thorn, Shane Thomas O’Conchúir, Danielle L. Swaney, Eric D. Chow, Hiten D. Madhani, Sy Redding, Daniel N. A. Bolon, Tanja Kortemme,

- Joseph L. DeRisi, Martin Kampmann, and James S. Fraser. Extending Chemical Perturbations Of The Ubiquitin Fitness Landscape In A Classroom Setting. *bioRxiv*, May 2017. doi: 10.1101/139352. URL <http://biorxiv.org/content/early/2017/05/17/139352.abstract>. 5
- [30] Andrew Leaver-Fay, Michael Tyka, Steven M. Lewis, Oliver F. Lange, James Thompson, Ron Jacak, Kristian W. Kaufman, P. Douglas Renfrew, Colin A. Smith, Will Sheffler, Ian W. Davis, Seth Cooper, Adrien Treuille, Daniel J. Mandell, Florian Richter, Yih-En Andrew Ban, Sarel J. Fleishman, Jacob E. Corn, David E. Kim, Sergey Lyskov, Monica Berrondo, Stuart Mentzer, Zoran Popović, James J. Havranek, John Karanicolas, Rhiju Das, Jens Meiler, Tanja Kortemme, Jeffrey J. Gray, Brian Kuhlman, David Baker, and Philip Bradley. Rosetta3: An Object-Oriented Software Suite for the Simulation and Design of Macromolecules. *Methods in Enzymology*, 487:545–574, 2011. doi: 10.1016/B978-0-12-381270-4.00019-6. URL <http://www.sciencedirect.com/science/article/pii/B9780123812704000196>. 7
- [31] Benjamin Webb and Andrej Sali. Comparative Protein Structure Modeling Using MODELLER. *Current Protocols in Bioinformatics*, 2002. doi: 10.1002/0471250953.bi0506s47. URL <http://onlinelibrary.wiley.com/doi/10.1002/0471250953.bi0506s47/abstract>.
- [32] David A. Case, Thomas E. Cheatham, Tom Darden, Holger Gohlke, Ray Luo, Kenneth M. Merz, Alexey Onufriev, Carlos Simmerling, Bing Wang, and Robert J. Woods. The Amber biomolecular simulation programs. *Journal of Computational Chemistry*, 26(16):1668–1688, December 2005. ISSN 1096-987X. doi: 10.1002/jcc.20290. URL <http://onlinelibrary.wiley.com/doi/10.1002/jcc.20290/abstract>.
- [33] William L. Jorgensen, David S. Maxwell, and Julian Tirado-Rives. Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *Journal of the American Chemical Society*, 118

- (45):11225–11236, January 1996. ISSN 0002-7863. doi: 10.1021/ja9621760. URL <http://dx.doi.org/10.1021/ja9621760>. 7
- [34] M. D. Shaji Kumar, K. Abdulla Bava, M. Michael Gromiha, Ponraj Prabakaran, Koji Kitajima, Hatsuho Uedaira, and Akinori Sarai. ProTherm and ProNIT: thermodynamic databases for proteins and protein–nucleic acid interactions. *Nucleic Acids Research*, 34(suppl 1):D204–D206, January 2006. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkj103. URL [http://nar.oxfordjournals.org/content/34/suppl\\_1/D204](http://nar.oxfordjournals.org/content/34/suppl_1/D204). 7, 12, 30
- [35] A. Patrícia Bento, Anna Gaulton, Anne Hersey, Louisa J. Bellis, Jon Chambers, Mark Davies, Felix A. Krüger, Yvonne Light, Lora Mak, Shaun McGlinchey, Michal Nowotka, George Papadatos, Rita Santos, and John P. Overington. The ChEMBL bioactivity database: an update. *Nucleic Acids Research*, 42(D1):D1083–D1090, January 2014. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkt1031. URL <http://nar.oxfordjournals.org/content/42/D1/D1083>.
- [36] Iain H. Moal and Juan Fernández-Recio. SKEMPI: a Structural Kinetic and Energetic database of Mutant Protein Interactions and its use in empirical models. *Bioinformatics*, 28(20):2600–2607, October 2012. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/bts489. URL <http://bioinformatics.oxfordjournals.org/content/28/20/2600>. 36
- [37] Julian Mintseris, Kevin Wiehe, Brian Pierce, Robert Anderson, Rong Chen, Joël Janin, and Zhiping Weng. Protein–protein docking benchmark 2.0: An update. *Proteins: Structure, Function, and Bioinformatics*, 60(2):214–216, August 2005. ISSN 1097-0134. doi: 10.1002/prot.20560. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.20560/abstract>. 7, 8
- [38] John Moult, Krzysztof Fidelis, Andriy Kryshtafovych, Torsten Schwede, and Anna

