5-1-2007

Experiment Planning for Protein Structure Elucidation and Site-Directed Protein Recombination

Xiaoduan Ye
Dartmouth College

Experiment Planning for Protein Structure Elucidation and Site-Directed Protein Recombination

A Thesis
Submitted to the Faculty
in partial fulfillment of the requirements for the
degree of
Doctor of Philosophy
in
Computer Science
by

Xiaoduan Ye
DARTMOUTH COLLEGE
Hanover, New Hampshire

May 2007

Examining Committee:

(chair) Chris Bailey-Kellogg

Bruce R. Donald

Peter Winkler

Alan M. Friedman

Charles K. Barlowe, PhD.
Dean of Graduate Studies

Dartmouth Computer Science Technical Report TR2008-614

Abstract

In order to most effectively investigate protein structure and improve protein function, it is necessary to carefully plan appropriate experiments. The combinatorial number of possible experiment plans demands effective criteria and efficient algorithms to choose the one that is in some sense optimal. This thesis addresses experiment planning challenges in two significant applications. The first part of this thesis develops an integrated computational-experimental approach for rapid discrimination of predicted protein structure models by quantifying their consistency with relatively cheap and easy experiments (cross-linking and site-directed mutagenesis followed by stability measurement). In order to obtain the most information from noisy and sparse experimental data, rigorous Bayesian frameworks have been developed to analyze the information content. Efficient algorithms have been developed to choose the most informative, least expensive, and most robust experiments. The effectiveness of this approach has been demonstrated using existing experimental data as well as simulations, and it has been applied to discriminate predicted structure models of the pTfa chaperone protein from bacteriophage lambda.

The second part of this thesis seeks to choose optimal breakpoint locations for protein engineering by site-directed recombination. In order to increase the possibility of obtaining folded and functional hybrids in protein recombination, it is necessary to retain the evolutionary relationships among amino acids that determine protein stability and functionality. A probabilistic hypergraph model has been developed to model these relationships, with edge weights representing their statistical significance derived from database and a pro-

tein family. The effectiveness of this model has been validated by showing its ability to distinguish functional hybrids from non-functional ones in existing experimental data. It has been proved to be NP-hard in general to choose the optimal breakpoint locations for recombination that minimize the total perturbation to these relationships, but exact and approximate algorithms have been developed for a number of important cases.

Acknowledgements

This work would never have been done without the support from many people. It is my pleasure to thank these people here.

It is impossible to overstate my gratitude to my Ph.D. advisor, Prof. Chris Bailey-Kellogg. Without his encouragement, inspiration, sound advice, and good teaching, I would never be able to accomplish this. I was always impressed by his insight in science, his great efforts to explain things clearly and simply, and his patience throughout my whole Ph.D. study and thesis-writing period.

I would like to acknowledge the help of Prof. Alan M. Friedman (Biology, Purdue). I stepped into the area of computational biology without any background of biology. This work can never be done without the guidance and help from Alan. This project is in collaboration with Alan's lab, and I did enjoy everyday we worked together. I would also like to extend my gratitude to other two members in Alan's lab, Dr. Patrick K. O'Neil and Adrienne N. Foster, who conducted cross-linking and mutagenesis experiments in this thesis.

I wish to thank many faculty and student colleagues. I thank Dr. Bruce Craig (statistics, Purdue), Anthony (Tony) K. Yan (CS, Duke), and Michal Gajda (IIMCB, Poland), for stim-

ulating discussion about regression analysis, Bayesian theory, protein structure modeling, and lots of interesting topics.

I am grateful to all my friends from CBK lab. They have been abundantly helpful in many aspects. These lovely people are: Dr. Shobha Potluri, John Thomas, Fei Xiong, Wei Zheng, Jairav Desai, Bornika Ghosh, and Himanshu Chandola.

At last, I would like to thank my wife, Yin (Amy) Yuan. She is certainly the best discovery during my Ph.D. life. Nothing would be worth doing without her.

Contents

List of Tables

List of Figures

1. INTRODUCTION

Proteins are ubiquitous in cells and essential to almost all biological processes. Discovering the amino acid composition, structures and functions of proteins is fundamental for understanding cellular processes, and also supports important applications such as drug design and enzyme design. With the extensive development of genome projects, more and more protein sequences have become available. As of April 2007, more than 500 genome projects, including the human genome project, have been completed, and about 1000 more genome projects are in progress [67]. The latest release (release 10.3) of the Universal Protein Resource (UniProt) [3] contains more than $4,500,000$ protein sequences. While amino acid sequences define proteins, it only is by folding into specific three-dimensional structures that proteins are able to perform their functions. Thus structure determination is essential to gaining a mechanistic understanding of proteins, but unfortunately it is much harder than sequencing. The number of experimentally-determined structures in the current (April 2007) Protein Data Bank [10] counts for only about 1% of the sequences in UniProt. Consequently, new techniques are required to achieve high-throughput protein structure elucidation.

Driven by the overwhelming number of targets available for structure determination and

the difficulties of traditional methods such as x-ray crystallography and NMR, a number of computational approaches have been developed to predict protein structures from sequences. Although the protein sequence space is enormous, the protein fold space is much more restricted. It has been suggested that there are only about 2000 folds existing among naturally-occurring proteins [42] and most of them can be found in the current Protein Data Bank (PDB) [132]. Therefore, it is very likely that a new protein will adopt a fold similar to that of an experimentally determined protein structure, which provides a starting point or template for the new structure. Based on this observation, computational methods such as homology modeling and threading have been developed to predict protein structures from sequences. Another classes of computational methods, *ab initio* methods, have also been developed to predict protein structures from sequence alone, when no template is available.

Computational methods are usually much cheaper and faster than x-ray or NMR and some of them, especially homology modeling, can produce models with an RMSD (Root Mean Square Deviation) of 1-3 Å from the corresponding x-ray/NMR structures [116]. However, it is difficult to distinguish (nearly) native structures from incorrect decoys using energy functions alone [82, 11, 111]. The best model is often among a pool of highly ranked models but not the highest-ranked one. Furthermore, different methods often have different scoring functions and different rankings for the same models. Selecting the (almost) correct one from a given set of predicted protein structural models is a critical step in protein structure determination by computational methods.

With the understanding of protein sequences and structures, it is possible to modify existing proteins or design novel ones. Although nature has produced proteins suitable for

functioning in living organisms, it has not yet explored all possibilities of viable proteins. Protein engineering techniques can improve existing proteins, *e.g.* providing increased stability or modified enzymatic activity. Protein engineering has produced proteins with desired features not observed in nature [4], and has significant impacts in applications such as drug design [44], industrial chemical synthesis [112], and nanotechnology [130]. It is also an important mechanism to gain structural and functional understanding of proteins. For example, site-directed mutagenesis has been widely used to help determine protein structures [84] and study binding properties [47].

Both protein structure determination and protein engineering rely on experiments to investigate structural or functional protein features. Since experiments are usually more expensive and time-consuming than computational methods, it is advantageous to consider the possible outcomes before experiments are conducted, in order to optimize experimental parameters and obtain the most information. However, the combinatorial number of possible plans causes challenging computational problems in choosing the one that is most informative and least expensive for a particular application. Noisy experimental data with sparse information content also places a significant burden on data interpretation, and requires associated planning algorithms to optimize for robustness. Focusing on overcoming such challenges, we develop experiment planning mechanisms for two significant applications, protein structure elucidation (Sec. 1.1) and site-directed recombination (Sec. 1.2).

1.1 Planned RApid eXperimental Investigation of Structure (PRAXIS)

In order to close the gap between protein structure prediction and model selection, we have developed a comprehensive computational-experimental protocol for the high-throughput discrimination of predicted protein structure models (Fig. 1.1). The hypothesis here is that the correct model should be more consistent with the true structure than other decoys, in many aspects such as geometry and thermodynamic stability. We measure these protein properties by relatively cheap and easy experimental tests such as cross-linking and mutagenesis, and confirm predicted protein structural models by quantifying their consistency with the experimental data. We call such experiments "minimalist" experiments since they are relatively cheap and easy, and also because the information gained from such experiments is usually sparse and noisy. This information alone is insufficient to determine protein structures, but it could be sufficient to select the correct model from a set of high quality predicted ones [128, 62]. The consistency between predicted models and experiments allows selection of the model(s) most likely to be correct (relative correctness). The consistency of a number of diverse experiments with a single model provides confidence in its absolute correctness. The challenges in this approach include selecting among the enormous number of possible experiments that could be done and analyzing the results accurately. Emphasizing the importance of experiment planning, we call this approach "PRAXIS" (Planned RApid eXperimental Investigation of Structure). We have developed specific methodology for experiment planning and data interpretation for cross-linking and site-directed mutagenesis followed by stability measurement.

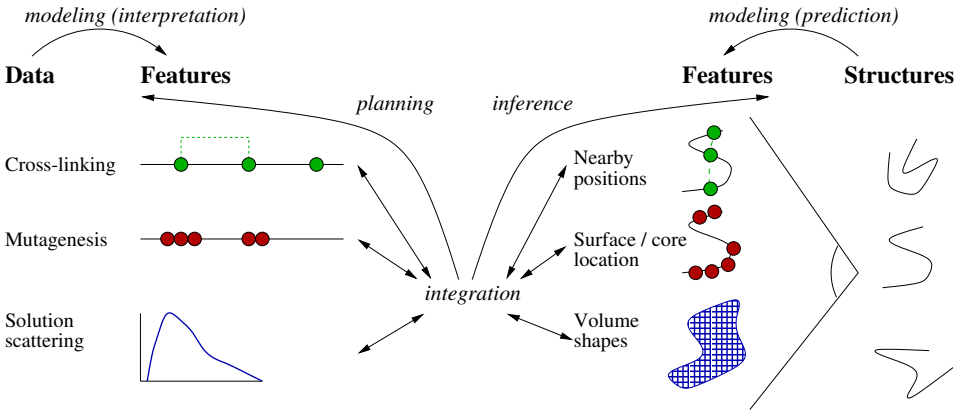


Fig. 1.1: Multimodal PRAXIS. Given predicted structural models, suitable features of the models are extracted and corresponding protein features (*e.g.* residue-residue distance, residue local environment) are tested in wet-lab experiments (*e.g.* cross-linking, mutagenesis, solution scattering). Models are confirmed or selected based on the consistency of their features with experimental data.

Specific sites in protein can be cross-linked (*i.e.* linked by covalent bonds), either by employing residue-specific cross-linker molecules or by disulfide bonding of specifically introduced cysteines. Detection of cross-link formation generally provides only the information that some pairs of residues are closer than a maximal cross-linking distance. Since residue-residue distances can be very different in different predicted models, cross-linking can be used to test which models correctly modeled a specific residue-residue distance. Although this information is approximate, sparse and noisy, it has been previously demonstrated sufficient for discriminating among predicted structural models. In this thesis, we address the essential question of the information content available from a cross-linking experiment, a question required to determine the utility of conducting any particular experiment and optimize experiments accordingly. Our analysis also includes consideration of multiple sources of experimental error such as false positive and false negative identification of cross-links. We addressed these requirements with a probabilistic framework that explicitly accounts for the expected experimental limitations. We also developed associ-

ated algorithms for selecting the most suitable set of experimental parameters (*e.g.* different cross-linkers) and the most informative and least expensive subset of experiments, subject to trade-offs in experimental design.

In addition to cross-linking, we exploit the known relationship between protein structure and thermodynamic stability to investigate protein structure. Several methods are now available for using an atomic model to predict changes in unfolding free energy upon site-directed mutagenesis (*i.e.* substitution of one residue type for another at a specific position), the $\Delta\Delta G^\circ$ values. Similar with residue-residue distance, different predicted models tend to have different thermodynamic properties, and hence different predicted $\Delta\Delta G^\circ$ values. The consistency between predicted and experimentally tested $\Delta\Delta G^\circ$ values allows selecting the correct model from a given set of models. When experiments are planned ahead of time, the experiment planner can select an efficient set of mutations whose stability changes can be most confidently predicted and that differ most greatly between atomic models. The planned mutations are made in a protein expression system. The stability of the expressed and purified mutants is determined and compared to wild-type, yielding experimental $\Delta\Delta G^\circ$ data. We call this combination of planned, site-directed mutation and stability measurement "stability mutagenesis."

Adopting the framework developed for cross-linking, we demonstrated that the information in stability mutagenesis is sufficient for discriminating predicted structural models of different folds using existing experimental data as well as simulations. Then we developed new criteria and corresponding planning algorithms specifically for stability mutagenesis that take full advantage of the information content in continuous $\Delta\Delta G^\circ$ data.

The main contributions of our PRAXIS approach include:

- Developing rigorous probabilistic frameworks to analyze information content in sparse and noisy experimental data (cross-linking and stability mutagenesis) for protein structure elucidation.

- Demonstrating by existing experimental data and simulation that mutagenesis information is sufficient for discriminating protein models of different folds.

- Developing efficient algorithms for choosing the most informative and least expensive experiments to discriminate a given set of protein structure models, both for discrete data (cross-linking) and continuous data (mutagenesis). The algorithms allow experimenters to make explicit trade-off among key properties of practical importance such as information gain, robustness and experimental cost.

- Putting the PRAXIS approach into practice on the pTfa chaperone protein of bacteriophage lambda. Optimal plans for cross-linking and stability mutagenesis have been selected that have been or are being conducted in wet-lab.

1.2 Site-Directed Protein Recombination

Mutagenesis is an effective mechanism to create new protein variants both in nature and in the lab [106, 19, 66]. A single mutation in an active site can change protein function significantly [34]. However, mutagenesis can only yield relatively minor modifications of existing proteins because the chance of obtaining beneficial mutations is very low and it

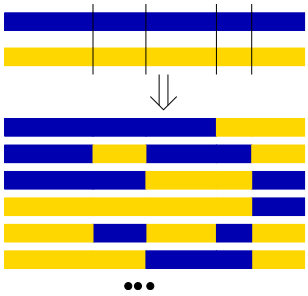


Fig. 1.2: Site-directed recombination experiments mix and match sequential fragments from homologous parents to construct a library of hybrids with the same basic structure but somewhat different sequences and thus different functions.

is hard to accumulate more than a couple. A more efficient way to accumulate beneficial mutations is by recombination of active variants, as occurs naturally during meiosis and has also been demonstrated to be extremely useful in laboratory protein evolution [105, 30, 77].

Protein recombination *in vitro* enables the design of protein variants with favorable properties and novel enzymatic activities, as well as the exploration of the relationships among protein sequence, structure, and function. In this approach, libraries of hybrid proteins are generated either by stochastic enzymatic reactions or intentional selection of breakpoints. Hybrids with unusual properties can either be identified by large-scale genetic screening and selection, or many hybrids can be evaluated individually to determine detailed sequence-function relationships for understanding and/or rational engineering. Both screening/selection and individually evaluated experiments benefit from recombination that preserves the most essential structural and functional features while still allowing variation.

We focus here on site-directed recombination (Fig. 1.2), in which parent genes are recombined at specified breakpoint locations, yielding hybrids in which different sequence fragments (between the breakpoints) can come from different parents. In order to enhance the probability of obtaining folded and functional hybrids, it is necessary to choose break-

point locations wisely to retain the evolutionary relationships among amino acids that determine protein stability and functionality. We developed a probabilistic hypergraph model to represent the evolutionary relationships among amino acids, and a statistical score to evaluate the significance of multi-order amino acid interactions. In support of this model, we developed criteria to evaluate the quality of hybrid libraries by considering the effects of recombination on multi-order amino acid interactions. Intuitively, optimizing the retainment of such relationships after recombination should help identify the best recombinants and thus the best locations for breakpoints. However, there are a combinatoric number of breakpoint location sets, making it difficult to choose the optimal one even with a naïve optimality criterion. We formulate the optimal selection of breakpoint locations as a sequentially-constrained hypergraph partitioning problem, *i.e.* breaking the hypergraph model of protein structure along the backbone. We proved that this problem is NP-hard in general and developed exact and heuristic algorithms when the order of amino acid interactions is limited to three and four, respectively.

Our main contributions for the site-directed protein recombination problem include:

- Developing a probabilistic hypergraph model of evolutionary relationships that generalizes traditional pairwise contact potentials to account for the statistics of multi-residue interactions in protein structures.

- Evaluating the significance of multi-residue interactions in a multiple sequence alignment by analytically calculating their p-values.

- Formulating the breakpoint selection problem in recombination as a sequentially-

constrained hypergraph partitioning problem, proving that it is NP-complete in general, and developing exact and heuristic polynomial-time algorithms for a number of important cases.

- Validating the hypergraph model by showing its ability to distinguish functional hybrids from non-functional one in existing experimental data.

2. RELATED WORK

2.1 Protein Structure Determination

Traditional experimental methods such as x-ray crystallography and Nuclear Magnetic Resonance (NMR) spectroscopy can reveal atomic-detail structures of proteins. In x-ray crystallography [14], proteins crystals are first formed in highly supersaturated solution and then grown under varied solution conditions. Once a high-quality crystal is obtained, it is used to diffract x-rays and create diffraction patterns that can be interpreted to determine the three-dimensional positions of the heavy atoms. The main difficulty of structure determination by x-ray crystallography is in obtaining sufficiently large, high-quality crystals providing sufficient diffraction to determine the three-dimensional structure. Many proteins are hard to crystallize because it is extremely difficult to predict the proper conditions for crystal formation and growth [88]. Among 75104 targets (45391 cloned) provided in phase one of Protein Structure Initiative [95], only 3311 crystallized and only 1307 of these crystals provided sufficient diffraction.

Nuclear Magnetic Resonance spectroscopy [14] can severely restrict the range of possible structures by generating a large number of structural restraints evaluated by distance geometry and molecular dynamics [121, 124]. A protein sample is placed in a strong mag-

netic field and its nuclear spins (an intrinsic property) are excited by radiation. The excited spins emit the absorbed radiation at frequencies determined by the electronic environment around the nuclei. With the technique of Fourier transform NMR (FT-NMR), a range of frequencies can be probed at once, allowing magnetization transfer between nuclei and, thereby, the detection of the nuclear-nuclear interactions, either through-bond or through-space. NMR spectroscopy records data from proteins in solution, rather than in a crystal, and thus enables the study of dynamic phenomena such as protein-protein interaction, reaction kinetics, and protein folding [86, 24, 6]. NMR spectroscopy is limited to relatively small proteins due to the low inherent sensitivity and the high complexity of NMR spectra, although technical advances such as isotopic labeling [101] constantly extend the size limit [96, 92, 129].