- Tramontano. Critical assessment of methods of protein structure prediction (CASP) — round x. *Proteins: Structure, Function, and Bioinformatics*, 82:1–6, February 2014. ISSN 1097-0134. doi: 10.1002/prot.24452. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.24452/abstract>. 8
- [39] Joël Janin and Shoshana Wodak. The Third CAPRI Assessment Meeting Toronto, Canada, April 20–21, 2007. *Structure*, 15(7):755–759, July 2007. ISSN 0969-2126. doi: 10.1016/j.str.2007.06.007. URL <http://www.sciencedirect.com/science/article/pii/S0969212607002134>. 8
- [40] Howook Hwang, Thom Vreven, Joël Janin, and Zhiping Weng. Protein–protein docking benchmark version 4.0. *Proteins: Structure, Function, and Bioinformatics*, 78(15): 3111–3114, November 2010. ISSN 1097-0134. doi: 10.1002/prot.22830. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.22830/abstract>. 8
- [41] Gavin E. Crooks, Gary Hon, John-Marc Chandonia, and Steven E. Brenner. WebLogo: A Sequence Logo Generator. *Genome Research*, 14(6):1188–1190, June 2004. ISSN 1088-9051, 1549-5469. doi: 10.1101/gr.849004. URL <http://genome.cshlp.org/content/14/6/1188>. 10
- [42] Raphael Guerois, Jens Erik Nielsen, and Luis Serrano. Predicting Changes in the Stability of Proteins and Protein Complexes: A Study of More Than 1000 Mutations. *Journal of Molecular Biology*, 320(2):369–387, July 2002. ISSN 0022-2836. doi: 10.1016/S0022-2836(02)00442-4. URL <http://www.sciencedirect.com/science/article/pii/S0022283602004424>. 12, 34
- [43] Vladimir Potapov, Mati Cohen, and Gideon Schreiber. Assessing computational methods for predicting protein stability upon mutation: good on average but not in the details. *Protein Engineering Design and Selection*, 22(9):553–560, Septem-

- ber 2009. ISSN 1741-0126, 1741-0134. doi: 10.1093/protein/gzp030. URL <http://peds.oxfordjournals.org/content/22/9/553>. 12
- [44] Elizabeth H. Kellogg, Andrew Leaver-Fay, and David Baker. Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins: Structure, Function, and Bioinformatics*, 79(3):830–838, March 2011. ISSN 1097-0134. doi: 10.1002/prot.22921. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.22921/abstract>. 12, 13, 14, 34, 39
- [45] Alexander Benedix, Caroline M. Becker, Bert L. de Groot, Amedeo Caflisch, and Rainer A. Böckmann. Predicting free energy changes using structural ensembles. *Nature Methods*, 6(1):3–4, January 2009. ISSN 1548-7091. doi: 10.1038/nmeth0109-3. URL <http://www.nature.com/nmeth/journal/v6/n1/full/nmeth0109-3.html>. 12, 35, 47
- [46] Matthew J. O’Meara, Andrew Leaver-Fay, Michael D. Tyka, Amelie Stein, Kevin Houlihan, Frank DiMaio, Philip Bradley, Tanja Kortemme, David Baker, Jack Snoeyink, and Brian Kuhlman. Combined Covalent-Electrostatic Model of Hydrogen Bonding Improves Structure Prediction with Rosetta. *Journal of Chemical Theory and Computation*, 11(2):609–622, February 2015. ISSN 1549-9618. doi: 10.1021/ct500864r. URL <http://dx.doi.org/10.1021/ct500864r>. 14, 21, 27
- [47] T. Clackson and J. A. Wells. A hot spot of binding energy in a hormone-receptor interface. *Science*, 267(5196):383–386, January 1995. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.7529940. URL <http://www.sciencemag.org/content/267/5196/383>. 15
- [48] Andrew A Bogan and Kurt S Thorn. Anatomy of hot spots in protein interfaces1. *Journal of Molecular Biology*, 280(1):1–9, July 1998. ISSN 0022-2836. doi:

- 10.1006/jmbi.1998.1843. URL <http://www.sciencedirect.com/science/article/pii/S0022283698918435>. 15
- [49] Steven J. Darnell, Laura LeGault, and Julie C. Mitchell. KFC Server: interactive forecasting of protein interaction hot spots. *Nucleic Acids Research*, 36(suppl 2):W265–W269, July 2008. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkn346. URL [http://nar.oxfordjournals.org/content/36/suppl\\_2/W265](http://nar.oxfordjournals.org/content/36/suppl_2/W265).
- [50] Xiaolei Zhu and Julie C. Mitchell. KFC2: A knowledge-based hot spot prediction method based on interface solvation, atomic density, and plasticity features. *Proteins: Structure, Function, and Bioinformatics*, 79(9):2671–2683, September 2011. ISSN 1097-0134. doi: 10.1002/prot.23094. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.23094/abstract>. 15
- [51] Tanja Kortemme and David Baker. A simple physical model for binding energy hot spots in protein–protein complexes. *Proceedings of the National Academy of Sciences*, 99(22):14116–14121, October 2002. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.202485799. URL <http://www.pnas.org/content/99/22/14116>. 15, 16, 34
- [52] Tanja Kortemme, David E. Kim, and David Baker. Computational Alanine Scanning of Protein-Protein Interfaces. *Science Signaling*, 2004(219):pl2–pl2, February 2004. doi: 10.1126/stke.2192004pl2. URL <http://stke.sciencemag.org/content/2004/219/p12>. 15, 16, 34
- [53] Noah Ollikainen and Tanja Kortemme. Computational Protein Design Quantifies Structural Constraints on Amino Acid Covariation. *PLoS Comput Biol*, 9(11):e1003313, November 2013. doi: 10.1371/journal.pcbi.1003313. URL <http://dx.doi.org/10.1371/journal.pcbi.1003313>. 17, 18, 20, 23
- [54] Noah Ollikainen, Colin A. Smith, James S. Fraser, and Tanja Kortemme. Chapter Four - Flexible Backbone Sampling Methods to Model and Design Protein Alterna-



- tive Conformations. *Methods in Enzymology*, 523:61–85, 2013. doi: 10.1016/B978-0-12-394292-0.00004-7. URL <http://www.sciencedirect.com/science/article/pii/B9780123942920000047>. 17
- [55] Nikolay V. Dokholyan and Eugene I. Shakhnovich. Understanding hierarchical protein evolution from first principles. *Journal of Molecular Biology*, 312(1):289–307, September 2001. ISSN 0022-2836. doi: 10.1006/jmbi.2001.4949. URL <http://www.sciencedirect.com/science/article/pii/S0022283601949496>. 17
- [56] Brian Kuhlman and David Baker. Native protein sequences are close to optimal for their structures. *Proceedings of the National Academy of Sciences of the United States of America*, 97(19):10383–10388, September 2000. ISSN 0027-8424. doi: 10.1073/pnas.97.19.10383. URL <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC27033/>. 17, 19
- [57] Robert D. Finn, Alex Bateman, Jody Clements, Penelope Coghill, Ruth Y. Eberhardt, Sean R. Eddy, Andreas Heger, Kirstie Hetherington, Liisa Holm, Jaina Mistry, Erik L. L. Sonnhammer, John Tate, and Marco Punta. Pfam: the protein families database. *Nucleic Acids Research*, 42(D1):D222–D230, January 2014. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkt1223. URL <http://nar.oxfordjournals.org/content/42/D1/D222>. 18
- [58] Russell J. Dickson, Lindi M. Wahl, Andrew D. Fernandes, and Gregory B. Gloor. Identifying and Seeing beyond Multiple Sequence Alignment Errors Using Intra-Molecular Protein Covariation. *PLoS ONE*, 5(6):e11082, June 2010. doi: 10.1371/journal.pone.0011082. URL <http://dx.doi.org/10.1371/journal.pone.0011082>. 18, 22
- [59] Golan Yona and Michael Levitt. Within the twilight zone: a sensitive profile-profile comparison tool based on information theory<sup>1</sup>. *Journal of Molecular Biology*, 315(5):1257–1275, February 2002. ISSN 0022-2836. doi: 10.1006/jmbi.2001.5293. URL <http://www.sciencedirect.com/science/article/pii/S0022283601952933>. 20