Another experimental technique for protein structure determination is Electron Microscopy (EM) [8], with the most popular form known as electron cryo-microscopy (cryo-EM) [33]. Cryo-EM can visualize molecules weighing over 150 kDa at a resolution of 5-15 Å [51, 23]. The information provided by EM alone is generally not sufficient to determine an atomic-detail protein structure, but it can be combined with other techniques such as x-ray crystallography or NMR to produce atomic-detail structures of macromolecular complexes [87]. When high-resolution x-ray/NMR structures of assembly components are not available, EM can also be combined with computational prediction methods (see Sec. 2.2) to produce complex structures at lower resolution [113].

2.2 Protein Structure Prediction

Driven by the overwhelming number of targets available for protein structure determination, computational methods such as homology modeling, threading, and *ab initio* prediction [99, 41, 58, 63] have been developed to compensate the shortcomings of experimental techniques. Homology modeling (or comparative modeling) relies on the identification of one or more previously determined protein structures (called templates) whose sequences are similar to the query sequence [99]. The idea is that protein structures are more conserved than sequences so that the structure of a new sequence can be derived from the structure of another similar sequence. The quality of the homology model depends on both the quality of the template and the similarity of sequences. The inaccuracies in homology modeling come from errors in the sequence alignment, improper template selection, regions of a model constructed without a template, and errors in side chain packing, among others [118]. Higher sequence similarity usually implies more significant structural similarity. Proteins with sequence identity over 30% to a known structure can often be modeled with an accuracy equivalent to a low-resolution X-ray structure [125].

In protein threading, one or more templates of the same fold (*i.e.* sharing the same major secondary structures in the same arrangement and topological connections) of the query sequence are recognized, and then the query sequence is threaded through the backbone structures of these templates. A scoring function, such as an empirical energy function derived from known protein structures [13], is used to evaluate the fitness for each sequence-structure alignment. Since most of protein folds can be found in the current PDB [132], it

is very likely that a new protein will have a similar fold to that of an existing experimentally determined protein structure and hence a threading model can be built. The quality of threaded models depends on the extent of structural similarity more than the degree of sequence similarity [15]. Naturally, protein threading is not very successful when no existing structure is similar to the query protein [122]. However, with the accumulation of new folds by the ongoing structural genomics projects [95], the applicability of threading will continue to expand.

While both homology modeling and threading use experimentally-determined structures as templates, *ab initio* methods try to estimate protein structures from sequences alone based on physical or statistical principles [58, 99]. Although homology modeling and threading are usually more accurate, *ab initio* methods are the only alternative in cases where no useful template is available, and several groups have shown good performance for *ab initio* different methods on some targets [59]. *Ab initio* methods often apply some stochastic approaches requiring significant computational resources to search possible solutions. Although currently limited to small proteins, this limit should be alleviated with the development of new algorithms and computing technologies.

The most objective way to evaluate computational modeling methods is to compare predicted models with the corresponding x-ray or NMR structures. The series of Critical Assessment of Structure Prediction (CASP) experiments [76] is dedicated to this purpose. A "blind prediction" process is adopted by CASP: target information is released to registered prediction groups and models must be submitted before the experimentally determined structures become public; submitted models are evaluated through detailed quantitative

comparisons with experimentally determined x-ray/NMR structures [76].

CASP provides an ideal benchmark for computational modeling methods and has been very helpful in advancing them. However, x-ray or NMR structures may not be available for a large number of targets so that we have to resort to other methods for evaluating predicted models. As we mentioned in Chapter 1, an energy function alone is not sufficient to distinguish decoys from near-native models [82]. An alternative method for distinguishing the (almost) correct model from decoys is to evaluate models by some relatively cheaper experimental tests as we do in this thesis. Several such techniques such as cross-linking [48], mutagenesis [126], and solution scattering [45], have been used to gain low-resolution protein structural information. Although the information provided by these techniques is usually noisy and sparse, it can be sufficient to discriminate correct models from incorrect decoys [128, 127]. In this thesis, we employ cross-linking (Sec. 2.3) and mutagenesis (Sec. 2.4) for model discrimination.

2.3 Protein Structure Elucidation by Cross-linking

The first experimental technique employed in this thesis for protein structure elucidation is cross-linking. Specific sites in the protein(s) are cross-linked, either by employing residue-specific cross-linker molecules, such as the lysine-specific bis-sulfo-succinimidyl suberate (BS^3) [107, 48, 128, 21, 64, 94, 117], or by disulfide bonding of specifically introduced cysteines whose C^β approach within 4.6 Å, with proper geometry, during the experiment [18, 55, 64]. Cross-links are then detected by protein chemical means [48, 109] and/or

mass spectrometry [128, 94, 62, 117], or by alteration in electrophoretic mobility [18, 64].

In these experiments, the cross-linking reaction is determined by the geometric feasibility between pairs of sites and the reactivity and accessibility of individual sites. The difference of these properties among models provides the base of model discrimination. In interpreting cross-linking experiments, models have so far been evaluated solely on the geometric feasibility of observed cross-links [48, 7, 128]. In the simplest case, straight-line distance between cross-link sites [128] has been used. Alternative methods have been proposed for computing lower and upper bounds on the lengths of paths exterior to a protein and thus accessible to a cross-linker without steric clashes [85]. The reactivity of the protein groups cannot be easily extracted from the model, but can be corrected for by measurements of reactivity with monofunctional reagents [78]. Finally, geometric feasibility depends on whether or not the cross-linker can bridge the distance between cross-linked atoms in the model, potentially with consideration for protein dynamics. For example, the cross-linker BS^3 reacts with amino groups, including the N-terminus and the N^{ζ} of `Lys` residues, and forms a bridge of up to 11 Å between such pairs.

Several independent experiments have demonstrated successful application of cross-linking, providing models that correlate with prior or subsequent crystal or NMR structures. Employing Edman sequencing and mass spectroscopy of the cross-links, Haniu *et al.* developed a model of human erythropoietin [48] via lysine-specific cross-linking. Young *et al.* pioneered the use of high-resolution mass spectroscopy alone to correctly discriminate threading models [128, 62]. Cross-linking has also been used to determine quaternary arrangements of proteins [55, 93, 109, 5, 117]. These methods are particularly

valuable for proteins, such as membrane proteins [7, 64], that are inherently resistant to traditional structure determination methods. Large sets of cross-links have also been treated as distance restraints in an alternative distance geometry structure determination protocol to determine the arrangement of transmembrane helices in *lac* permease [103], a case where no models were available beforehand.

Cross-linking by oxidation of introduced dicysteine residues has a number of favorable properties for elucidating protein [7] and complex [55] structure and properties. Since each pair of cysteine substitutions is made and tested directly for cross-linking, independence is assured and error can be reduced since the approach eliminates the possibility of assignment error in Mass Spectrometry. From the experiment planning point of view, disulfide trapping also provides more freedom and thus the most informative set of experiments can be selected in order to discriminate a given set of protein structure models.

2.4 $\Delta\Delta G^\circ$ Prediction for Point Mutations

Mutagenesis is another experimental technique employed in this thesis for protein structure elucidation. The effects of mutation on protein stability, *i.e.* the $\Delta\Delta G^\circ$ values, can be predicted from structural models and measured experimentally. The consistency between predicted and experimental values provides a criterion for ranking models. The success of this approach relies on an accurate and reliable prediction of the $\Delta\Delta G^\circ$ values. The various published $\Delta\Delta G^\circ$ prediction methods report good results in the aggregate, or for a defined subset of mutations, demonstrating their value as $\Delta\Delta G^\circ$ predictors [13, 114, 46, 20, 39].

Bowie *et al.* [13] associated the fitness of amino acid sequences into a known 3D structure with the frequencies of residues according to specific environment classes. The environment of a residue was classified into 18 subclasses according to the area of the residue buried in the protein (*i.e.* inaccessible to solvent), the fraction of side-chain area covered by polar atoms (O and N), and the local secondary structure. This method has been extended to predict the change of free folding energy upon mutation, using more detailed classification of environment classes [114] (thus we refer to this method as "ENV"). A statistical pseudo-potential score was derived from the frequencies of finding different residue types in each environment class. The effect of mutation on protein stability was then predicted by the change of this potential score between a mutant residue and the wild-type residue.

Guerois *et al.* [46] developed an empirical energy function to predict the stability of wild-type proteins and site-directed mutants, and the $\Delta\Delta G^{\circ}$ value was then determined as the difference between them (the FOLD-X method). A number of different energy terms that contribute to protein stability, including van der Waals, solvation, hydrogen bonding, and electrostatics, were taken into account in the energy function and weighted using a large amount of empirical mutagenesis data. FOLD-X achieved a global correlation of 0.83 between the predicted and experimental $\Delta\Delta G^{\circ}$ values for 95% of more than 1000 point mutations, with a standard deviation of 0.81 kcal/mol.

Carter *et al.* [20] developed a four-body likelihood potential to predict the change of protein stability for mutations in the hydrophobic core. Three-dimensional protein structures were tiled with tetrahedra by Delaunay tessellation, where the vertices were the mass centers of amino acids and tetrahedra types were defined by the amino acid types. The log-

likelihoods of all 8855 types of possible tetrahedra were computed from a large database of experimentally determined protein structures. The total change of these log-likelihoods of all tetrahedra involved in a point mutation was used to predict the change of protein stability. Strong correlation between the predicted and experimental $\Delta\Delta G^{\circ}$ values was obtained for five proteins, but the data set was not as large as those used in other methods such as FOLD-X.

The PoPMuSiC method [39] considered two types of potential, the torsion angle potential and the distance potential, in order to evaluate protein stability. The potentials were derived from observed frequencies of sequence and structure patterns in a large dataset of x-ray protein structures. Correlation coefficients between 0.80 and 0.87 were obtained between predicted and experimental $\Delta\Delta G^{\circ}$ values on hundreds of mutations in various environments, for seven different proteins and a synthetic peptide. However, the performance is relatively poor for mutations with solvent accessibility in the 40-50% range [38].

The area of $\Delta\Delta G^{\circ}$ prediction is still under active development. For example, the web tool CUPSAT [83] used structural environment specific atom potentials and torsion angle potentials to predict the $\Delta\Delta G^{\circ}$ values of point mutations. Some other methods also employ neural networks and support vector machines [16, 17, 22].

2.5 Site-Directed Protein Recombination

Protein recombination takes several forms including DNA shuffling [105], ITCHY [80] and SCRATCHY [69], StEP [1], and RACHITT [25]. In site-directed recombination, a set

of homologous parent genes are recombined at specified breakpoint locations, yielding a combinatorial set of hybrids [119, 72, 81, 89]. In contrast to stochastic library construction methods [105, 1, 25], site-directed approaches intentionally select breakpoint locations to optimize expected library quality, *e.g.* predicted disruption [72, 31, 127], and thus can be optimized beforehand in order to increase the possibility of obtaining useful hybrids. Recombination-based approaches, when combined with high-throughput screening and selection, can avoid the need for precise modeling of the biophysical implications of mutations. They employ an essentially "generate-and-test" paradigm. As always, the goal is to bias the "generate" phase to improve the hit rate of the "test" phase.

In order to increase the probability of obtaining folded and functional hybrids in site-directed recombination, it is desirable to retain the evolutionary relationships among amino acids that determine protein stability and function. The labs of Mayo and Arnold [119, 72] have established criteria that evaluate retainment of contacting residue pairs after recombination, and demonstrated the relationship between the amount of contact disruption and functional hybrids. Saraf and Maranas *et al.* [91] defined residue-residue "clash" based on charge, volume, and hydrophobicity and demonstrated the correlation between the number of such residue-residue clashes and the activities of functional hybrids. Such non-random association of amino acids, as expressed in pairwise potentials, has also been usefully applied in a number of other situations. For example, pairwise contact potentials [108, 73] play a large role in evaluating the quality of models in protein structure prediction [70, 99, 58, 41]. It has been suggested, however, that "it is unlikely that purely pairwise potentials are sufficient for structure prediction" [11, 20].

To better model the evolutionary relationships that determine protein stability and functionality, it may be necessary to capture the higher-order amino acid interactions that are ignored in simple pairwise models. Researchers have begun to demonstrate the importance of accounting for higher-order terms. The four-body potential discussed in Sec. 2.4 is a good example of employing a higher-order potential to predict protein stability changes upon mutation [20]. Similar formulations have been used to discriminate native from non-native protein conformations [61]. Geometrically less restricted higher-order interactions have also been utilized for recognition of native-like protein structures [100]. Recent work on correlated mutation analysis has moved from identifying pairwise correlations [40] to determining clusters or cliques of mutually-dependent residues that identify subclasses within a protein family and provide mechanistic insights into function [68, 110].

As we discussed in Chapter 1, the combinatorial number of breakpoint location sets makes it difficult to choose the optimal even with respect to a naïve criterion. Along with the search for effective criteria, there is also an on-going search for efficient algorithms to select the optimal breakpoint locations in site-directed recombination. Endelman *et al.* [81] formulated the breakpoint selection problem, using only pairwise potentials, as a shortest path problem. For choosing n breakpoints from N residues, an $N \times n$ matrix of nodes is built and edges are placed between two nodes in adjacent columns if the right index is larger than the left index. Breakpoint locations were added one by one from left to right, so the length of an edge represents the additional number of pairwise amino acid contacts broken by adding the current breakpoint given the previous one. In this representation, every feasible n-breakpoint library is represented as a path of length n from left to right, with

the node visited in column k corresponding to the position of the k^{th} breakpoint location. The optimal set of breakpoint locations thus corresponds to the shortest path, which can be identified in polynomial time.

Saraf *et al.* [90] developed an algorithm OPTCOMB that optimally balances library size and quality. OPTCOMB employs linear programming technique to determine the optimal breakpoint locations and which parents can be used for each fragment. The authors identified an optimal library size for the well-studied dihydrofolate reductase proteins from *E. coli*, *B. subtilis*, and *L. casei*, minimizing both the number of clashes between the fragments composing the library and the average number of clashes per hybrid in the library.

3. MODEL DISCRIMINATION BY CROSS-LINKING

It is advantageous to consider the possible outcomes of cross-linking before an experiment is conducted, in order to optimize experimental parameters and obtain the most information from an experiment. Similarly, if interpretation of the results of an experiment proves to be ambiguous, a subsequent experiment can be optimized to reduce the ambiguity. Variable experimental parameters include the cross-linker (particularly specificity and length) and the sequence itself, altered by planned mutations that are unlikely to affect the parent structure. For example, we could make a conservative change to `Lys` in order to introduce additional possible cross-links for BS^3, or make non-drastic substitutions in two residues to the widely-accepted `Cys` in order to test disulfide bond formation. Selecting cross-linker and mutation can be repeated, generating a family of experiments, each potentially providing additional information for model selection. This chapter develops experiment planning and data interpretation mechanisms for cross-linking. We demonstrate that our approach is extremely efficient and produces high-quality designs.

Our probabilistic cross-link analysis and experiment planning method is summarized in Fig. 3.1. First, computational analysis assesses feasibility of cross-links on a set of predicted models of a protein as discussed in Chapter 2. Since a cross-linker can span only

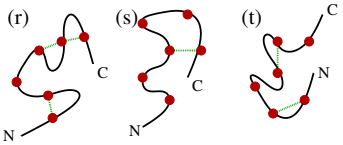
some maximum distance between a pair of residues, the feasibility of the possible cross-links varies across models. We perform this geometric feasibility analysis and collect the information into "cross-link maps", which indicate the conditional probabilities for cross-links for each model, under the particular experimental conditions. The example simply indicates high (H) or low (L) probability for some potential cross-links on three models. The potential for cross-linking for some pairs will be hard to evaluate, especially when significant dynamics are possible. These cross-links can be put into a third, ambiguous (A) class.

Based on the cross-link maps, our experiment planning algorithm evaluates the relative potential for positive support in order to select a set of experiments (shown in Fig. 3.1 as ovals) to cover the various possible pair-wise discriminations. Selected sites are cross-linked and then cross-links are detected in experiments (see Chapter 2).

Once experimental data are collected, characterization of the set of observed (and potentially the unobserved) cross-links provides evidence regarding the consistency of the models with the data. An observed high feasibility cross-link supports a model. A low feasibility cross-link that is not observed can also support a model, once the likelihood of cross-link detection is explicitly considered. Conversely, unobserved high feasibility and observed low feasibility cross-links provide evidence against a model. To account for limitations in the experimental detection of cross-links and potential experimental errors, we include two parameters, capture rate κ, indicating the rate of detection of feasible cross-links (that is, $1-\kappa$ equals the rate of false negatives), and noise rate ν, the detection rate of spurious infeasible ones (*i.e.* false positives). These rates will depend upon the cross-linker

Fig. 3.1: Model discrimination by cross-linking. (1) Different predicted models of a protein have different patterns of feasible cross-links (dotted lines). Cross-link maps capture the feasibilities (H, L, or A) in terms of conditional relationships for cross-links (rows) given models (columns). (2) Different experimental choices (cross-linkers, mutations) yield different cross-link maps. An experiment could enable selection of one model, if correct, over another (*e.g.* $r > s$) if the first model has enough potential positive support (H entries in the cross-link map) where the other doesn't (L entries). Some experiments provide only ambiguous information for a particular model (A entries). An experiment plan evaluates the relative potential for positive support in order to select a set of experiments (ovals) to cover the various possible pair-wise discriminations. (3) Experimental data, *e.g.* from mass spectrometry or gel electrophoresis, provide support for particular cross-links. (4) Experimental identification of cross-link $\mathbf{l}_{1,2}$ provides evidence for and against models r and s, based on consistency with cross-link maps and modulated by the capture and noise rates of the experimental method (here constant values κ and ν, for capture and noise respectively). The discrimination ratio combines terms for both observed and unobserved cross-links.