- [60] Andrew Leaver-Fay, Matthew J. O'Meara, Mike Tyka, Ron Jacak, Yifan Song, Elizabeth H. Kellogg, James Thompson, Ian W. Davis, Roland A. Pache, Sergey Lyskov, Jeffrey J. Gray, Tanja Kortemme, Jane S. Richardson, James J. Havranek, Jack Snoeyink, David Baker, and Brian Kuhlman. Chapter Six - Scientific Benchmarks for Guiding Macromolecular Energy Function Improvement. *Methods in Enzymology*, 523:109–143, 2013. doi: 10.1016/B978-0-12-394292-0.00006-0. URL <http://www.sciencedirect.com/science/article/pii/B9780123942920000060>. 21, 27, 36
- [61] S. D. Dunn, L. M. Wahl, and G. B. Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, February 2008. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btm604. URL <http://bioinformatics.oxfordjournals.org/content/24/3/333>. 22
- [62] Colin A. Smith and Tanja Kortemme. Backrub-Like Backbone Simulation Recapitulates Natural Protein Conformational Variability and Improves Mutant Side-Chain Prediction. *Journal of Molecular Biology*, 380(4):742–756, July 2008. ISSN 0022-2836. doi: 10.1016/j.jmb.2008.05.023. URL <http://www.sciencedirect.com/science/article/pii/S0022283608005779>. 23, 24
- [63] Raffi Tonikian, Yingnan Zhang, Stephen L Sazinsky, Bridget Currell, Jung-Hua Yeh, Boris Reva, Heike A Held, Brent A Appleton, Marie Evangelista, Yan Wu, Xiaofeng Xin, Andrew C Chan, Somasekar Seshagiri, Laurence A Lasky, Chris Sander, Charles Boone, Gary D Bader, and Sachdev S Sidhu. A Specificity Map for the PDZ Domain Family. *PLoS Biol*, 6(9):e239, September 2008. doi: 10.1371/journal.pbio.0060239. URL <http://dx.doi.org/10.1371/journal.pbio.0060239>. 24, 26
- [64] Andreas Ernst, Stephen L. Sazinsky, Shirley Hui, Bridget Currell, Moyez Dharsee, Somasekar Seshagiri, Gary D. Bader, and Sachdev S. Sidhu. Rapid Evolution of

- Functional Complexity in a Domain Family. *Science Signaling*, 2(87):ra50–ra50, September 2009. ISSN 1945-0877, 1937-9145. doi: 10.1126/scisignal.2000416. URL <http://stke.sciencemag.org/content/2/87/ra50>. 24
- [65] Colin A. Smith and Tanja Kortemme. Predicting the Tolerated Sequences for Proteins and Protein Interfaces Using RosettaBackrub Flexible Backbone Design. *PLoS ONE*, 6(7):e20451, July 2011. doi: 10.1371/journal.pone.0020451. URL <http://dx.doi.org/10.1371/journal.pone.0020451>. 24, 25, 35
- [66] Colin A. Smith and Tanja Kortemme. Structure-Based Prediction of the Peptide Sequence Space Recognized by Natural and Synthetic PDZ Domains. *Journal of Molecular Biology*, 402(2):460–474, September 2010. ISSN 0022-2836. doi: 10.1016/j.jmb.2010.07.032. URL <http://www.sciencedirect.com/science/article/pii/S0022283610007850>. 24, 25, 35
- [67] Amelie Stein and Tanja Kortemme. Improvements to Robotics-Inspired Conformational Sampling in Rosetta. *PLoS ONE*, 8(5):e63090, May 2013. doi: 10.1371/journal.pone.0063090. URL <http://dx.doi.org/10.1371/journal.pone.0063090>. 26, 27
- [68] Benjamin D. Sellers, Kai Zhu, Suwen Zhao, Richard A. Friesner, and Matthew P. Jacobson. Toward better refinement of comparative models: Predicting loops in inexact environments. *Proteins: Structure, Function, and Bioinformatics*, 72(3):959–971, August 2008. ISSN 1097-0134. doi: 10.1002/prot.21990. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.21990/abstract>. 26
- [69] Chu Wang, Philip Bradley, and David Baker. Protein–Protein Docking with Backbone Flexibility. *Journal of Molecular Biology*, 373(2):503–519, October 2007. ISSN 0022-2836. doi: 10.1016/j.jmb.2007.07.050. URL <http://www.sciencedirect.com/science/article/pii/S0022283607010030>. 26
- [70] András Fiser, Richard Kinh Gian Do, and Andrej Šali. Modeling of loops in protein