Step 1: Analyze geometric feasibility



	r	*s*	*t*
$\mathbf{l}_{1,2}$	*H*	*L*	*H*
$\mathbf{l}_{3,6}$	*L*	*H*	*L*
$\mathbf{l}_{4,5}$	*H*	*L*	*H*
$\mathbf{l}_{5,6}$	*H*	*L*	*L*
…			

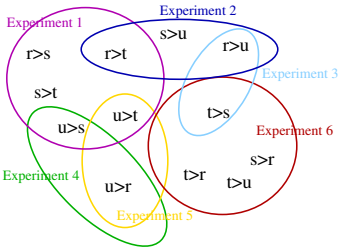
Step 2: Plan experiment(s) to maximize differences

Experiment 1

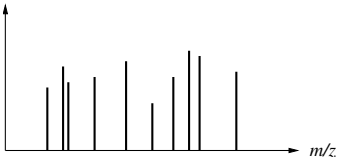
	r	*s*	*t*	*u*
$\mathbf{l}_{1,2}$	*H*	*H*	*L*	*A*
$\mathbf{l}_{1,3}$	*H*	*L*	*A*	*H*
$\mathbf{l}_{1,4}$	*H*	*L*	*L*	*H*
…				
$\mathbf{l}_{2,3}$	*H*	*H*	*L*	*L*
$\mathbf{l}_{2,4}$	*L*	*A*	*L*	*H*
…				

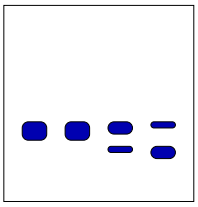
Experiment 2

	r	*s*	*t*	*u*
$\mathbf{l}_{1,2}$	*H*	*H*	*L*	*L*
$\mathbf{l}_{1,3}$	*H*	*H*	*A*	*L*
$\mathbf{l}_{1,4}$	*H*	*L*	*L*	*L*
…				
$\mathbf{l}_{2,3}$	*L*	*H*	*H*	*L*
$\mathbf{l}_{2,4}$	*A*	*H*	*H*	*H*
…				

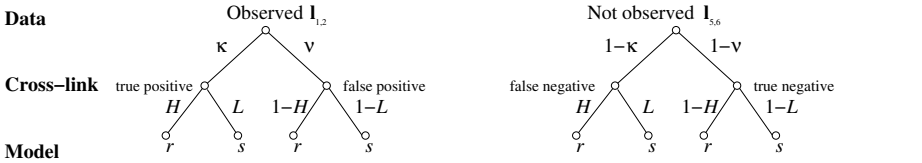


Step 3: Conduct experiments





Step 4: Discriminate models given experimental data



Discrimination $r > s$?

$$\frac{r}{s} \quad \frac{\kappa H + \nu(1-H)}{\kappa L + \nu(1-L)} \quad * \quad \frac{(1-\kappa)H + (1-\nu)(1-H)}{(1-\kappa)L + (1-\nu)(1-L)} \quad * \ \ldots$$

and peptides involved, the detection methods, and the experimental effort, but we will consider the simplest case of fixed rates. Support for models provides probabilities (Eq. 3.1), which are used in a ratio to compare two models (Eq. 3.3). When one model is sufficiently better than every other, model selection results.

In the rest of this chapter, we first develop a probabilistic framework for reasoning and data interpretation (Sec. 3.1); followed by the experiment planning metrics and algorithms (Sec. 3.2). Then we present the planning and simulation results for residue-specific cross-linking and disulfide trapping on several proteins, followed by a practical application of discriminating models of the pTfa protein from bacteriophage lambda by disulfide trapping (Sec. 3.3).

3.1 Probabilistic Framework

This section develops a basic framework for probabilistic reasoning about cross-links. Most probabilistic functions depend on the choice of experimental parameters; we leave those terms implicit except where necessary for clarity.

We are given a set $\mathcal{S}$ of predicted structure models. Each model $s \in \mathcal{S}$ has a prior probability, $p(s)$, which can be uniform or can incorporate scoring information from the modeling process. The task is to identify the model in S that is best, in terms of the prior and agreement with experimental data regarding a set $\mathcal{L}$ of possible cross-links. We bridge the gap between model and data in two steps: (1) consistency of cross-links with models, and (2) evidence for cross-links from data. Consistency of a cross-link $\mathbf{l}_i$ with a model

s is modeled with a conditional probability $p(\mathbf{l}_i \mid s)$. Support for a cross-link $\mathbf{l}_i$ from experimental data $\mathbf{d}$ is modeled with likelihood $p(\mathbf{d} \mid \mathbf{l}_i)$. Since we concentrate on the information content available via cross-linking, we take as given an interpretation of the data. For example, in the case of cross-link identification by mass spectrometry, likelihoods could be computed by predicting expected mass peaks for a given cross-link and comparing with observed spectra, using a distribution to model measurement error, and a mixture model to handle experimental complexities (*e.g.* missed proteolytic cleavage). A key part of these likelihoods that we explicitly model is the sparsity (false negatives) and noise (false positives) of the data.

Combining these terms then yields the support for each model from the data, by marginalizing over cross-link existence. We treat cross-links as independent, although it is certainly possible to model dependence due to such effects as common reactivity arising from cross-links sharing an amino acid side chain. Similarly, a model is conditionally independent of the data given the cross-links (models are not, for example, optimized with respect to the data). Thus we have

$$p(\mathbf{d} \mid s) = \prod_{i \in \mathcal{L}} \sum_{\ell \in \{0,1\}} p(\mathbf{d} \mid \mathbf{l}_i = \ell) \cdot p(\mathbf{l}_i = \ell \mid s) \tag{3.1}$$

In this approach, a model is supported by high feasibility cross-links that are observed and low feasibility ones that aren't. It is penalized by low feasibility cross-links that are observed and high feasibility ones that aren't. Fig. 3.1, Step 4, has two simple examples for one observed and one unobserved cross-link. The $p(\mathbf{d} \mid \mathbf{l}_i)$ terms are the noise and capture rates, and the $p(\mathbf{l}_i \mid s)$ terms arise from the cross-link map.

An interesting consequence of this realistic model is that, depending on the number of

potential cross-links, and their cross-link feasibility (H, L), capture (κ), and noise (ν) values, we should expect to observe some cross-links that are considered low feasibility in the correct structure. The expected number of identified cross-links among B low feasibility ones is $(\kappa L + \nu(1 - L)) \cdot B$. If $\kappa = \frac{1}{3}$, $\nu = 0.05$, $L = 0.1$ and $B = 25$, we expect to see about 2 infeasible cross-links wrongly identified. The potential identification of incorrect cross-links points out the need for multiple possible cross-links supporting a model selection.

Employing Eq. 3.1, we can reweight the prior distribution $p(s)$ by the information provided by the data:

$$p(s \mid \mathbf{d}) \propto p(\mathbf{d} \mid s) \cdot p(s) \tag{3.2}$$

and identify the maximum *a posteriori* model, or maximum likelihood model in the absence of informative priors.

A *posterior ratio* allows comparison of the consistency of two models ($r, s \in \mathcal{S}$) with the data.

$$\phi_{rs}(\mathbf{d}) = \frac{p(\mathbf{d} \mid r)p(r)}{p(\mathbf{d} \mid s)p(s)} \tag{3.3}$$

In the present context, we allow for the possibility of priors, although we treat them as uniform. When priors are ignored, this ratio becomes a so-called Bayes factor. A model can be confidently selected when the ratio with respect to every other model is sufficiently large.

3.2 Experiment Planning Metrics and Algorithm

The problem of characterizing the utility of an experiment has been well-studied in the statistical literature; for example, relative entropy (Kullback-Leibler distance) between posterior and prior distributions is one natural approach that would capture the expected effects of reweighting the models given data. We employ a complementary approach that uses pairwise differences in cross-link maps so that we can make explicit trade-offs among key properties of practical importance for our application — discriminability, coverage, balance, ambiguity, and cost. Cost becomes simply the number of experiments for those with uniform cost such as disulfide trapping.

Intuitively, a pair of models with very different cross-link maps (*i.e.* disagreeing about feasibility of many cross-links) has a higher probability of being discriminated than a pair with very similar cross-link maps. We separately consider the two directed discriminations in favor of one or the other model, which we characterize as *cross-link map differences*, $d(r, s)$ and $d(s, r)$. Using H and L feasibilities as in Fig. 3.1, $d(r, s)$ would simply be the size of the set $\mathcal{L}_r$ of cross-links that have H in r and L in s, and similarly for $d(s, r) = |\mathcal{L}_s|$ (note that cross-links for which they agree cancel out in the discriminability ratio, Eq. 3.3). Now we consider whether the discriminability ratio ϕ is sufficient to select r if r is indeed correct. Since this analysis is done before data are collected, we must take the expectation over all possible datasets

$$E\{\phi_{rs} \mid r\} = \int_{\mathbf{d}} \phi_{rs}(\mathbf{d}) \cdot p(\mathbf{d} \mid r)\, \mathrm{d}\mathbf{d} \tag{3.4}$$

In general, this integral cannot be evaluated analytically. However, it can be simplified

under the assumptions we have been discussing: independent feasibility of cross-links using fixed H and L, detected under fixed rates for capture κ and noise ν. The probability of capturing a high feasibility cross-link is then $\alpha = H\kappa + (1 - H)\nu$, the sum of capturing it correctly and of it showing up incorrectly. The probability of capturing a low feasibility cross-link is $\beta = L\kappa + (1 - L)\nu$. If r is the correct model, then each cross-link from $\mathcal{L}_r$ contributes $\frac{\alpha}{\beta}$ to the ratio if observed or $\frac{1-\alpha}{1-\beta}$ if not (both contribution ratios are reciprocated for the $\mathcal{L}_s$ cross-links). Assuming independence of cross-links, the expected value is multiplicative, and we can separately analyze the expected contribution of each cross-link to the ratio. Each cross-link in $\mathcal{L}_r$ contributes

$$\lambda = \alpha \cdot \frac{\alpha}{\beta} + (1 - \alpha) \cdot \frac{1 - \alpha}{1 - \beta} \tag{3.5}$$

Cross-links in $\mathcal{L}_s$ have a similar formula with α and β switched, giving γ. Examination of γ and λ demonstrates that κ must be greater than ν for effective discrimination, and the greater the difference, the greater the effectiveness of the experimental system.

The expected ratio in Eq. 3.4 then becomes

$$E\{\phi_{rs} \mid r\} = \lambda^{|\mathcal{L}_r|} \cdot \gamma^{|\mathcal{L}_s|} \tag{3.6}$$

We can rewrite λ as

$$1 + \frac{(\alpha - \beta)^2}{\beta(1 - \beta)} \tag{3.7}$$

to see that it is greater than one (assuming $\kappa > \nu$). Similarly, $\gamma > 1$, so the expectation of the ratio $E\{\phi_{rs} \mid r\}$ increases monotonically with $|\mathcal{L}_r|$ and $|\mathcal{L}_s|$. Thus we can use the cross-link map differences as an easily interpretable measurement of the potential for correctly making a selection.

Averaging (or simply summing) the expectation of ratios over all model pairs yields a measure of the overall expected information provided by an experiment. In our cross-link map difference approach, we simply sum up the number of model pairs with a cross-link map difference of at least some threshold Δ.

$$c(\mathcal{S};\,\Delta) = \sum_{r \neq s \in \mathcal{S}} I\{d(r,s) \geq \Delta\} \tag{3.8}$$

where the indicator I takes value 1 if the predicate is true and 0 if it is false. We call this the discriminable model-pair *coverage* of the experiment. We note that this metric does not require the same data to be used to achieve acceptable discriminability for one model against different models (*e.g.* r can be better than s and t under two different subsets of its cross-links). While model selection only requires finding one model to be better than the rest, the coverage metric seeks to support discrimination of all pairs of models. In the absence of an informative prior, any model could be the selected one, so we must consider all pairwise discriminations.

Whether an observed cross-link in $\mathcal{L}_r$ or an unobserved cross-link in $\mathcal{L}_s$ provides a larger contribution for r (*i.e.* whether λ or γ is bigger) depends on the values of H, L, κ, and ν. The uncertainty in the relative values of λ and γ provides additional motivation for a balanced design, *i.e.* an (approximately) equal number of cross-links in $\mathcal{L}_r$ and $\mathcal{L}_s$. Formally, we evaluate imbalance ib in terms of the potential positive evidence for each pair of models, as measured by cross-link map differences, with Δ difference considered always sufficient. Additional discriminability greater than Δ is neither penalized nor selected for

in our planning algorithm.

$$ib(\mathcal{S};\,\Delta) = \sum_{r \neq s \in \mathcal{S}} |\min(d(r,s),\,\Delta) - \min(d(s,r),\,\Delta)| \tag{3.9}$$

Since variability in H and L arises from modeling uncertainty and protein flexibility, it is intuitively desirable to use for discrimination only cross-links that are most feasible in one model and least feasible in the other. We make this property explicit in terms of a parameter we call the *distance ambiguity region* A — a range of cross-linking distances which cannot be associated with strong feasibility or infeasibility with respect to a model. In the current case of discrete cross-link map differences, we simply don't include such positions in the formulas for discriminability and coverage. In the more general case, this region would be reflected in the choice of distribution for $p(\mathbf{l}_i \mid s)$.

Our experiment planning mechanism takes as input a set of possible experiments $\mathcal{E}$ to be considered, each with an associated set of cross-link maps $p(\mathbf{l}_i \mid s; e)$. It then determines experiments to be conducted $\mathcal{E}' \subset \mathcal{E}$, so as to maximize discriminable model pair coverage C and minimize imbalance ib and number of experiments $N = |\mathcal{E}'|$.

The optimization problem can be shown to be a member of the class of NP-hard problems. NP-hardness follows by reduction from SETCOVER: the objects to be covered correspond to model pairs, the covering sets correspond to experiments, and a binary cross-link map indicates which objects (model pairs) a particular set (experiment) covers (discriminates). In fact, our problem generalizes many variations on SETCOVER that maximize coverage and minimize the number of sets. Greedy algorithms have proved effective in such contexts, both practically and theoretically (*e.g.* the greedy algorithm for SETCOVER would provide an approximation to $1 + \log |S|$ in covering a set S [56]). Thus we pursue a

XLINKPLAN($\mathcal{S}$, $\mathcal{E}$, Δ, A, $N_{\max}$, $C_{\max}$)
$\mathcal{E}' \leftarrow \emptyset$
$P \leftarrow \{\langle r, s\rangle \;\; r \neq s \in \mathcal{S}\}$
$P' \leftarrow \emptyset$
for each $\langle r, s\rangle \in P$
    if residue-specific with possible cross-linkable residue pairs $\mathcal{L}$
        $w(r, s) \leftarrow 1$
        for each $e \in \mathcal{E}$
            let $\mathcal{L}'$ be $\{\mathbf{l}_i \in \mathcal{L} \;\; \|\mathbf{l}_i\|_r < A.\min \text{ and } \|\mathbf{l}_i\|_s > A.\max\}$
            covers(e, $\langle r, s\rangle$) $\leftarrow |\mathcal{L}'| \geq \Delta$
    else if disulfide
        $w(r, s) \leftarrow \Delta$
        for each $e \in \mathcal{E}$
            covers(e, $\langle r, s\rangle$) $\leftarrow (\|\mathbf{l}_e\|_r < A.\min \text{ and } \|\mathbf{l}_e\|_s > A.\max)$
repeat
    let e be $\arg\max_{e \in \mathcal{E} - \mathcal{E}'} \sum_{\langle r,s\rangle \in P} w(r, s) \cdot \text{covers}(e, \langle r, s\rangle)$
    $\mathcal{E}' \leftarrow \mathcal{E}' \cup \{e\}$
    for each $\langle r, s\rangle$ such that covers(e, $\langle r, s\rangle$) $= 1$
        $w(r, s) \leftarrow \max(0, w(r, s) - 1)$
        if $w(r, s) = 0$: $P' \leftarrow P' \cup \{\langle r, s\rangle\}$
until $|\mathcal{E}'| = N_{\max}$ or $|P'| \geq C_{\max}$

Fig. 3.2: Greedy algorithm, XLINKPLAN, given a set $\mathcal{S}$ of models, a set $\mathcal{E}$ of possible experiments, desired discriminability Δ, ambiguity region A, and maximum number of experiments $N_{\max}$ and coverage $C_{\max}$. The output is a subset $\mathcal{E}'$ of experiments covering a subset P' of model pairs. For residue-specific cross-linking plans, a model pair must be covered by a single experiment at the given Δ level. For disulfide trapping plans, the weight w on a model pair keeps track of the remaining coverage to complete Δ, to be provided by subsequent experiments.

greedy approach.

Fig. 3.2 outlines our algorithm, XLINKPLAN. We employ directed pairs of models (*i.e.* both $\langle r, s \rangle$ and $\langle s, r \rangle$), rather than undirected pairs, in order to reach balanced design. A potential cross-link $\mathbf{l}_i$ is informative in discriminating a directed model pair if its cross-linking distance (denoted $\|\mathbf{l}_i\|$) is short enough in the first model and long enough in the second, relative to the ambiguity region. In this manner, the algorithm considers only positive evidence towards a particular discrimination goal, minimizing imbalance. Negative evidence will also arise in the experiment and contribute additional discrimination as in Eq. 3.1 (but this is not considered in planning). For lysine-specific experiments, a particular experiment is noted as covering a model pair if at least Δ cross-links are informative. For disulfide experiments, coverage is accumulated over multiple experiments, and so each experiment is consider as (partially) covering if it has an informative cross-link. The total number of covering experiments must then reach Δ, and a pair's weight w keeps track of the remaining coverage required. The algorithm greedily selects experiments, stopping when the number of experiments reaches the maximum number $N_{\max}$ or the desired coverage $C_{\max}$ is satisfied. At each point, the marginal utility of an additional experiment is evaluated by the weighted coverage sum, in which the weight w represents the current importance of covering a particular model pair. Each pair's weight is initially simply 1 (residue-specific) or the desired discriminability Δ (disulfide), and coverage by an experiment then decrements the weight.

In fact, it can be proved that XLINKPLAN provides an approximation to $1 + \log \Delta + \log |S|$ by slightly modifying the amortized analysis for the GREEDY-SET-COVER prob-

lem described by Cormen *et al.* [29]. Basically, we assign a cost of 1 to each set (residue pair) selected by XLINKPLAN, distribute this cost over the elements (model pairs) covered by the selected set based on their current weights, and then use these costs to derive the desired relationship between the size of an optimal set cover and the size of the set cover returned by XLINKPLAN. Usually we would have a Δ much smaller than $|S|$.

3.3 Results

3.3.1 Residue-specific Cross-link for Model Discrimination

We first studied probabilistic discriminability analysis and experiment planning using lysine-specific cross-linking and three different proteins. The primary test case is basic fibroblast growth factor (FGF-2, PDB ids 4FGF, crystal, and 1BLA, NMR), due to its earlier use in model discrimination by cross-linking [128]. Alternative threading models for FGF-2, using twelve of the published template structures, were obtained via the protein fold-recognition meta-server [63]; several of the published templates could not be suitably matched to the FGF-2 sequence given current threading programs queried by the server. Two of the models are of the same fold (β trefoil) as the current structure, and the correct NMR structure (PDB id 1BLA) is also included in the model set. The other test cases were chosen from CASP4 [75] targets with many high-quality models: deoxyribonucleoside kinase (PDB id 1J90) and α-catenin (PDB id 1L7C). Predicted models that are less complete than the correct one are ignored. In total we employed 13 models for FGF-2, 85 models for

deoxyribonucleoside kinase and 50 models for α-catenin.

We consider five commercially-available, water-soluble, and primary amine-reactive N-hydroxysuccinimide, sulfo-N-hydroxysuccinimide, or imido ester cross-linkers with different lengths between the reactive groups: sulfo-DST cross-links `Lys` N^{ζ} to `Lys` N^{ζ} at a distance of 6.4 Å, DSG at 7.7 Å, DMP at 9.2 Å, BS^3 at 11.4 Å, and sulfo-EGS at 16.1 Å. Previously, only information of geometric feasibility has been employed for making structural inference from cross-linking. We follow that procedure here, while recognizing that accessibility and reactivity can be measured separately by reaction with monofunctional reagents [78].

Following earlier work [128], for each model, we computed `Lys` C^{α} to `Lys` C^{α} straight-line distance (the position of the reactive N^{ζ} atom is generally both uncertain and mobile); this requires adding 12.4 ($2 * 6.2$) Å to the maximal cross-linker length to allow for the maximal C^{α}–N^{ζ} side chain length. Because distributions of distances less than maximal are most highly populated in solution [43], it is reasonable that potential cross-links with distances that are some value less than the maximum should be considered most feasible. At the same time, cross-links with distances exceeding the maximum are considered infeasible, while those in between are considered ambiguous. Our strategy of ignoring the ambiguous cross-links for model discrimination leads to a smaller number of utilized cross-links and thus a smaller probability of making a decision. However, it simultaneously reduces the possibility of utilizing a spurious cross-link and thus increases the probability that, when a decision is made, it is a correct one.

Chemically, there are two components to the reduction in effective cross-linker length.

One arises from the relative rarity of the maximally extended conformation of the cross-linker, and the other from lack of maximum extent and deviation from in-line orientation of the lysine side chains. For cross-linker BS^3, the cross-linker conformation component is 2.5 Å [43], and we estimate the same value for the side chain component. This creates an ambiguous region 5 Å wide where cross-links are feasible but less probable. For BS^3, this region extends from 19 Å to the maximum C^α–C^α distance of 24 Å. We have checked this ambiguity region against FGF-2 cross-linking data [128], and have found that, as expected, the capture rate for geometrically feasible cross-links ($<$ 19 Å) is greater than that for the ambiguous ones (19–24 Å), 31% vs. 24%. Then, we employ a capture rate κ of $\frac{1}{3}$. In addition to the expected effect on κ, the application of an ambiguity region is expected to also improve our ability to accurately classify potential cross-links as feasible or infeasible (increase the difference between the probabilities H and L). In all subsequent analyses, we define each cross-linker's ambiguity region ranging from its maximum extent to 5 Å less.

The discriminability Δ reflects the extent of confidence that we plan for in the selection of one model over another in a pair. It is the anticipated discriminability value for positive data if all possible cross-links in a planned set of experiments were detected without errors. Due to the inevitability of errors, the expected level achievable on average is $\kappa\Delta - \nu\Delta$ (see Sec. 3.2). Thus the experimenter must plan for discriminability greater than the level that is satisfactory for discrimination after collecting experimental data. An extreme example arises if we expect a low capture rate, as from residue-specific cross-linking; then we must require a high Δ so that the expected contribution to discrimination is sufficient. Thus when we plan for $\Delta = 6$ but have a capture rate of $\kappa = \frac{1}{3}$ and noise rate of $\nu = 0.05$,

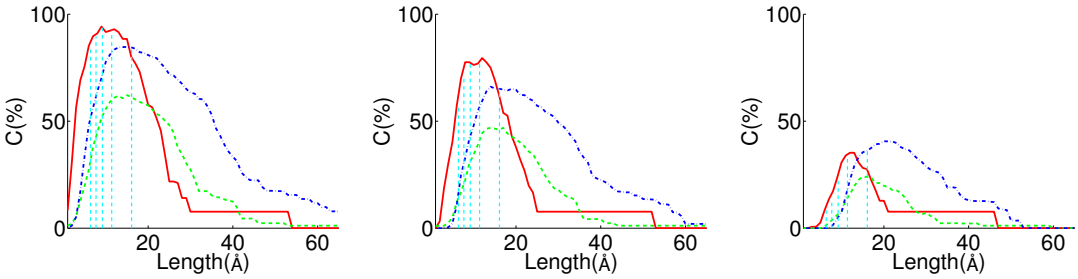


Fig. 3.3: Optimal cross-linker lengths for 3 different sets of protein models — FGF-2 (solid line), deoxyribonucleoside kinase (dashed line), α-catenin (dotted line) — over potential lengths from 1 to 65 Å in 1 Å steps. For each length, discriminable model-pair (directed) coverage was determined at Δ of 3, 6, and 12 respectively. Lengths are indicated (thin dashed vertical lines) for five commercially-available cross-linkers, sulfo-DST (6.4 Å), DSG (7.7 Å), DMP (9.2 Å), BS^3 (11.4Å), and sulfo-EGS (16.1 Å), and results are tabulated in Tab. 3.2.

Tab. 3.1: Optimal cross-linker length for three proteins of varying size, with Δ at 3, 6, and 12. Among the five commercially available cross-linkers we predict that three of them, DMP (9.2 Å), BS^3 (11.4Å), and sulfo-EGS (16.1 Å), would be variously optimal for these models.

protein	# residues	# lysines	#models	optimal cross-linker length $\Delta =$ 3	6	12
FGF-2	146	14	13	9Å	12Å	12Å
deoxyribonucleoside kinase	230	13	85	15Å	14Å	15Å
α-catenin	269	14	50	15Å	14Å	20Å

we can expect to actually observe 1.7 discriminatory cross-links on average in favor of the winning model.

In Fig. 3.3, we plot the discriminable model-pair percent coverage at Δ of 3, 6, and 12, while varying potential cross-linker length for the three test proteins. The optimal cross-linker length, summarized in Tab. 3.1 for our examples, depends on the models and the relative positions of the reactive sites. Theoretically, with the same number of reactive sites and a random distribution of them on the protein surface, the optimal cross-linker length would

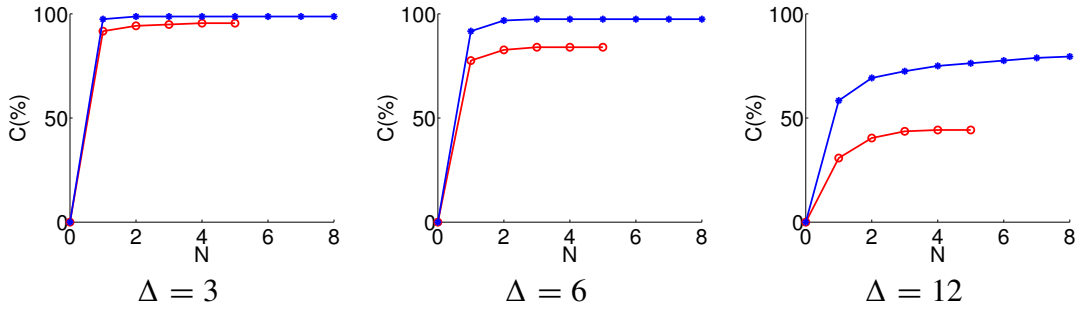


Fig. 3.4: Improvement in coverage by multiple-experiment plans for FGF-2. Sets of experiments were planned by XLINKPLAN, choosing for each experiment a cross-linker from among five commercially-available reagents (dashed line) or choosing both a cross-linker and a possible conservative mutation to `Lys` (solid line). Sets of up to five experiments were planned for the former case, saturating the possibilities of cross-linker choice, while sets of up to eight were planned for the latter case, which includes mutations. The coverage was determined for each plan, at different choices for discriminability Δ. The set of 8 experiments selected with choice of mutation at $\Delta = 12$ is listed in Tab. 3.2

be a function of protein size — the larger the protein, the longer the optimal cross-linker length. The three proteins have a similar number of lysines; hence the larger proteins deoxyribonucleoside kinase and α-catenin are better discriminated with longer cross-linkers than the smaller FGF-2. Our planning method can be used for choosing suitable cross-linkers for a particular protein or as a guide for designing novel cross-linkers [117]. The strange right tail of the FGF-2 curve is due to the elongated model based on the D-UTPase (β-Clip) template, which requires longer cross-linkers for discrimination.

Fig. 3.4 shows the coverage achieved as additional experiments are added to the plan by XLINKPLAN. For residue-specific cross-linking, data from each experiment are handled independently; that is, each pair is distinguished based on data gathered from a single experiment. This allows closer approximation to the probabilistic assumption of independence. For each experiment, selecting the optimal cross-linker improves coverage, although

Tab. 3.2: Cross-linking experiment plan for FGF-2. The greedy set of 8 experiments, each involving one possible Arg, Asn, Gln, or His to Lys mutation, and a choice of commercially available cross-linker, was determined. Each experiment is shown on a line, along with the coverage (percentage of directed model-pairs discriminated) at $\Delta = 12$. The total coverage provided by all 8 experiments is 79.49% of the 156 directed model-pairs, which is very close to the plateau value of 80.13%.

Experiment	Cross-linker (length, Å)	Mutated residue	Single coverage	Cumulative coverage
1	DMP (9.2)	Arg69	58.33%	58.33%
2	BS^3 (11.4)	Arg116	53.85%	69.23%
3	BS^3 (11.4)	Arg53	58.33%	72.44%
4	sulfo-EGS (16.1)	Arg90	41.03%	75.00%
5	DMP (9.2)	Arg106	50.64%	76.28%
6	sulfo-EGS (16.1)	Arg118	44.23%	77.56%
7	DMP (9.2)	Asn110	52.56%	78.85%
8	DMP (9.2)	Arg129	55.77%	79.49%
Total				79.49%

a plateau of diminishing returns is reached.

With the ease of making site-directed mutations by high-throughput means, a natural extension to cross-linking strategies is the creation of new sites for cross-linking reaction. In particular, conservative mutations (from Arg, Asn, Gln, or His) to Lys can be planned to add reactive sites. As can be seen, the addition of making just one conservative mutation as an experimental option allows higher coverage and/or discriminability, demonstrating that this is a valuable strategy if the number of natural sites is insufficient. Tab. 3.2 shows a sample planning result for a set of eight experiments with one cross-linker choice and one mutation possible per experiment, increasing the coverage from 58.3% to 79.5% after combining eight. In this strategy, if there are k possibilities for conservative changes in a protein and l choices of cross-linkers, then there are $\binom{kl}{N}$ possible experiment plans for N experiments. For FGF-2 and five potential cross-linkers there were $\binom{22*5}{8} \approx 4.1 * 10^{11}$ possibilities for this 8-experiment plan; our algorithm provides a valuable tool for selecting

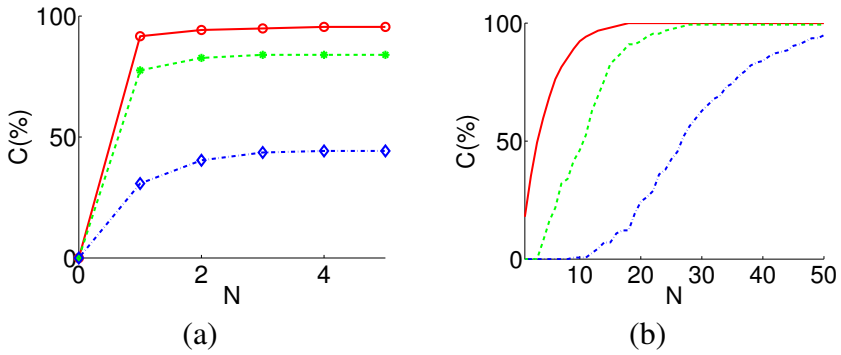


Fig. 3.5: Coverage of FGF-2 models by (a) lysine-specific and (b) disulfide cross-linking experiments planned by XLINKPLAN, as a function of desired discriminability Δ. (a) The set of N (from 1 to 5) experiments involving five commercially available cross-linkers were planned by XLINKPLAN using Δ of 3 (solid line), 6 (dashed line), or 12 (dotted line). The coverage at the chosen discriminability is indicated for each set of experiments. (b) The XLINKPLAN set of N (from 1 to 50) disulfide trapping experiments were planned using Δ of 1 (solid line), 2 (dashed line), or 4 (dotted line), and an ambiguity region of 9–21 Å.

the best ones.

The threshold discriminability value Δ has a significant influence on the planning result. Different levels of Δ affect the choice of experiments, as well as the coverage attainable. Fig. 3.5(a) shows the coverage resulting from multiple experiments at different Δ values. While good coverage can be achieved at low Δ values with a smaller number of experiments, the chance for error is higher.

3.3.2 Disulfide Trapping for Model Discrimination

In planning for disulfide trapping, XLINKPLAN considers pairs of residues for cysteine mutation (excluding drastic mutations from `Phe`, `Trp`, `Tyr`, `Pro`, and `Gly`). As before, planning parameters include the desired discriminability level Δ and the ambiguity region A. In this case, we construct A around a model C^{β}–C^{β} distance of 13 Å, the midpoint of

a sigmoidal transition of a 3 log difference in rates of disulfide formation [18], and expand A in increments of -1 and $+2$ to account for the asymmetry in the distribution of C^{β}–C^{β} distances relative to the transition midpoint value. Beyond estimating C^{β}–C^{β} distances, we do not construct a full geometric analysis of disulfide geometry [104], since protein dynamics override these considerations for many proteins [18] and our method does not require picking those disulfides that impart the greatest stability.

Fig. 3.5(b) shows disulfide trapping experiment plans for FGF-2, produced by XLINK-PLAN. While, as with residue-specific cross-linking, there are diminishing returns from doing more experiments, the enormous variety of possible disulfide experiments allows nearly full coverage to be achieved even at high Δ levels if enough experiments are conducted. Assuming, as above, that κ in lysine-specific cross-linking is about $\frac{1}{3}$, the $\Delta = 3, 6, 12$ curves in Fig. 3.5(a) are analogous to the $\Delta = 1, 2, 4$ curves in Fig. 3.5(b).

Since `Phe`, `Trp`, `Tyr`, `Pro`, and `Gly` comprise approximately 21% of the residues in an average protein, the number of possible disulfide trapping experiments is about $\binom{0.79n}{2} \approx 0.31n^2$; for N planned experiments, the number of possible combinations is about $\binom{0.31n^2}{N}$. In the FGF-2 case, there are in total 5565 possible dicysteine mutations. The number of all possible combinations of choosing 5 experiments from these is more than 10^{16}, while choosing 50 is more than 10^{120}. These numbers are clearly intractable to an exhaustive search for the optimal plan.

In disulfide trapping, different numbers of experiments generate a wide range of coverage. Depending on the planned Δ, 100% coverage is achieved only with a large number of experiments. However, coverage can be viewed as a conservative estimate of ability to

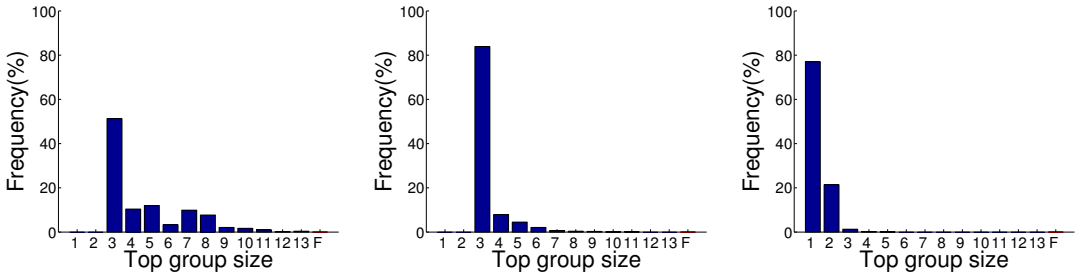


Fig. 3.6: Simulation of disulfide trapping for FGF-2, using a set of planned experiments for each coverage level, at $\Delta = 3$. Simulations employ high feasibility $H = 0.9$, low feasibility $L = 0.1$, capture rate $\kappa = 0.95$, and noise rate $\nu = 0.05$, hence $\lambda = \gamma = 5.31$. As explained in the text, we plan conservatively for $\Delta = 3$ and discriminate with $\Delta = 2$ to allow for the anticipated errors. In this case the appropriate threshold for the posterior ratio is still greater than 400-fold ($\lambda^{1.8}\gamma^{1.8}$). Shown is the frequency of the size of the top group in each simulation, over 1000 runs. The failure group (F) indicates cases when the correct structure has been eliminated from the top group.

discriminate, and practical experiment plans need not attain 100% coverage. To illuminate the relationship between coverage and experimental success, a simulation of a disulfide experiment plan at $\Delta = 3$ was conducted, using different numbers of experiments and corresponding coverage levels (Fig. 3.6). The result of each disulfide cross-linking experiment was simulated according to the geometric feasibility in the correct structure. Simulated errors were introduced according to feasibility $H = 0.9$ and $L = 0.1$ and capture and noise rates $\kappa = 0.95$ and $\nu = 0.05$. The simulation results are robust to a range of these parameters (not shown). Models are discriminated based on the posterior ratio exceeding a selected threshold. If we plan for a particular Δ, positive evidence (cross-links expected by the winning model that correctly show up, minus those expected by the losing model that spuriously show up) is expected to contribute a factor of $\gamma^{\kappa\Delta-\nu\Delta}$ to the posterior ratio, and negative evidence (cross-links expected by the losing model that correctly don't

show up, minus those expected by the winning model that fail to show up) contributes $\lambda^{(1-\nu)\Delta-(1-\kappa)\Delta}$ (Eq. 3.6, adjusted for expected error). By planning for $\Delta = 3$, confident discrimination by a ratio corresponding to $\Delta = 2$ can be expected even in the presence of this noise. In each simulation, we determined, with respect to the $\Delta = 2$ threshold, which models were eliminated by losing a pairwise comparison. The remaining "top group" of uneliminated models typically contains the correct structure and as few as one or two others, typically the other β trefoil models. With 86% coverage, the top group contains only these models in more than 80% of the cases. With sufficiently many experiments ($N = 42$), even the two most similar models can be distinguished more than 75% of the time. Due to false positives and negatives (since $\kappa \neq 1$ and $\nu \neq 0$), the correct structure might be eliminated. However, in this simulation, elimination of the correct model happens infrequently (less than 0.01%) since we require a sufficiently high ratio in order to make a decision.

3.3.3 Practical Example: Disulfide Trapping for pTfa Model Discrimination

We put our planning mechanism into practice on the pTfa protein of bacteriophage lambda. The pTfa protein and its homologs are chaperones required for the assembly of trimeric tail fibers in those phage lambda strains ("Ur-lambda") resembling the original wild-type isolate [52], and in related phages such as T4 [74, 50]. Genetic data suggest that the activity of pTfa and its homologs is an extreme example of chaperone activity, in which the structure of the final tail fibers (their ability to bind host membrane components) is partially

determined by the structure of the chaperone [49].