- structures. *Protein Science*, 9(9):1753–1773, January 2000. ISSN 1469-896X. doi: 10.1110/ps.9.9.1753. URL <http://onlinelibrary.wiley.com/doi/10.1110/ps.9.9.1753/abstract>.
- [71] Carol A. Rohl, Charlie E.M. Strauss, Dylan Chivian, and David Baker. Modeling structurally variable regions in homologous proteins with rosetta. *Proteins: Structure, Function, and Bioinformatics*, 55(3):656–677, May 2004. ISSN 1097-0134. doi: 10.1002/prot.10629. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.10629/abstract>.
- [72] Kai Zhu, David L. Pincus, Suwen Zhao, and Richard A. Friesner. Long loop prediction using the protein local optimization program. *Proteins: Structure, Function, and Bioinformatics*, 65(2):438–452, November 2006. ISSN 1097-0134. doi: 10.1002/prot.21040. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.21040/abstract>.
- [73] Matthew P. Jacobson, David L. Pincus, Chaya S. Rapp, Tyler J.F. Day, Barry Honig, David E. Shaw, and Richard A. Friesner. A hierarchical approach to all-atom protein loop prediction. *Proteins: Structure, Function, and Bioinformatics*, 55(2):351–367, May 2004. ISSN 1097-0134. doi: 10.1002/prot.10613. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.10613/abstract>. 26
- [74] Maxim V. Shapovalov and Roland L. Dunbrack Jr. A Smoothed Backbone-Dependent Rotamer Library for Proteins Derived from Adaptive Kernel Density Estimates and Regressions. *Structure*, 19(6):844–858, June 2011. ISSN 0969-2126. doi: 10.1016/j.str.2011.03.019. URL <http://www.sciencedirect.com/science/article/pii/S0969212611001444>. 26, 36, 37
- [75] Suwen Zhao, Kai Zhu, Jianing Li, and Richard A. Friesner. Progress in super long loop prediction. *Proteins: Structure, Function, and Bioinformatics*, 79(10):

- 2920–2935, October 2011. ISSN 1097-0134. doi: 10.1002/prot.23129. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.23129/abstract>. 26
- [76] Adrian A. Canutescu and Roland L. Dunbrack. Cyclic coordinate descent: A robotics algorithm for protein loop closure. *Protein Science*, 12(5):963–972, May 2003. ISSN 09618368, 1469896X. doi: 10.1110/ps.0242703. URL [http://www.proteinscience.org/details/journalArticle/112851/Cyclic\\_coordinate\\_descent\\_A\\_robotics\\_algorithm\\_for\\_protein\\_loop\\_closure.html](http://www.proteinscience.org/details/journalArticle/112851/Cyclic_coordinate_descent_A_robotics_algorithm_for_protein_loop_closure.html). 26
- [77] Evangelos A. Coutsias, Chaok Seok, Matthew P. Jacobson, and Ken A. Dill. A kinematic view of loop closure. *Journal of Computational Chemistry*, 25(4):510–528, March 2004. ISSN 1096-987X. doi: 10.1002/jcc.10416. URL <http://onlinelibrary.wiley.com/doi/10.1002/jcc.10416/abstract>. 26
- [78] Bagley D, Fulgham B, and Gouy I. The Computer Language Benchmarks Game, 2004. URL <http://benchmarksgame.alioth.debian.org/>. 30, 31
- [79] Calpini A. The Great Win32 Computer Language Shootout, 2003. URL <http://dada.perl.it/shootout/>. 30, 31
- [80] Harry C. Jubb, Arun P. Pandurangan, Meghan A. Turner, Bernardo Ochoa-Montaño, Tom L. Blundell, and David B. Ascher. Mutations at protein-protein interfaces: Small changes over big surfaces have large impacts on human health. *Progress in Biophysics and Molecular Biology*, 128:3–13, September 2017. ISSN 0079-6107. doi: 10.1016/j.pbiomolbio.2016.10.002. URL <http://www.sciencedirect.com/science/article/pii/S0079610716300311>. 33
- [81] Hetunandan Kamisetty, Arvind Ramanathan, Chris Bailey-Kellogg, and Christopher James Langmead. Accounting for conformational entropy in predicting binding