Lambda pTfa is a small 194 amino acid protein, but no structural information is available for it or any homolog. Crystallization trials of pTfa readily yield crystals, but they fail to diffract (Hashemolhosseini *et al.*, 1996; van der Woerd and Friedman, unpublished results). We submitted the pTfa sequence to the fold recognition meta-server [63]. Alignments returned from the fold recognition meta-server [63] (see also the references to the individual methods cited therein) were evaluated based on the agreement between the patterns of secondary structure predicted for pTfa and observed in the potential templates, as well as via structural assessment of the resulting crude models [60]. Three potential templates were identified by different fold recognition programs (Tab. 3.3). The DnaK template (1DKZ) reported by FUGUE [97] was selected as a potential template and used to build a model of the full-length pTfa protein using the "Frankenstein's monster approach" [60]. Additional fold recognition analyses were performed using the multiple sequence alignment of the pfam02413 family, of which pTfa is a member. Templates 1LIZ and 1CKM of the OB-fold were found using the pfam alignment and used to generate two additional models.

A large number of decoy models for the 1–108 residue fragment were also developed with the *ab initio* folding program Rosetta [99]. 15456 decoy models were clustered and 100 top clusters were selected for further analysis. Rosetta was only rarely able to generate decoy models with high contact order, presumably due to the high β-strand content. Poorly modeled regions provide further incentive for requiring positive data in discrimination.

If our primary concern is to distinguish the three high quality threading models, their

Tab. 3.3: Three potential templates for pTfa protein, their source, fold, and function.

Index	Program	Template	Fold-type	Function
1	Fugue	1dkz	DnaK-like	Chaperone DnaK substrate binding domain
2	3D-PSSM	1liz	OB-fold	Heme chaperone Ccme
3	Fugue	1ckm	OB-fold	mRNA capping enzyme

Tab. 3.4: A full coverage plan for three pTfa models with potential templates in Tab. 3.3 with discriminability $\Delta = 2$, ambiguity region 10–19 Å, and number of experiments $N = 6$. Each model pair is covered twice (a coverage pattern value of 1 indicates support for the first model over the second), and each model is expecting the same number (3) of high feasibility and low feasibility cross-links, a perfect balanced design ($ib(\mathcal{S}, \Delta) = 0$).

Residue Pair		Distances			Feasibilities			Coverage Pattern					
		1	2	3	1	2	3	1vs2	1vs3	2vs1	2vs3	3vs1	3vs2
ASN59	VAL68	26.95	28.33	5.63	*L*	*L*	*H*	0	0	0	0	1	1
ALA40	ALA63	21.19	6.86	40.99	*L*	*H*	*L*	0	0	1	1	0	0
GLN8	ASP83	8.61	24.04	23.62	*H*	*L*	*L*	1	1	0	0	0	0
THR75	SER88	6.79	6.92	24.52	*H*	*H*	*L*	0	1	0	1	0	0
LEU18	ASP83	29.98	8.40	8.95	*L*	*H*	*H*	0	0	1	0	1	0
LYS13	ASN22	4.94	19.88	3.15	*H*	*L*	*H*	1	0	0	0	0	1
Total					3*H*,3*L*	3*H*,3*L*	3*H*,3*L*	2	2	2	2	2	2

small number allows explicitly attaining balance by seeking dicysteine mutations for all feasibility patterns of desired coverage, here 2 (Tab. 3.4). Due to the small number of models, there are many dicysteine mutations with the same model-pair coverage, so we employed as a tie-breaking metric the standard deviation of the difference in cross-linking distance across the model pairs. The final plan with ambiguity region 10–19 Å is summarized in Tab. 3.4. Some automatically-selected residue pairs were manually excluded from further consideration when the residues were poorly modeled, when there was substantial protein between two close residues (which would require internal motions to allow cross-linking) or when there was thought to be insufficient mobility to allow cross-linking.

The optimal experiment plan (Tab. 3.4) for discrimination of the threading models of

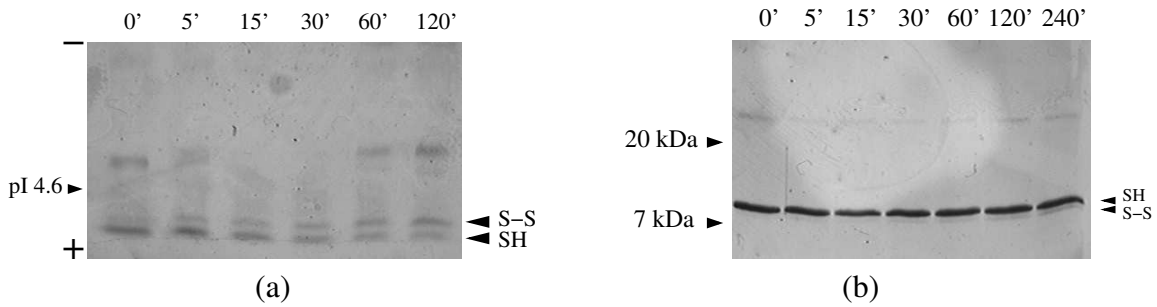


Fig. 3.7: Disulfide cross-linking of mutants (a) *H H L* and (b) *L H H*. Oxidation of dicysteine mutants by atmospheric oxygen was catalyzed by 15 μM Cu^{2+} ions for the indicated time before quenching. Dicysteine mutant *H H L* has only 11 residues between the two cysteines, resulting in an unobservable difference in mobility on SDS gels (not shown). This mutant was thus analyzed by isoelectric focusing (a). The disulfide form runs further from the anode, which is at the bottom of the gel as shown. Mutant *L H H* was analyzed on 20% homogeneous Phast gels (b), where the disulfide form has slightly greater electrophoretic mobility.

the lambda pTfa protein has been conducted by Dr. Patrick K. O'Neil in Alan M. Friedman's lab at Purdue University. The phage lambda pTfa 1-108 fragment was produced from the intact pTfa protein by PCR subcloning into the pET30 vector (Novagen) using the restriction sites NdeI and HindIII, leaving a product without N or C terminal tags. Dicysteine mutants of the pTfa 1-108 fragment were made by the Quik-Change method (Stratagene), and confirmed by sequencing both strands of the entire gene. Proteins were overexpressed in E. coli strain BL21(DE3)/pRIL, and purified by ammonium sulfate precipitation, hydrophobic interaction chromatography on Bakerbond HI-Propyl (Baker), followed by ion exchange chromatography on HiTrap Q (Pharmacia). Purified proteins were more than 95% homogeneous as detected by SDS-PAGE. The results consistently support the DnaK model. Fig. 3.7 shows the results of oxidizing mutants with patterns *H H L* and *L H H*. Oxidation by atmospheric oxygen, using a Cu^{2+} catalyst, reveals that *H H L* oxidizes at least 20-fold faster than *L H H*. Concentrations of catalyst from 5 to 25 μM give consis-

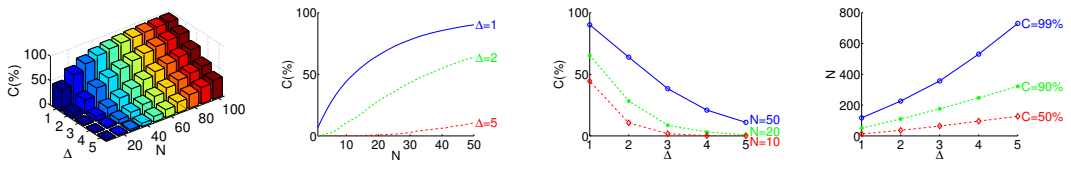


Fig. 3.8: The relationship between coverage percentage C (%), discriminability Δ, and number of experiments N in disulfide experiments planned by XLINKPLAN for 103 pTfa models with ambiguity region 9–21 Å. (a) Varying all parameters. (b), (c), (d) Varying pairs of parameters, while fixing the third at the indicated values.

tent results, while mM concentrations of catalyst rapidly oxidize both proteins. A complete kinetic analysis for the entire experiment plan is beyond the computational focus of this thesis, however these preliminary results point out the importance of a kinetic analysis to avoid false positives and thus improve ν.

We have also analyzed the potential to discriminate the entire set of 103 models (3 threading models plus 100 Rosetta decoys) under disulfide cross-linking. We applied our planning algorithm with an ambiguity region of 9–21 Å. Fig. 3.8 summarizes the results in terms of the three planning parameters, number of experiments N, discriminability threshold Δ, and coverage $C(\%)$. The 3D plot of these three variables in shown in Fig. 3.8(a); 2D slices in several directions are shown in Fig. 3.8(b,c,d).

We further focused on the differential ability to discriminate sets of relatively different models as compared to relatively similar ones. We identified a set of 21 decoys all contained within a single one of the 100 Rosetta clusters. They have pairwise RMSDs ranging from 4.1 Å to 9.9 Å (mean 7.9 Å) according to a MaxSub [98] superposition. We then identified a same-size (21-member) random subset of the 100 decoys. These have significantly larger pairwise RMSDs, ranging from 8.5 to 17.0 Å (mean 13.5 Å) in a MaxSub super-

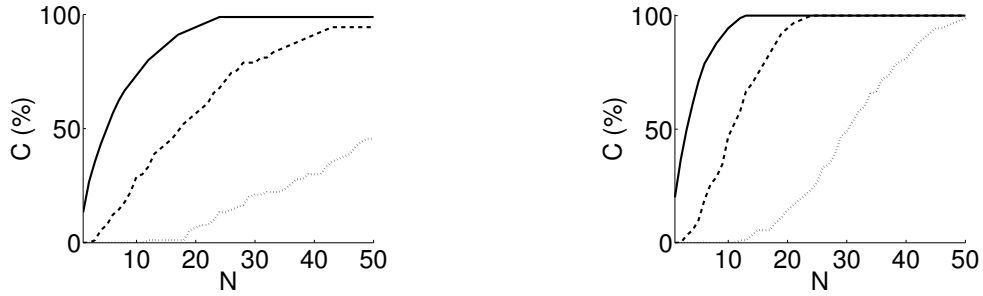


Fig. 3.9: The impact of structural similarity on disulfide experiments planned by XLINK-PLAN. The plots show the relationship between coverage percentage C (%), and number of experiments N, at various discriminability levels (solid: $\Delta = 1$; dashed: $\Delta = 2$; dotted: $\Delta = 4$). (a) A set of 21 similar pTfa decoys, all contained within one of the 100 Rosetta clusters, were used for planning. They have a mean pairwise RMSD of 7.9 Å. (b) A random subset of 21 of the 100 final Rosetta decoys were used for planning. They have a mean pairwise RMSD of 13.5 Å.

position. Fig. 3.9 shows the coverage as a function of number of experiments, at different discriminability thresholds. As would be expected, planning for discrimination of more similar models reduces the coverage achievable for any given number of experiments, but significant coverage is still attainable, even within a cluster of similar models.

The trends in Fig. 3.8 make the diminishing returns in experiment coverage very apparent; *e.g.* for most discriminability levels, moving from 90% to 99% requires a much bigger investment than moving from 50% to 90% (Fig. 3.8(d)). At the same time, as shown by the simulations above (Fig. 3.6), full coverage is generally not required. These two results suggest an efficient semi-sequential experiment planning approach: instead of conducting over 100 experiments (90% coverage with $\Delta = 2$) as the first step, conduct about 20 experiments (25% coverage with $\Delta = 2$) and then plan additional experiments only if the result proves to be ambiguous. Since we seek one model which overrides others, the result of

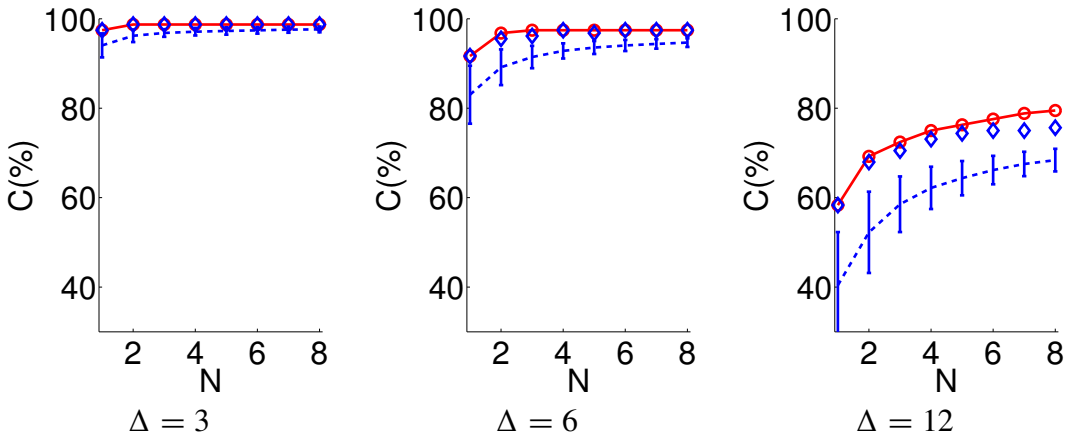


Fig. 3.10: Relative performance of different approaches to planning lysine-specific experiments for FGF-2 model discrimination. Shown are a "planning-free" expectation over 1000 random plans (mean is shown as dashed line with standard deviation error bars), a randomized planning approach (separate circles) considering the best from the set of random plans, and finally XLINKPLAN (solid line).

these experiments could be sufficient to select such a model unambiguously. If additional experiments are required, losing models need not be planned for, thereby pruning the planning problem. As an example, if the results anticipated for threading model 1 were found in the six experiments of the 3-model plan (Tab. 3.4), then 34 of the 103 models would be eliminated according to $\Delta = 2$. This process could be repeated, ending in a final experiment that is explicitly balanced as in the 3-model plan, to discriminate the last few, most similar models.

3.3.4 Algorithmic Considerations

Before this experiment planning method was proposed, investigators might have conducted cross-linking experiments less systematically. We have compared the effects of non-systematic

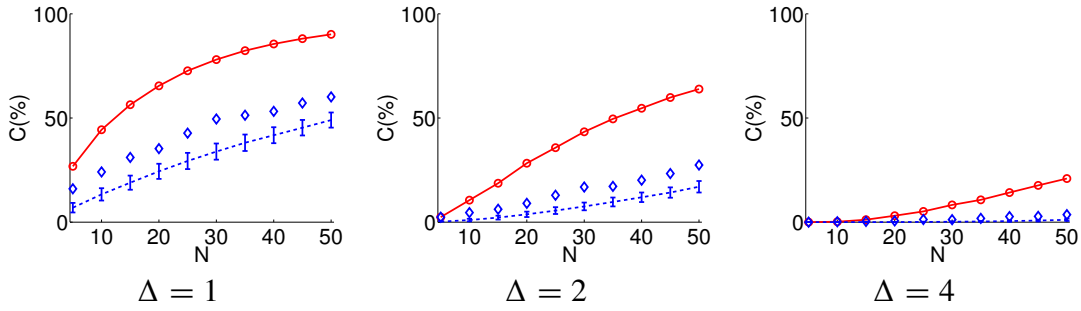


Fig. 3.11: Relative performance of different approaches to planning disulfide trapping experiments for pTfa model discrimination. Shown are a "planning-free" expectation over 1000 random plans (mean is shown as dashed line with standard deviation error bars), a randomized planning approach (separate circles) considering the best from the set of random plans, and finally XLINKPLAN (solid line). Experiments that don't discriminate any model pairs are excluded beforehand.

experimentation with our planning method (Fig. 3.10 and Fig. 3.11). One method would be to simply select experiments without any planning. The expected results and variation of this "planning-free" approach are illustrated by the mean and standard deviation of 1000 random plans. A better alternative, once the problem has been formulated as here, would be to randomly generate sets of plans and select the best. This approach is illustrated by the best of 1000 random plans. Our planning algorithm bests both of these methods, especially with the enormous degrees of freedom and complex restraints of disulfide trapping planned at high Δ. At the same time, our algorithm also achieves balance.

Our planning algorithm (Fig. 3.2) effectively navigates the design space defined by discriminability, coverage, balance, and cost. Its direct encoding of these terms offers advantages over other approaches such as decision trees. For example, multiple coverage of a particular model pair (to attain desired discriminability) is achieved straightforwardly by using initial weights greater than 1, and decrementing them with each covering experiment.

Similarly, balance is achieved automatically by basing our analysis on directed model pairs, $\langle r, s \rangle$ and $\langle s, r \rangle$, and using only positive evidence for planning. Although we use uniform initial weights and weight decrements for all pairs in our algorithm, differential weights and reductions are possible and would provide greater flexibility in trading off among desired criteria for experimental design, either to focus on models of interest or to avoid spending resources on barely distinguishable pairs. While there is additional cost in explicitly considering each pair of models (rather than using a linear-cost metric such as entropy), we have found that the coverage is sparse, with each experiment covering many fewer than the possible quadratic number of model pairs. Thus in practice thousands of models can be handled quite efficiently.

A comparison with the best possible scenario shows that, in practice, our plans propose a number of experiments near the minimum. The best possible scenario has no experimental redundancy (*i.e.* each structure pair is covered by exactly Δ experiments). Therefore the total number of structure pair discriminations must be at least $C \Delta P$ to reach C percentage coverage at discriminability Δ for P pairs. Also under the best possible scenario, each experiment will discriminate a disjoint set of model pairs. This disjointness can be approximated by not considering which model pairs an experiment covers, but only taking the number of expected discriminable pairs. The smallest number of experiments whose expected discriminable pairs sum to the $C \Delta P$ threshold (again, without considering which pairs are covered) defines a lower bound on the optimal experiment number. The plans in Fig. 3.8 are within roughly twice this very simplistic lower bound. Since the true minimum will be greater than this simplistic bound, the selected experiments are well within two-fold

of the optimal number.

Our algorithm balances speed and quality. It takes only seconds on a Pentium 4 computer to generate any of the plans in this chapter, even with reasonably large sets of models and sizes of experiment plans. As previously discussed, the problem is NP-hard, and as we further illustrate, the combinatorics do not permit an exhaustive exploration even for the problem sizes studied here. Yet XLINKPLAN results are well within a factor of two of optimal, and significantly better than a randomized algorithm as the number of degrees of freedom increase.

3.4 Discussion

In this chapter, we have developed a probabilistic mechanism for analyzing cross-linking information with respect to a set of protein structure models, estimating the ability of experiments to discriminate among those models, and optimizing experiments accordingly. A probabilistic framework allows explicit characterization of errors that are present in all experimental data, enabling careful quantification of the extent of support for a particular model. The probabilistic approach allows explicit consideration of the experiment in classical statistical terms of sensitivity and power (type I and type II errors). Under our mechanism, an experimenter can establish and plan for a sufficient level of evidence required to support model selection, and thereby avoid false confidence in committing to an ambiguous decision. Similarly, the ability to select a posterior ratio as well as plan further discriminatory experiments provides control over type II errors.

We employ a small set of readily interpretable parameters to characterize key factors underlying errors in data (κ, ν) and interpretation (H, L). Such parameters remain unstated in other approaches; for example, a violation-counting approach [128] implicitly assumes $\nu = 0$ (no false positives), and $H = 1$ and $L = 0$ (no errors in interpretation of models). Although we adopt the simplest possible forms for these parameters (fixed constant values), we show that they can constitute a rational basis for interpretation and planning. Furthermore, as we found with Fig. 3.6, the results are fairly insensitive to the exact parameter values. In general, cross-link map values would be determined by the reactivity of the protein groups being linked, their accessibility to the cross-linking reagent and the geometric feasibility of the cross-linking reaction given the finite length of the cross-linking molecule. The reactivity of the protein groups cannot be easily extracted from the model, but can be corrected for by measurements of reactivity with monofunctional reagents [78]. For the studies in this thesis, we will assume constant reactivity. Similar considerations hold for accessibility (although some portion of the relative accessibility of sites may be extracted from the predicted model). Finally, geometric feasibility depends on whether or not the cross-linker can bridge the distance between cross-linked atoms in the model, potentially with consideration for protein dynamics. For example, the cross-linker bis-sulfo-succinimidyl suberate (BS^3) reacts with amino groups, including the N-terminus and the N^{ζ} of `Lys` residues, and forms a bridge of up to 11 Å between such pairs. Similarly, in disulfide trapping, disulfide bonds are formed upon oxidation of cysteines whose C^{β} approach within 4.6 Å, with proper geometry, during the experiment.

Our formulation of experiment planning makes explicit the key factors of discriminabil-

ity, coverage, balance, ambiguity, and cost. While the experiments we plan here contain from 10^{11} to 10^{120} combinatorial possibilities, our greedy algorithm is efficient and effective in identifying plans expected to achieve these specified criteria. When confident selection requires planning larger experiments, a proposed semi-sequential approach, conducting batches of experiments that focus on remaining ambiguities, allows researchers to balance the desire for conservative plans with the need for experimental efficiency. This approach can also potentially integrate residue-specific and disulfide cross-linking, once put on common probabilistic ground, using an initial residue-specific experiment to eliminate many models and subsequent disulfide experiments to discriminate remaining ones. Disulfide cross-linking could also readily be supplemented by the use of cysteine-specific cross-linkers operating on the same dicysteine mutants to obtain more distance information [64]. These semi-sequential and hybrid mechanisms are very general, and we will study incorporating different types of experimental data, *e.g.* the combination of cross-linking and mutagenesis (see Chapter 4). Finally, our planner can be applied to additional discrimination problems, for example, selecting among models of protein-protein complexes provided by docking procedures.

Our analysis raises some questions about the value of residue-specific cross-linking, especially when compared with disulfide trapping. If many residue-specific cross-links can be identified in a single experiment, then residue-specific cross-linking can be very powerful. However, whenever residue-specific cross-links are difficult to identify, then our analysis indicates that disulfide cross-linking is a more powerful alternative. We believe a major practical problem, then, is the low and variable capture rate of residue-specific ex-

periments. Even extensive experimentation [128] yielded a capture rate κ of only about $\frac{1}{3}$, while less extensive experiments [48] gave far less ($< 10\%$). New cross-linkers and new detection methods would improve these results, but at present κ is far less than can be achieved with disulfide cross-linking. As a result of the low capture rate, residue-specific experiments effectively provide less information. We estimate that under the current κ and ν, one disulfide cross-link is approximately equivalent to several expected residue-specific ones. In addition, the coverage of residue-specific experiments "saturates" early and dramatically, while disulfide trapping experiments provide enormous degrees of freedom for further, fine-grained model discrimination.

4. MODEL DISCRIMINATION BY STABILITY MUTAGENESIS

As the second part of our PRAXIS approach (Fig. 1.1), this chapter develops methods that exploit the known relationship between protein structure and thermodynamic stability in order to discriminate protein structure models. Our method of "stability mutagenesis" employs stability measurements of planned mutants upon denaturant unfolding to evaluate consistency with predictions made under competing structural models (Fig. 4.1). We first predict changes in unfolding free energy upon mutagenesis, $\Delta\Delta G^{\circ}$, for the same substitution mutations in each atomic model by one of several available methods. Next, experiments are planned to use an efficient set of mutations whose stability changes can be most confidently predicted and that differ most greatly between atomic models. The planned mutations are made in a protein expression system, and the stability of the expressed and purified mutants is determined and compared to wild-type, yielding experimental $\Delta\Delta G^{\circ}$ data. (Throughout this thesis, ΔG° is defined as $G_{unfolded} - G_{folded}$. $\Delta\Delta G^{\circ}$ is $\Delta G^{\circ}{}_{mutant} - \Delta G^{\circ}{}_{wildtype}$. Thus negative $\Delta\Delta G^{\circ}$ indicates destabilization of the wild-type by the mutation.) The consistency between the predicted and the experimental $\Delta\Delta G^{\circ}$ values allows evaluation of the confidence in the models, while the ratio of the confidence allows selection of the model(s) most likely to be correct (relative correctness). The consistency of a

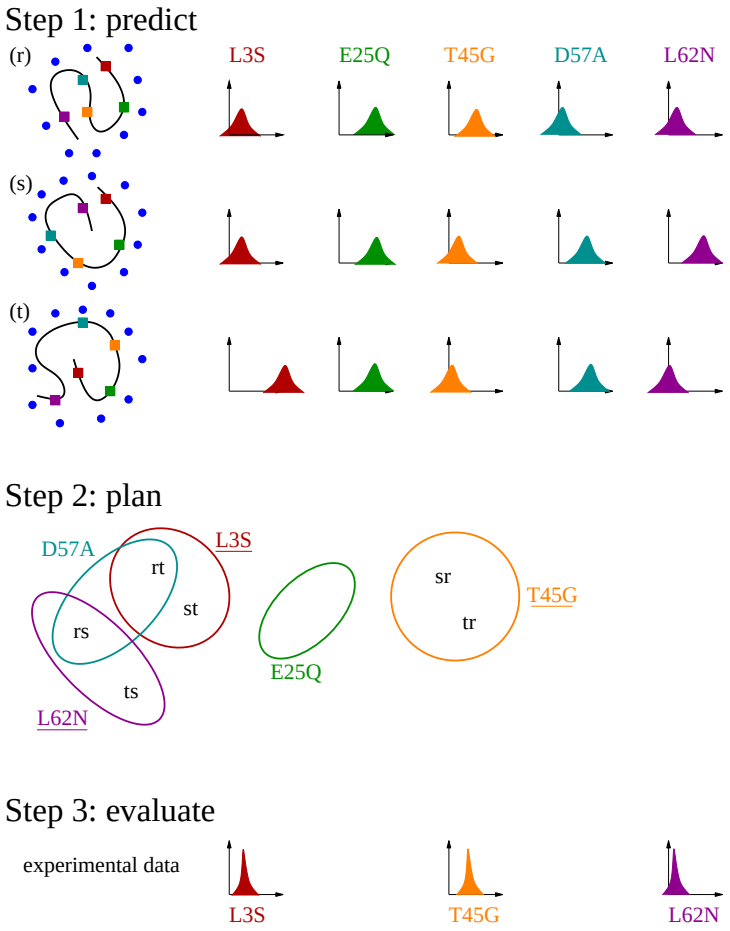


Fig. 4.1: Illustration of planned mutation and stability measurement for discrimination of protein structure models. **Step 1**: $\Delta\Delta G^\circ$ predictions are made for possible mutations (L3S, E25Q, T45G, D57A, and L62N) according to structure models (r, s, t). Mutation L3S is discriminatory, predicted to be significantly more destabilizing in both r and s than in t; in contrast, E25Q has roughly the same effect on each model and is thus not discriminatory. **Step 2**: An experimental plan is optimized by selecting sets of experiments discriminating pairs of models. Schematically, the model pairs discriminated by each mutation are contained within an oval for that mutation. A plan (mutations underlined) should seek to discriminate (cover) all pairs. Good plans should also exhibit a balanced design, so that selection decisions are based upon reliable and representative features of a model, and not on idiosyncratic features or the protein's overall response to mutation or denaturant. **Step 3**: Experimental $\Delta\Delta G^\circ$ data for the selected experiments are interpreted to provide evidence for the models based on consistency of predictions with observations. In the example, the stability measurements are most consistent with the predictions of model r (greatest overlapping area between prediction and measurement).

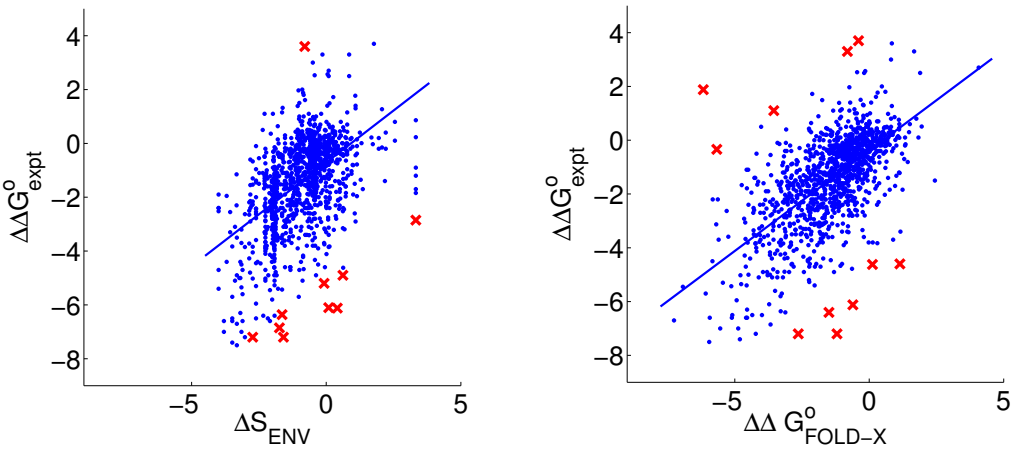
number of diverse mutations with a single model also enhances confidence in its absolute correctness.

We first test our method retrospectively on two proteins, T4 Lysosyme and Staphylococcal Nuclease, which have previously been the object of extensive mutagenesis and stability testing during investigations of the thermodynamics of protein folding. For these two proteins, we demonstrate that the differences between stability predictions based upon crystal structure and threading models are sufficient to discriminate models of distinct folds, when the predictions are evaluated using literature $\Delta\Delta G^{\circ}$ data. While protein structure prediction methods have been increasing in accuracy, the value of experimental discrimination can be seen in the results of the recent CASP6 competition [122]. In the FR/H category, containing proteins whose homology to an existing template could only be determined by knowledge of the structure, the correct fold was identified by a majority of the predictors for only nine of the 22 targets. Furthermore, in the FR/A category, containing proteins analogous to the most similar templates, the correct majority prediction happened for only one of the 16 targets. Thus correct fold recognition templates are frequently available for many sequences, but cannot be reliably distinguished from incorrect ones by current methods. We applied our method on CASP targets, with mutations planned for models and experimental $\Delta\Delta G^{\circ}$ values simulated on x-ray/NMR structures. The simulation shows promising results on targets with high-quality models. Then we present prospective experiment plans for discriminating three high-quality threading models of the bacteriophage lambda pTfa chaperone protein. At the end of this chapter, we compare mutagenesis plans with disulfide cross-linking plans (Chapter 3) and study the effect of combined plans.

4.1 Probabilistic Prediction of $\Delta\Delta G^\circ$

Structure elucidation by stability mutagenesis relies on the accuracy and robustness of $\Delta\Delta G^\circ$ prediction. We employ standard linear regression techniques to determine the relationship between potential score changes and experimental $\Delta\Delta G^\circ$ values. A large number of mutations with experimentally measured $\Delta\Delta G^\circ$ values in the ProTherm database [9] were used to train previously developed $\Delta\Delta G^\circ$ predictors. For the regression, all errors are assumed to be in $\Delta\Delta G^\circ_{\text{expt}}$ and Normally distributed. The regression provides the basis for evaluating the likelihood, indicating the consistency of experimental data with the prediction for structure models. We start with two representative scores that potentially lead to reliable $\Delta\Delta G^\circ$ prediction, the residue environment score ENV and the empirical effective energy potential of FOLD-X (see Sec. 2.4). For ΔS_{ENV} we use the original table of 20 residue types times 18 environment classes that depend upon solvent accessibility, polar fraction, and secondary structure [13]. We use programs NACCESS [54], WHAT-IF [120], and DSSP [57] to determine accessibility, polar fraction, and secondary structure, respectively. For $\Delta\Delta G^\circ_{\text{FOLD}-\text{X}}$, we employ WHAT-IF to model the mutant structures that are used by FOLD-X to calculate the stability of mutant, then compare with and stability of wild-type proteins to get the $\Delta\Delta G^\circ$ values.

Our goal here is to find individual mutations that best discriminate pairs of models. We thus seek reliable individual predictions, and need not use mutations for which we are less confident in their predicted effects. We apply restrictions on allowable substitutions and their locations in the structural models in order to improve the robustness of our ap-



ENV: $R = 0.52$, $\sigma = 1.37$, $y = 0.77x - 0.71$ FOLD-X: $R = 0.66$, $\sigma = 1.21$, $y = 0.75x - 0.38$

Fig. 4.2: Correlation analysis, for 1177 mutations on 74 proteins, between the potential scores, according to the ENV and FOLD-X methods, and the experimentally measured $\Delta\Delta G^\circ$ values. The correlation coefficient R, linear regression function, and standard deviation σ of residuals shown are calculated after removing the outliers (red '×'s).

proach. While applying restrictions reduces the number of possible experimental plans, the remaining combinations should yield more reliable predictions and thus more reliable comparisons between prediction and experiment. We use only *non-augmenting* substitutions, those whose mutant structures are easy to predict because they involve either a substitution to a smaller sidechain (*e.g.* Ile → Val) or direct replacement of atoms (*e.g.* Asp → Asn). These mutants can be easily modeled from the parent structure by WHAT-IF. Non-augmenting mutants form the great majority of mutations in the ProTherm database. Augmenting mutants are more difficult to model and their limited numbers provide insufficient data for training. Cys and Pro are excluded as residues to mutate from or to because of their special structural roles. We used only those evaluated by thermal denaturation, or by chemical denaturation using either urea or guanidine HCl and monitored by either fluorescence or circular dichroism at a temperature between 18 and 25 degrees. The ProTherm database

contains several independent measurements of $\Delta\Delta G^{\circ}$ for the same substitution. For chemical denaturation, the measurement made closest to 20 degrees is selected, followed by the measurement made closest to pH 7. For the remaining competing measurements (the same temperature and pH, or both thermal and chemical denaturant), the independent values are averaged so long as their difference is less than 1 kcal/mol. (If the measured values are greater than 1 kcal/mol, then the measurements are excluded as unreliable.) When the FOLD-X program fails to make a prediction for a mutation, the mutation is excluded from consideration for both ENV and FOLD-X methods, for consistency. For robustness consideration, mutation types with less than 10 instances in the ProTherm database are considered as unreliable and excluded. Starting from the ProTherm database [9], 1177 mutants in 74 proteins satisfied the restrictions and thus serve as the training set for our $\Delta\Delta G^{\circ}$ predictors.

Selected mutants were used to train predictors by linear regression between potential score changes and database $\Delta\Delta G^{\circ}$ values (Fig. 4.2). Outlier points were identified as those whose studentized deleted residuals [28] were greater than 3.0, reflecting approximately deviations more than 3σ from the mean. Similar regression lines were obtained with or without outliers removed; removal of outliers primarily affected σ (not shown). For the regression, all errors are assumed to be in $\Delta\Delta G^{\circ}_{\text{expt}}$ and Normally distributed. The regression provides the basis for evaluating the likelihood, indicating the consistency of experimental data with the prediction for structure model.

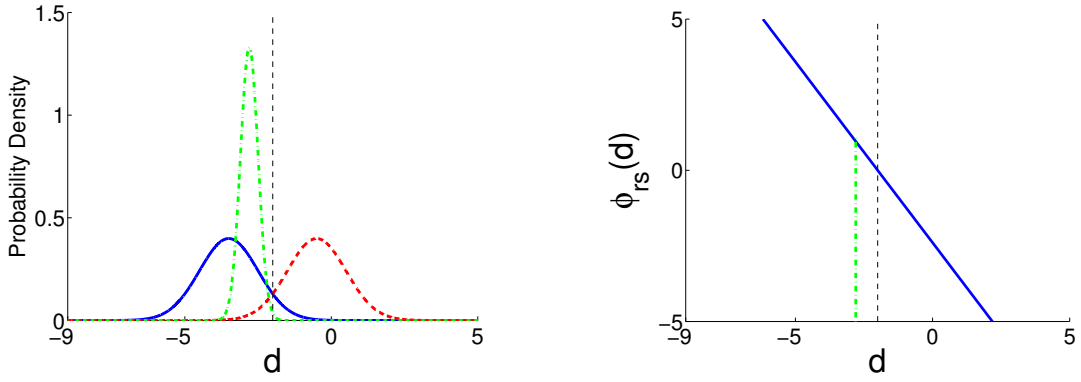


Fig. 4.3: Illustration of a Normal distribution for the $\Delta\Delta G^\circ$ prediction error. (a) $\Delta\Delta G^\circ$ prediction for a mutation on two models, $p(\mathbf{d} \mid r)$ (red dashed) and $p(\mathbf{d} \mid s)$ (blue solid), with means equal to -0.5 kcal/mol and -3.5 kcal/mol and standard deviation 1.0. An example experimental value of -2.8 with an error of of 0.3 is also shown (green dash-dot). The dashed vertical line shows the point at which the data gives no information in favor of one model over the other. (b) Logarithm (base 10) posterior ratio (ϕ_{rs}) of the two models (r over s) given each possible experimentally measured $\Delta\Delta G^\circ$ value.

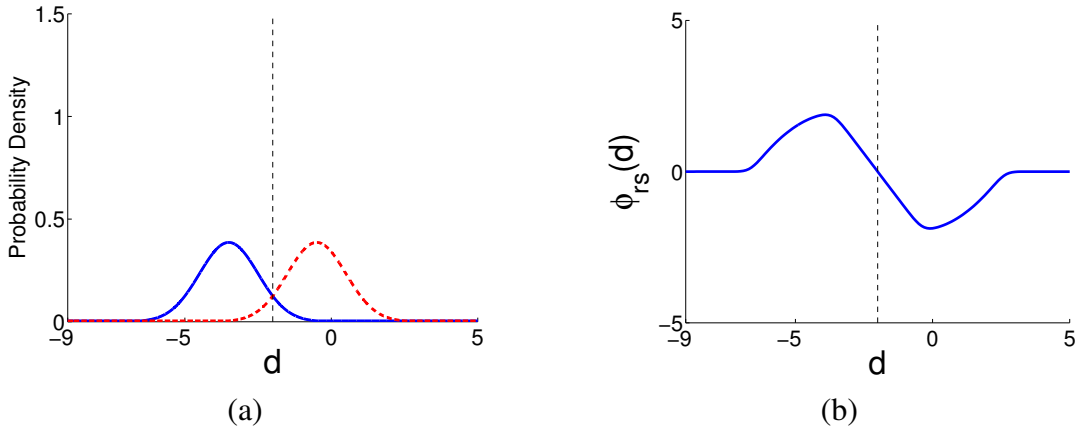


Fig. 4.4: Illustration of a flat-tailed model for the $\Delta\Delta G^\circ$ prediction error. (a) $\Delta\Delta G^\circ$ prediction for a mutation on two models, truncating the two Normal distributions in Fig. 4.3, adding flat tails out to -9 and 5, and renormalizing (the raised tail is hard to see at regular resolution). (b) Logarithm (base 10) posterior ratio of the two models (r over s) given each possible experimentally measured $\Delta\Delta G^\circ$ value.