- free energies of protein-protein interactions. *Proteins: Structure, Function, and Bioinformatics*, 79(2):444–462, February 2011. ISSN 1097-0134. doi: 10.1002/prot.22894. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.22894/abstract>. 34
- [82] Yves Dehouck, Jean Marc Kwasigroch, Marianne Rooman, and Dimitri Gilis. BeAt-MuSiC: prediction of changes in protein–protein binding affinity on mutations. *Nucleic Acids Research*, 41(Web Server issue):W333–W339, July 2013. ISSN 0305-1048. doi: 10.1093/nar/gkt450. URL <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3692068/>. 34
- [83] Iain H. Moal and Juan Fernandez-Recio. Intermolecular Contact Potentials for Protein–Protein Interactions Extracted from Binding Free Energy Changes upon Mutation. *Journal of Chemical Theory and Computation*, 9(8):3715–3727, August 2013. ISSN 1549-9618. doi: 10.1021/ct400295z. URL <http://dx.doi.org/10.1021/ct400295z>.
- [84] Anna Vangone and Alexandre MJJ Bonvin. Contacts-based prediction of binding affinity in protein–protein complexes. *eLife*, 4:e07454, July 2015. ISSN 2050-084X. doi: 10.7554/eLife.07454. URL <https://elifesciences.org/articles/07454>. 34
- [85] Minghui Li, Marharyta Petukh, Emil Alexov, and Anna R. Panchenko. Predicting the Impact of Missense Mutations on Protein–Protein Binding Affinity. *Journal of Chemical Theory and Computation*, 10(4):1770–1780, April 2014. ISSN 1549-9618. doi: 10.1021/ct401022c. URL <http://dx.doi.org/10.1021/ct401022c>. 34
- [86] Douglas E. V. Pires, David B. Ascher, and Tom L. Blundell. mCSM: predicting the effects of mutations in proteins using graph-based signatures. *Bioinformatics*, 30(3):335–342, February 2014. ISSN 1367-4803. doi: 10.1093/bioinformatics/btt691. URL <https://academic.oup.com/bioinformatics/article/30/3/335/228906/mCSM-predicting-the-effects-of-mutations-in>. 34

- [87] Daniel F. A. R. Dourado and Samuel Coulbourn Flores. A multiscale approach to predicting affinity changes in protein–protein interfaces. *Proteins: Structure, Function, and Bioinformatics*, 82(10):2681–2690, October 2014. ISSN 1097-0134. doi: 10.1002/prot.24634. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.24634/abstract>. 34, 36, 39
- [88] Kristian W. Kaufmann, Gordon H. Lemmon, Samuel L. DeLuca, Jonathan H. Sheehan, and Jens Meiler. Practically Useful: What the Rosetta Protein Modeling Suite Can Do for You. *Biochemistry*, 49(14):2987–2998, 2010. ISSN 0006-2960. doi: 10.1021/bi902153g. URL <http://dx.doi.org/10.1021/bi902153g>. 34
- [89] Martin J Boulanger, Alexander J Bankovich, Tanja Kortemme, David Baker, and K. Christopher Garcia. Convergent Mechanisms for Recognition of Divergent Cytokines by the Shared Signaling Receptor gp130. *Molecular Cell*, 12(3):577–589, September 2003. ISSN 1097-2765. doi: 10.1016/S1097-2765(03)00365-4. URL <http://www.sciencedirect.com/science/article/pii/S1097276503003654>. 34
- [90] Benjamin J. McFarland, Tanja Kortemme, Shuyuan F. Yu, David Baker, and Roland K. Strong. Symmetry Recognizing Asymmetry. *Structure*, 11(4):411–422, April 2003. ISSN 0969-2126. doi: 10.1016/S0969-2126(03)00047-9. URL <http://www.sciencedirect.com/science/article/pii/S0969212603000479>. 34
- [91] Deanne W. Sammond, Ziad M. Eletr, Carrie Purbeck, Randall J. Kimple, David P. Siderovski, and Brian Kuhlman. Structure-based Protocol for Identifying Mutations that Enhance Protein–Protein Binding Affinities. *Journal of Molecular Biology*, 371(5):1392–1404, August 2007. ISSN 0022-2836. doi: 10.1016/j.jmb.2007.05.096. URL <http://www.sciencedirect.com/science/article/pii/S0022283607007668>. 34
- [92] Gang Song, Greg A. Lazar, Tanja Kortemme, Motomu Shimaoka, John R. Desjarlais, David Baker, and Timothy A. Springer. Rational Design of Intercellular Adhe-