4.2 Flat-tailed Distribution of $\Delta\Delta G^{\circ}$

The Normal distribution model catches the main body of $\Delta\Delta G^{\circ}$ value, but it is very inaccurate in the tails, and it can cause some problems. Fig. 4.3(a) illustrates calculation of likelihoods for a mutation assuming Normal prediction error. That is, based on the potential score change (ΔS_{ENV} or $\Delta\Delta G^{\circ}_{FOLD-X}$) for a structure model, we use the regression line to derive a mean and σ representing the predicted $\Delta\Delta G^{\circ}$ value and uncertainty. If we assume no experimental error, then with an experimental value $\mathbf{d}$, the likelihood $p(\mathbf{d} \mid r)$ is simply the value of the Normal distribution at $\mathbf{d}$. We do this in retrospective simulation studies, since experimental errors are not available. If we do have experimental error (as with our prospective study), then we integrate the overlapped area, within the allowed limits of -9 to 5 kcal/mol. (An experimental $\Delta\Delta G^{\circ}$ value outside of the $-9, 5$ range is regarded as an error and excluded from model discrimination.) For the example in Fig. 4.3(a), we obtain $p(\mathbf{d} \mid r) \approx 0.305$ and $p(\mathbf{d} \mid s) \approx 0.034$, for a posterior ratio of 9.042.

Fig. 4.3(b) illustrates the posterior ratio of ≈ 0.96 in favor of s over r that results from these likelihoods. The plot also shows the posterior ratios for all possible experimental values. If the predictions from the models have means μ_1 and μ_2 and the same σ, then the posterior ratio is $\exp(\frac{2d(\mu_1-\mu_2)-\mu_1^2+\mu_2^2}{2\sigma^2})$. This is an exponential function of the experimental mean d given the fixed μ_1, μ_2 and σ. Experiments with large positive or negative values interpreted with a Normal distribution can dominate several experiments with less extreme values, and thereby skew the analysis.

In order to obtain more realistic posterior ratios in the presence of extreme values of

d, we modified the Normal distribution to explicitly account for outliers, producing our "flat-tailed distribution" (Fig. 4.4) for approximating the prediction error. This distribution is Normal within $\pm 3\sigma$, but uniform outside that interval, at a value equal to the Normal distribution at 3σ. The values outside $\pm 3\sigma$ approximately match the outlier frequency we observed (Fig. 4.2). The distribution is also truncated beyond the range $-9, 5$ and then renormalized. With this flat-tailed distribution, the posterior ratio from one experiment is bounded at about 100, so the data from a single mutation cannot dominate the others. Furthermore, the ratio approaches 1 if the experimental $\Delta\Delta G^{\circ}$ is in the tails of both distributions, in which case either the experimental value or the predictions or both are most likely outliers and the outcome appropriately does not convey any information to discriminate these two models. This decreasing and ultimately zero confidence in the most extreme values is reflected in Fig. 4.4(b). The calculation of $p(\mathbf{d} \mid r)$ likelihoods (with or without experimental error) proceeds the same as described above for the Normal distribution. For the example in Fig. 4.4 (a), we obtain $p(\mathbf{d} \mid r) \approx 0.296$ and $p(\mathbf{d} \mid s) \approx 0.033$, and thus a posterior ratio of 9.039.

4.3 Evaluation of Mutation Coverage and Utility

Intuitively, among the mutations whose effects can be reliably predicted, those whose predictions are most different between the potential models are the experiments that should be conducted. In this section, we formalize that intuition with definitions of the utility of mutations for discriminating models and their coverage of the discriminations to be made.

In the context of stability mutagenesis, discrimination is based mostly on relative destabilization. (There are relatively few predicted stabilizing mutants.) We run a risk, however, of the experimental data having a consistent bias in the difference between experimental and predicted $\Delta\Delta G^\circ$ values, due to some protein-specific properties. Such a bias could arise from a protein that is relatively easier or harder to destabilize than the proteins against which the predictor is trained. If we selected mutations in which one model was predominantly predicted to be more destabilized than the other, that model would be favored if the protein were relatively easy to destabilize. A good experiment plan needs to achieve a balance in the direction of relative destabilization. We do this by employing the concept of *directed model-pair* that we developed in Chapter 3, *i.e.* separate each pair of models $\{r, s\}$ into two directed pairs $\langle r, s\rangle$ and $\langle s, r\rangle$ and plan separately for these two pairs. By also evaluating the posterior ratios for the directed discriminations separately, consistent bias in interpretation is avoided, reducing the likelihood of making an incorrect decision.

Suppose the distributions of expected $\Delta\Delta G^\circ_{\text{expt}}$ for mutation m on two models r and s are p_r and p_s (with means μ_r and μ_s) respectively. The potential of a mutation m to discriminate the model-pair $\langle r, s\rangle$ can be measured by the expected logarithm of the posterior ratio, which is just the likelihood ratio when the priors are equal. The expected logarithm of the likelihood ratio is given by the relative entropy between the two prediction distributions, denoted as $D(p_r||p_s)$ and $D(p_s||p_r)$. The relative entropy is a standard measure for the difference between two probability distributions; here, greater relative entropy indicates an easier discrimination. The utility of the single mutation m for the directed model-pair

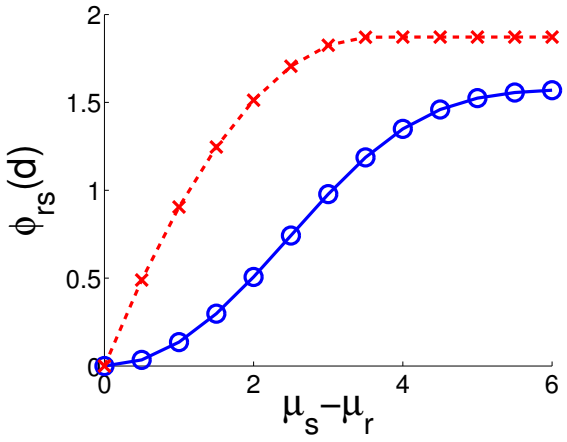


Fig. 4.5: Relationship between the difference of two prediction means and the information provided by mutation, as measured by the mutation utility (solid line, blue circles). Mutation utility is calculated using the flat-tailed distribution with $\sigma = 1.21$ kcal/mol and assuming that the two prediction means μ_r and μ_s are symmetric around the midpoint of the allowed range. Also shown (dashed line, red '×'s) is the maximum logarithm of the posterior ratio under the flat-tailed distribution, the most information we could obtain from an experiment assuming experiments that match predictions perfectly.

$\langle r, s \rangle$ is defined as follows

$$u_{m,\langle r,s \rangle} = I\{\mu_r < \mu_s\} \cdot \frac{D(p_r||p_s) + D(p_s||p_r)}{2}. \tag{4.1}$$

where the indicator $I\{\mu_r < \mu_s\}$ is non-zero (equaling 1) only if $\mu_r < \mu_s$ (model r is more destabilized than model s by mutation m), thus making $u_{m,\langle r,s \rangle}$ a directed criterion.

An intuitive way to assess the significance of $u_{m,\langle r,s \rangle}$ is that the expected posterior ratio between models r and s, $\phi_{rs}(\mathbf{d})$, is $10^{u_{m,\langle r,s \rangle}}$. A directed model-pair $\langle r, s \rangle$ is called "discriminable" by mutation m if $u_{m,\langle r,s \rangle} \geq T$, where $T > 0$ is a user-specified threshold. Fig. 4.5 presents the relationship between $u_{m,\langle r,s \rangle}$ and the difference between prediction means for a mutant under two models. It is sigmoidal, fairly linear when $\mu_s - \mu_r < 4$ kcal/mol (evaluated when σ is 1.21 kcal/mol), but displaying diminishing returns for model discrimination after that. These diminishing returns reflect the influence of the more robust

flat-tailed distribution.

Building on the utility of a single mutation for a single model-pair, we can evaluate the information provided by a single mutation for all model-pairs. The *coverage* c_m measures how many model-pairs mutation m is expected to discriminate (according to the discriminability threshold T discussed above):

$$c_m = \sum_{r,s \in S} I\{u_{m,\langle r,s \rangle} \geq T\} \tag{4.2}$$

Two mutations might have the same coverage, but one might tend to have a greater expected utility in distinguishing the covered model-pairs. Thus we employ as a second criterion u_m, the average utility for a mutation over all discriminable model-pairs:

$$u_m = \frac{1}{c_m} \sum_{r,s \in S} u_{m,\langle r,s \rangle} \cdot I\{u_{m,\langle r,s \rangle} \geq T\} \tag{4.3}$$

4.4 Experiment Planning

After identifying a set of possible mutations that are expected to be reliably predicted and to provide discriminatory coverage for a set of models, we next select a subset of mutations to be conducted, which efficiently discriminate the models. We employ the *set cover* formulation developed in Chapter 3 with coverage as the primary criterion for choosing plans.

$$c_M = \sum_{r,s \in S} I\{(\sum_{m \in M} I\{u_{m,\langle r,s \rangle} \geq T\}) \geq \Delta\} \tag{4.4}$$

where Δ is a user-specified minimum number of distinct experiments that are desired for successfully discriminating the two models, as defined in Chapter 3. Since coverage is

an integral number, many plans may have the same c_M. The average expected utility u_M provides a useful tie-breaker, defined by:

$$u_M = \frac{\sum_{m \in M} c_m \cdot u_m}{\sum_{m \in M} c_m} \cdot 2\Delta \tag{4.5}$$

The fraction in Eq. 4.5 is the average of $u_{m,\langle r,s \rangle}$ over all selected mutations and all discriminable model-pairs, and the 2Δ factor is the desired number of mutations discriminating each pair of models $\{r, s\}$ (Δ in each direction). u_M provides an intuitive way to assess the expected posterior ratio between the correct model and a wrong one after measuring the covering mutants in an experimental plan. The expected posterior ratio is approximately 10^{u_M} if both directed model-pairs are covered by exactly Δ mutations. Additional metrics for finding a plan that is balanced and robust to experimental idiosyncrasies are discussed in the next section.

If the numbers of candidate mutations and models are small, then a brute force method suffices to find the best experiment plan. In this case, we simply enumerate all possible experimental plans (sets of mutations) of the desired size, and evaluate each set according to c_M and u_M. If the number of combinations becomes prohibitive, we can employ our greedy algorithm for multiple-coverage set-cover, which has been demonstrated to yield high-quality plans in the case of planning cross-link experiments (see Chapter 3).

4.4.1 Robustness Considerations

We have already applied restrictions on mutations in order to obtain the most reliable $\Delta\Delta G^\circ$ prediction (Sec. 4.1). Furthermore, in order to choose mutations that are most ro-

bust to model discrimination, we apply some additional restrictions on selected mutations. We first exclude the N-terminal residue of the mature protein, whether methionine or not, to avoid interference with translation or N-terminal processing. We can also apply model-specific restrictions. We have manually excluded from consideration stretches of residues in each model that are potentially poorly predicted in our example of three pTfa models. For a larger number of models, automated criteria are required to exclude poorly modeled residues such as those based with low contact orders.

Mutations that are predicted to be substantially stabilizing are excluded. While stabilizing substitutions are known, and the structural origins of their stability have been investigated for some [71], these substitutions are generally quite rare, and only 1% of all mutations investigated in the training set are substantially stabilizing ($\Delta\Delta G^{\circ} >$ 2 kcal/mol). Because substitutions that enhance stability are relatively rare, there are not sufficient numbers of them in the database to assess the accuracy of the methods for predicting their stability changes. Thus we have excluded any mutation which is predicted to enhance stability by more than a nominal amount of 2 kcal/mol.

Mutations predicted to be extremely destabilizing, reducing stability by more than 6 kcal/mol, are also excluded. These extremely destabilizing mutations are also poorly represented in the database. In addition, mutations that have an extreme effect on structure may alter the native or denatured state, making it difficult to compare their ΔG° values with wild-type. Note that as an additional protection against artifacts arising from substantial changes in the native or denatured state, we also recommend excluding from the analysis any mutation whose observed m-value (slope of the denaturation curve) changes

by more than 15% from wild-type.

The planning approach already seeks robustness in a plan by employing directed model-pairs and optimizing for coverage of all the directed model-pairs. Furthermore, in order to ensure that the plan does not focus too much on local aspects of a model but is evaluating its overall correctness, we also enforce a level of spatial balance: at most one mutation at each position can be selected, and no two mutations in the same plan can be within 5 Å (C^{β} to C^{β} distance). Finally, while not employed here, experimenters might desire to select mutations from different substitution types or categories (*e.g.* nonpolar to polar) to reduce the reliance on specific parameters of the empirical energy model of the $\Delta\Delta G^{\circ}$ predictor.

4.5 Results

4.5.1 Retrospective Testing

To our knowledge, this is the first work studying the effectiveness of mutagenesis and stability measurement as information for discriminating predicted atomic models. To validate our method, we performed several retrospective analyses, discriminating crystal structures and fold recognition models of distinct folds using mutations available in the ProTherm database [9]. We studied several proteins that have many mutations with experimentally measured $\Delta\Delta G^{\circ}$ values, and present the results with T4 Lysozyme and Staphylococcal Nuclease. Predicted models were obtained via the protein fold-recognition meta-server [63]. Models containing less than 90% of the sequence or having an RMSD vs. the crystal struc-

ture greater than 20 Å were ignored.

We first identified reliable discriminatory mutations with available experimental $\Delta\Delta G^\circ$ values and m-value changes less than 15% from wild-type. Then we selected the most informative (maximum u_M) balanced set for each model vs. the crystal structure. At most one mutation was selected for each position. Posterior ratios were computed with these mutants using the database values for $\Delta\Delta G^\circ_{\text{expt}}$ (no experimental σ values were available) and our flat-tailed distribution (Sec. 4.2) of prediction error.

Tab. 4.1 illustrates discrimination of one predicted model from the crystal structure of T4 Lysozyme. We can see that most of the database $\Delta\Delta G^\circ$ values are significantly closer to those predicted from the crystal structure than to those predicted from the model, yielding a posterior ratio approaching 1 to 4000 against the model. The ⟨model, xtal⟩ comparison contributes $0.014 \times 0.72 \approx 0.0098$, disfavoring the model relative to the crystal structure. In the ⟨xtal, model⟩ comparison, one mutation actually slightly favors the model (1.76), but the other mutation strongly disfavors it (0.015), for a total of 0.026, again disfavoring the model. Therefore we can select the crystal structure over the predicted model, although more mutations (or more discriminatory ones than found in the database) might be necessary to increase our confidence in doing so.

Fig. 4.6 shows a set of more extensive tests for T4 Lysozyme and Staphylococcal Nuclease. We first considered the effect of the number of mutations, and show discrimination results between crystal structures and models for (a) $\Delta = 2$ (4 mutations) and (b) $\Delta = 4$ (8 mutations). Although the selection of mutations is limited in the database, Fig. 4.6(a) and (b) nonetheless demonstrate the effectiveness of using stability mutagenesis for model

Tab. 4.1: Illustration of model discrimination for T4 Lysozyme (comparing model *fugue-1fch-A*[(a)] vs. crystal structure *2lzm* (RMSD = 13.09 Å)

mutation	$\Delta\Delta G^{\circ}_{\text{model}}$	$\Delta\Delta G^{\circ}_{\text{xtal}}$	u_m	$\Delta\Delta G^{\circ}_{\text{expt}}$	$\frac{p(\text{model} \mid \text{expt})}{p(\text{xtal} \mid \text{expt})}$
N163D	-4.01	-0.45	1.21	-0.21	0.014
L39A	-1.95	-0.57	0.26	-0.90	0.72
⟨model,xtal⟩ subtotal					0.0098
L84A	-0.25	-3.34	1.02	-3.90	0.015
I3G	-0.84	-3.66	0.90	-1.95	1.76
⟨xtal,model⟩ subtotal					0.026
Overall					2.6×10^{-4}

[(a)] Model *fugue-1fch-A* was derived from threading program FUGUE (Ver2.0) [97] using as template the C-terminal TPR region of Peroxisomal Targeting Signal 1 Receptor (pdb id: 1fch), which is in a different fold (alpha-alpha superhelix) from T4 Lysozyme (Lysozyme-like).

discrimination when structures are substantially different, and also indicate the sensitivity of this approach. As we can see, models with large RMSD from the crystal structure can be discriminated when using an appropriate number of experiments, while models within 5 Å are rarely discriminable. With RMSD < 5 Å corresponding to models of the same fold, this demonstrates that models of different folds are almost always discriminable, while models within the same fold are almost never discriminable. The confidence of choosing the crystal structure increases with the number of experiments (compare Fig. 4.6(a) and (b)), indicating the trade-off between information gain and experimental effort. Since the selection of mutations is limited to those with database $\Delta\Delta G^{\circ}$ values, the effectiveness of discrimination shown here can be regarded as a lower bound of that which could be obtained when the best mutations can be selected and conducted prospectively.