- sion Molecule-1 (ICAM-1) Variants for Antagonizing Integrin Lymphocyte Function-associated Antigen-1-dependent Adhesion. *Journal of Biological Chemistry*, 281(8): 5042–5049, February 2006. ISSN 0021-9258, 1083-351X. doi: 10.1074/jbc.M510454200. URL <http://www.jbc.org/content/281/8/5042>. 34
- [93] Gregory T. Kapp, Sen Liu, Amelie Stein, Derek T. Wong, Attila Reményi, Brian J. Yeh, James S. Fraser, Jack Taunton, Wendell A. Lim, and Tanja Kortemme. Control of protein signaling using a computationally designed GTPase/GEF orthogonal pair. *Proceedings of the National Academy of Sciences*, 109(14):5277–5282, April 2012. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1114487109. URL <http://www.pnas.org/content/109/14/5277>. 34, 35
- [94] Brett S. Chevalier, Tanja Kortemme, Meggen S. Chadsey, David Baker, Raymond J. Monnat, and Barry L. Stoddard. Design, Activity, and Structure of a Highly Specific Artificial Endonuclease. *Molecular Cell*, 10(4):895–905, October 2002. ISSN 1097-2765. doi: 10.1016/S1097-2765(02)00690-1. URL <http://www.sciencedirect.com/science/article/pii/S1097276502006901>. 34
- [95] Sarel J. Fleishman, Timothy A. Whitehead, Damian C. Ekiert, Cyrille Dreyfus, Jacob E. Corn, Eva-Maria Strauch, Ian A. Wilson, and David Baker. Computational Design of Proteins Targeting the Conserved Stem Region of Influenza Hemagglutinin. *Science*, 332(6031):816–821, May 2011. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1202617. URL <http://science.sciencemag.org/content/332/6031/816>. 34
- [96] B. C. Cunningham and J. A. Wells. High-resolution epitope mapping of hGH-receptor interactions by alanine-scanning mutagenesis. *Science*, 244(4908):1081–1085, June 1989. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.2471267. URL <http://www.sciencemag.org.ucsf.idm.oclc.org/content/244/4908/1081>. 34
- [97] Mitsugu Araki, Narutoshi Kamiya, Miwa Sato, Masahiko Nakatsui, Takatsugu Hi-



rokawa, and Yasushi Okuno. The Effect of Conformational Flexibility on Binding Free Energy Estimation between Kinases and Their Inhibitors. *Journal of Chemical Information and Modeling*, 56(12):2445–2456, December 2016. ISSN 1549-9596. doi: 10.1021/acs.jcim.6b00398. URL <http://dx.doi.org/10.1021/acs.jcim.6b00398>. 35

[98] Elisabeth L. Humphris and Tanja Kortemme. Prediction of Protein-Protein Interface Sequence Diversity Using Flexible Backbone Computational Protein Design. *Structure*, 16(12):1777–1788, December 2008. ISSN 0969-2126. doi: 10.1016/j.str.2008.09.012. URL <http://www.sciencedirect.com/science/article/pii/S0969212608003869>. 35

[99] Christian D. Schenkelberg and Christopher Bystroff. Protein backbone ensemble generation explores the local structural space of unseen natural homologs. *Bioinformatics*, 32(10):1454–1461, May 2016. ISSN 1367-4803. doi: 10.1093/bioinformatics/btw001. URL <https://academic.oup.com/bioinformatics/article/32/10/1454/1742694/Protein-backbone-ensemble-generation-explores-the>. 35

[100] Noah Ollikainen, René M. de Jong, and Tanja Kortemme. Coupling Protein Side-Chain and Backbone Flexibility Improves the Re-design of Protein-Ligand Specificity. *PLOS Comput Biol*, 11(9):e1004335, September 2015. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1004335. URL <http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004335>. 35

[101] James A. Davey and Roberto A. Chica. Improving the accuracy of protein stability predictions with multistate design using a variety of backbone ensembles. *Proteins: Structure, Function, and Bioinformatics*, 82(5):771–784, May 2014. ISSN 1097-0134. doi: 10.1002/prot.24457. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.24457/abstract>. 35

- [102] Cunliang Geng, Anna Vangone, and Alexandre M. J. J. Bonvin. Exploring the interplay between experimental methods and the performance of predictors of binding affinity change upon mutations in protein complexes. *Protein Engineering, Design and Selection*, 29(8):291–299, August 2016. ISSN 1741-0126. doi: 10.1093/protein/gzw020. URL <https://academic.oup.com/peds/article/29/8/291/2462288/Exploring-the-interplay-between-experimental>. 36
- [103] James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M. Deane. SAbDab: the structural antibody database. *Nucleic Acids Research*, 42(D1):D1140–D1146, January 2014. ISSN 0305-1048. doi: 10.1093/nar/gkt1043. URL <https://academic.oup.com/nar/article/42/D1/D1140/1044118/SAbDab-the-structural-antibody-database>. 36
- [104] Sarel J. Fleishman, Andrew Leaver-Fay, Jacob E. Corn, Eva-Maria Strauch, Sagar D. Khare, Nobuyasu Koga, Justin Ashworth, Paul Murphy, Florian Richter, Gordon Lemmon, Jens Meiler, and David Baker. RosettaScripts: A Scripting Language Interface to the Rosetta Macromolecular Modeling Suite. *PLOS ONE*, 6(6):e20161, June 2011. ISSN 1932-6203. doi: 10.1371/journal.pone.0020161. URL <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0020161>. 36
- [105] Yifan Song, Michael Tyka, Andrew Leaver-Fay, James Thompson, and David Baker. Structure-guided forcefield optimization. *Proteins: Structure, Function, and Bioinformatics*, 79(6):1898–1909, June 2011. ISSN 1097-0134. doi: 10.1002/prot.23013. URL <http://onlinelibrary.wiley.com/doi/10.1002/prot.23013/abstract>. 36
- [106] Andrew Leaver-Fay, Ron Jacak, P. Benjamin Stranges, and Brian Kuhlman. A Generic Program for Multistate Protein Design. *PLOS ONE*, 6(7):e20937, July 2011. ISSN 1932-6203. doi: 10.1371/journal.pone.0020937. URL <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0020937>. 37

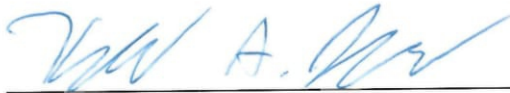
- [107] Simon N. Wood. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(1):3–36, January 2011. ISSN 1467-9868. doi: 10.1111/j.1467-9868.2010.00749.x. URL <http://onlinelibrary.wiley.com/doi/10.1111/j.1467-9868.2010.00749.x/abstract>. 46
- [108] Andrea Zen, Cristian Micheletti, Ozlem Keskin, and Ruth Nussinov. Comparing interfacial dynamics in protein-protein complexes: an elastic network approach. *BMC Structural Biology*, 10:26, August 2010. ISSN 1472-6807. doi: 10.1186/1472-6807-10-26. URL <https://doi.org/10.1186/1472-6807-10-26>. 46
- [109] Rebecca F. Alford, Andrew Leaver-Fay, Jeli azko R. Jeli azkov, Matthew J. O’Meara, Frank P. DiMaio, Hahnbeom Park, Maxim V. Shapovalov, P. Douglas Renfrew, Vikram K. Mulligan, Kalli Kappel, Jason W. Labonte, Michael S. Pacella, Richard Bonneau, Philip Bradley, Roland L. Dunbrack, Rhiju Das, David Baker, Brian Kuhlman, Tanja Kortemme, and Jeffrey J. Gray. The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design. *Journal of Chemical Theory and Computation*, 13(6):3031–3048, June 2017. ISSN 1549-9618. doi: 10.1021/acs.jctc.7b00125. URL <http://dx.doi.org/10.1021/acs.jctc.7b00125>. 47

**Publishing Agreement**

*It is the policy of the University to encourage the distribution of all theses, dissertations, and manuscripts. Copies of all UCSF theses, dissertations, and manuscripts will be routed to the library via the Graduate Division. The library will make all theses, dissertations, and manuscripts accessible to the public and will preserve these to the best of their abilities, in perpetuity.*

**Please sign the following statement:**

*I hereby grant permission to the Graduate Division of the University of California, San Francisco to release copies of my thesis, dissertation, or manuscript to the Campus Library to provide access and preservation, in whole or in part, in perpetuity.*



Author Signature



Date