To evaluate the performance of this approach in practice, when no crystal structure would be available, we also tested discrimination vs. a predicted model of the correct fold,

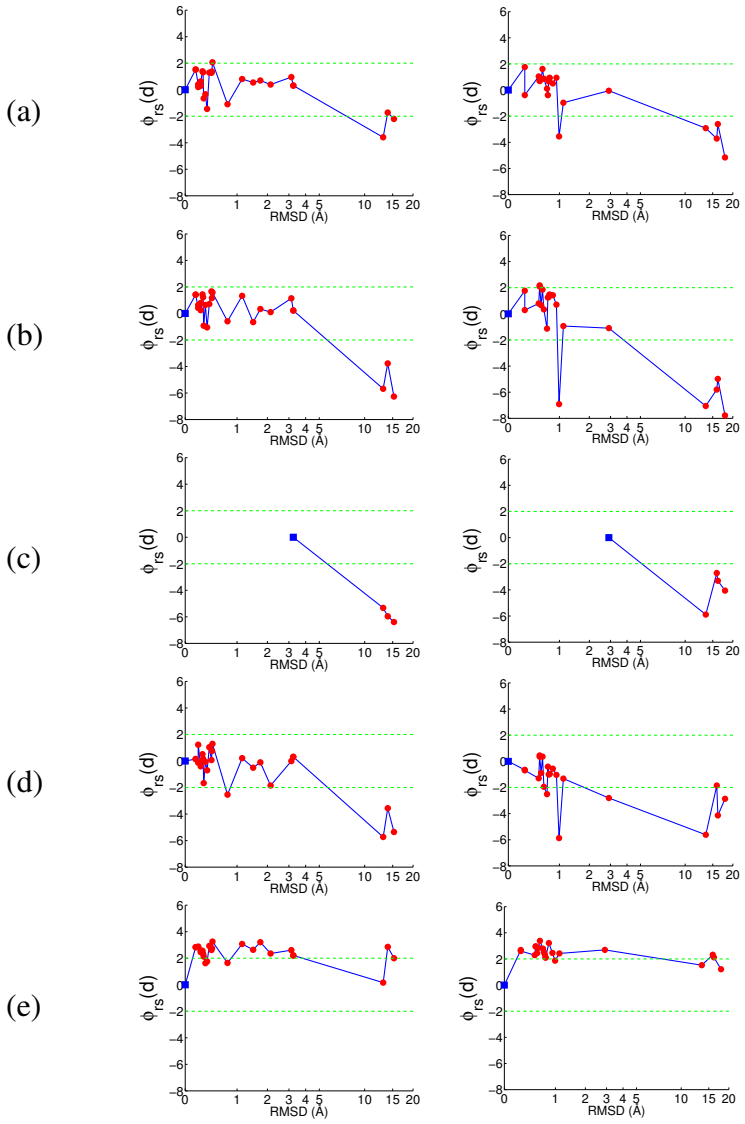


Fig. 4.6: Retrospective model discrimination using mutations in the ProTherm database for T4 Lysozyme (left; 84 mutations considered) and Staphylococcal Nuclease (right; 240 mutations considered). The logarithm (base 10) of the posterior ratio of the predicted model over the reference crystal structure or model of the same fold is plotted. An indiscriminable region $-2, 2$ (posterior ratios less than 100-fold) is indicated by the green dashed lines; models within the region are considered indiscriminable from the reference structure/model with the given data, while those below the region are disfavored relative to the reference structure/model. In rare cases, points above the region indicate that the model is favored over the reference structure/model. (a,b) Crystal structure vs. models with (a) 4 and (b) 8 mutations. (c) Worst model of the correct fold vs. other models. (d,e) Crystal structure vs. models, with constant bias added to the data, using either a (d) balanced or (e) unbalanced plan.

Fig. 4.6(c). To create a conservative test, we chose the model with the highest RMSD to the crystal structure. The results illustrate that even in situations where models are being compared with other models, the model of the correct fold will be chosen.

Finally we tested the value of a balanced plan (directed model pairs) in overcoming potential systematic bias in experimental $\Delta\Delta G^\circ$ values. Fig. 4.6(d) shows the discrimination result of a balanced $\Delta = 4$ plan when all experimental $\Delta\Delta G^\circ$ values have been biased to simulate an easily destabilized protein, using an offset of -1.21 kcal/mol (the prediction uncertainty). Even with biased data, balanced plans robustly select the crystal structure over models with distinct folds, with appropriately reduced confidence. In contrast, Fig. 4.6(e) shows the discrimination result of an unbalanced plan (in which all 8 selected mutations are more destabilizing in the predicted model than in the crystal structure), tested with the same biased $\Delta\Delta G^\circ$ values. In this situation, models of correct or incorrect fold are sometimes selected over the crystal structure. The balanced plan is thus much more robust to systematic error in $\Delta\Delta G^\circ$ values.

4.5.2 Simulation on CASP Targets

The retrospective testing shows that stability mutagenesis is most useful for discriminating models of different folds. To evaluate the performance of our approach on larger sets of high quality, yet different models, we also tested it on Fold Recognition targets from CASP6 [122]. For a fair comparison between predicted $\Delta\Delta G^\circ$ values for models and those for x-ray/NMR structures, only single-domain targets are selected. For each of the

Tab. 4.2: Selected FR targets from CASP6. The number of residues is that in the target sequence, which may be different from that in the x-ray/NMR structure. For example, residues 1-23 are missing in the NMR structures of target T0215. The third column shows both the number of mutations and, in parentheses, unique positions. The number of models is limited to those that passed our filters.

target	#residue	#mutation	#model	category	method	pdb id	description
T0213	103	252 (93)	63	FR/H	NMR	1te7	Hypothetical protein, E. coli
T0214	110	252 (96)	55	FR/H	NMR	1s04	Hypothetical protein, P. furiosus
T0215	76	115 (49)	75	FR/A	NMR	1x9b	Hypothetical membrane protein, T. acidophilum
T0224	87	199 (77)	66	FR/H	NMR	1rhx	TM0979, T. maritima
T0263	101	201 (84)	62	FR/H	x-ray	1wd6	Hypothetical protein, E. coli
T0281	70	129 (53)	67	FR/A	x-ray	1whz	Hypothetical protein, T. thermophilus

selected targets, we chose the first model provided by each prediction group (labeled "1"), and excluded models that were unrefined (labeled "u"), without sidechains, segmental or incomplete, or raised exceptions under WHAT-IF or FOLD-X. Target T0206 was excluded because only the x-ray structure of the C-terminal domain was available; target T0212 was excluded because WHAT-IF reported a fatal error on the x-ray structure (pdb id: 1tza).

In order to satisfy our "closed world" assumption (see Sec. 4.6 for more discussion), we only choose targets with high quality models. Specifically, we require that at least one model has rmsd (of all C_α atoms) within 5Å of the x-ray/NMR structure. Tab. 4.2 lists the 6 selected targets. Mutations were planned by our greedy algorithm at discriminability threshold $T = 0.7$ and coverage threshold $\Delta = 1$, and $\Delta\Delta G^\circ$ values were simulated using the x-ray/NMR structures. Models were evaluated against the model with the highest posterior probability and those with log posterior ratios of at most -2 were selected. In other words, a model was discarded if it was 100 fold worse than some other model. In order to evaluate the performance of our approach, Fig. 4.7 presents the number of models

selected and the average GDT_TS z-score (a relative measurement of the similarity between model and the corresponding x-ray/NMR structure) [131] of these models w.r.t. a varying number of mutations used for discrimination. As we can see, in all six targets, the average z-score of the top models is consistently increasing with the number of mutations. The median z-score increases roughly in parallel with the mean z-score (not shown). The model with the highest z-score is included in the selected group for four of six targets (T0215, T0224, T0263 and T0281), and the model with the second highest z-score is included for the other two (T0213 and T0214). Fig. 4.7 definitely shows non-random selection of models by planned mutations, although the GDT_TS score is not a hard criterion for model quality.

4.5.3 Prospective Experiment Planning for pTfa

We put our planning mechanism into practice on the pTfa protein of bacteriophage lambda. We use the same three high-quality threading models as in Chapter 3: *r* (template: chaperone DnaK substrate binding domain, pdb id 1dkz), *s* (template: heme chaperone Ccme, pdb id 1liz), and *t* (template: mRNA capping enzyme, pdb id 1ckm).

There are altogether 2052 possible substitutions, 19 at each of 108 positions. Our restrictions immediately excluded Cys, Pro and Met1, poorly modeled regions (residues 97–108 in the two OB-fold models), augmenting mutations, and mutation types poorly represented in the ProTherm database. The restrictions excluding mutations with extreme $\Delta\Delta G^\circ$ predictions (predicted $\Delta\Delta G^\circ$ outside of the range $-6, 2$) had very little impact in

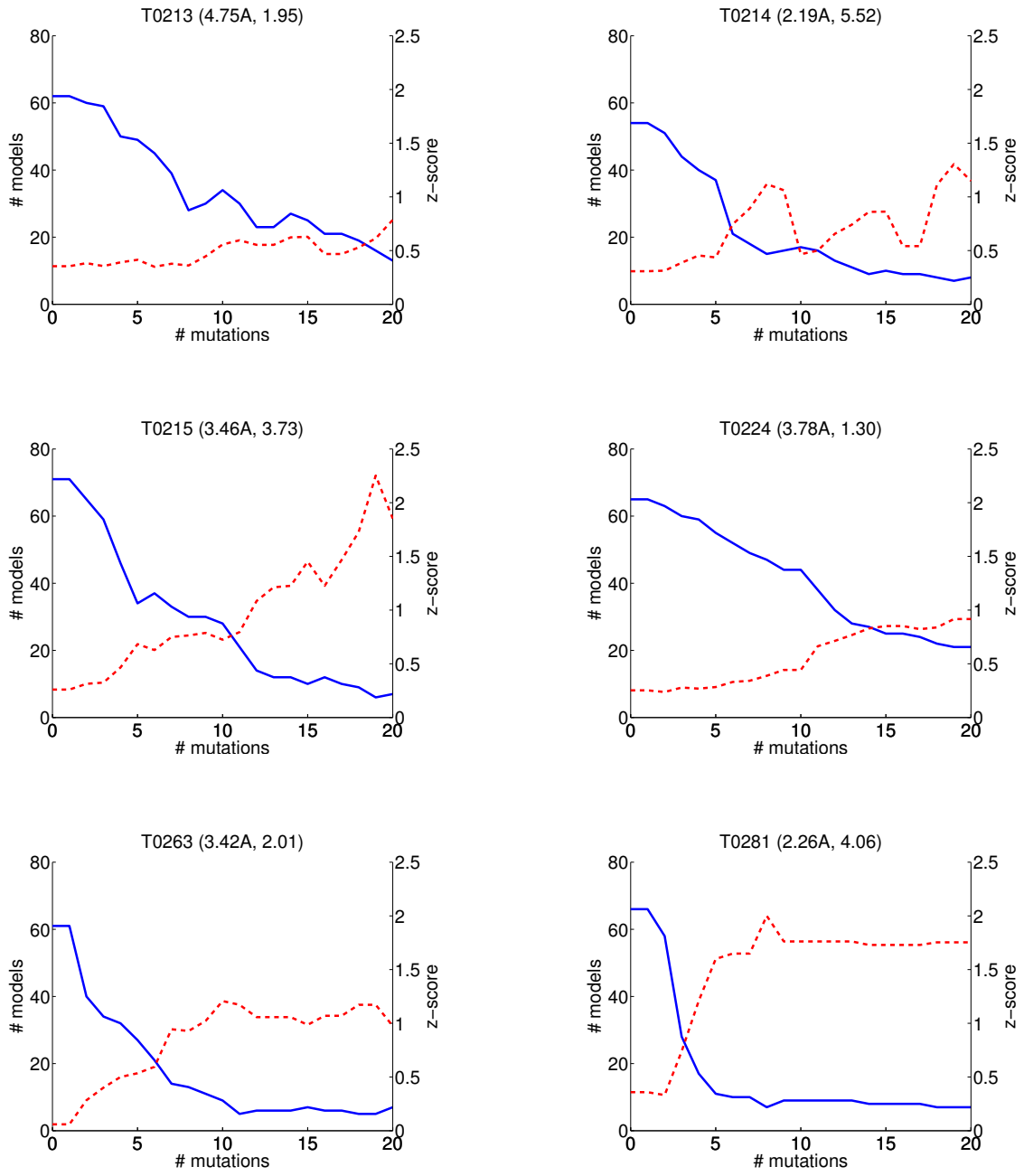


Fig. 4.7: Model discrimination for targets in Tab. 4.2. The number of models selected (blue solid line) and the average GDT_TS z-score of these models (red dashed line) are shown w.r.t. a varying number of mutations used for discrimination. The target number is shown on the top of each plot, followed by the smallest rmsd of a model to the x-ray/NMR structure and the highest z-score among all models (these numbers are not necessarily from the same model).

this case. After applying all these restrictions, we were left with 192 possible mutations at 77 positions. The user-specified threshold T sets a minimum log posterior ratio; a mutation whose expected log posterior ratio for a model-pair exceeds this threshold is considered likely to discriminate the model-pair. With $T = 0.3$ (corresponding to an expected posterior ratio of about 2), we have 28 informative mutations at 21 positions (Tab. 4.3). Plans were developed by the brute-force algorithm, and the top plan (maximizing c_M and u_M) is presented in Fig. 4.8. In fact, each of the mutations in the top plan has $u_m > 0.6$, corresponding to an expected posterior ratio of about 4.

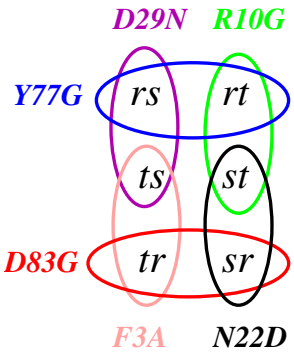
In order to assess the effectiveness of the six selected experiments in Fig. 4.8, we conducted three sets of simulations of the experimental outcomes, assuming in turn that each of the three models was correct. The simulated $\Delta\Delta G^{\circ}_{\text{expt}}$ values were randomly drawn from the prediction distributions according to the model assumed to be correct, and error added according to an experimental error distribution (0.3 kcal/mol in the initial simulation). We varied the threshold θ for the posterior ratio, choosing a model over another one if and only if the posterior ratio exceeded 10^{θ} (otherwise the models were considered indiscriminable). If we selected the postulated model as the winner, the decision was counted as a correct one; if either of the other two was selected, it was counted as an error. We calculated frequencies of making correct and incorrect decisions, averaged over the three models with 10, 000 runs each.

Fig. 4.9(a) illustrates the ability of the plan to identify the correct model under different thresholds. Consider first the case of standard experimental error of $\sigma = 0.3$ kcal/mol (blue solid lines). At $\theta = 1$, the correct model would have about a 90% chance to be selected

Tab. 4.3: The informative mutations for discriminating three pTfa models, after applying restrictions and a threshold $T = 0.3$.

index	mutation	c_m	u_m	$\Delta\Delta G^{\circ}_{\text{model}}$			coverage pattern						rank$^{(a)}$
				r	s	t	rs	rt	sr	st	tr	ts	
1	F3A	2	0.86	0.15	-0.09	-2.72	0	0	0	0	1	1	3
2	R10A	2	0.45	-0.99	-0.60	1.07	0	1	0	1	0	0	26
3	R10G	2	0.79	-1.35	-0.84	1.50	0	1	0	1	0	0	7
4	T11A	2	0.33	0.19	0.06	-1.45	0	0	0	0	1	1	28
5	T11G	2	0.59	-0.56	0.19	-2.35	0	0	0	0	1	1	13
6	K13G	2	0.53	0.64	-1.07	1.30	0	0	1	1	0	0	17
7	N16D	2	0.67	-0.71	-2.78	-0.17	0	0	1	1	0	0	9
8	N22D	2	1.22	0.69	-3.27	0.02	0	0	1	1	0	0	1
9	I25A	2	0.51	-2.68	-0.24	-1.15	1	1	0	0	0	0	21
10	I25G	2	0.82	-3.92	-0.79	-1.72	1	1	0	0	0	0	5
11	E27G	2	0.46	-0.54	1.02	-1.17	1	0	0	0	0	1	24
12	D29A	2	0.34	-0.36	1.26	-0.35	1	0	0	0	0	1	27
13	D29N	2	0.63	-0.90	1.62	-0.38	1	0	0	0	0	1	11
14	D29G	2	0.56	-1.42	1.01	-0.79	1	0	0	0	0	1	14
15	I32G	2	0.53	-0.06	-2.47	-1.71	0	0	1	0	1	0	16
16	L38G	2	0.49	-0.39	-2.27	-0.22	0	0	1	1	0	0	23
17	N41D	2	0.80	-2.82	0.13	-0.52	1	1	0	0	0	0	6
18	V55S	2	0.53	-0.17	-0.83	-2.53	0	0	0	0	1	1	19
19	V57S	2	0.50	-0.77	0.75	-1.65	1	0	0	0	0	1	22
20	E62G	2	0.54	1.25	1.29	-0.81	0	0	0	0	1	1	15
21	E69Q	2	0.53	1.82	-0.28	-0.16	0	0	1	0	1	0	18
22	T75A	2	0.63	-2.33	0.11	-0.23	1	1	0	0	0	0	10
23	T75G	2	0.86	-2.99	-0.75	0.27	1	1	0	0	0	0	4
24	T75S	2	0.52	-2.34	-1.77	-0.03	0	1	0	1	0	0	20
25	Y77G	2	1.07	-3.27	0.23	-0.33	1	1	0	0	0	0	2
26	V79S	2	0.60	-2.46	-2.94	-0.50	0	1	0	1	0	0	12
27	D83G	2	0.77	1.74	-1.30	-0.35	0	0	1	0	1	0	8
28	E89Q	2	0.45	-0.48	1.07	-1.11	1	0	0	0	0	1	25

$^{(a)}$ In lexicographic order of c_m and u_m.



mutation	c_m	u_m	$\Delta\Delta G^\circ_{\text{model}}$		
			r	s	t
F3A	2	0.86	0.15	-0.09	-2.72
R10G	2	0.79	-1.35	-0.84	1.50
N22D	2	1.22	0.69	-3.27	0.02
D29N	2	0.63	-0.90	1.62	-0.38
Y77G	2	1.07	-3.27	0.23	-0.33
D83G	2	0.77	1.74	-1.30	-0.35

Fig. 4.8: Full coverage plan ($c_M = 6$) at discriminability $\Delta = 2$ for discriminating three pTfa models, with discrimination utility $u_M = 3.56$.

(upper blue solid line) and an incorrect model would have about a 1% chance (lower blue solid line), while the remaining 9% of the time, we would not be able to make a decision. At $\theta = 2$, there is very little chance of choosing the wrong model, but the probability of making a correct decision is reduced to around 75%.

With modern methods, determination of $\Delta\Delta G^\circ_{\text{expt}}$ can be done with an accuracy of < 0.3 kcal/mol. Measurements of this quality, however, are dependent upon obtaining true equilibrium values from well-behaved samples. Since this will not be possible for all samples of interest, we have simulated the effects of lower quality measurements, by increasing the simulated experimental error from 0.3 to 0.6 and 1.2 kcal/mol. As Fig. 4.9(a) shows, the frequency of making a correct decision is reduced from 90% with accurate measurements to around 75% at $\theta = 1, \sigma = 1.2$, and from 75% to 54% at $\theta = 2, \sigma = 1.2$. The error frequency increases from 1% to 5%, and from almost 0% to 2%, respectively. While the highest quality data allow for more powerful discrimination, even low quality data can be useful combined with an appropriate plan.

Fig. 4.9(b) illustrates the importance of carefully planning experiments. We generated

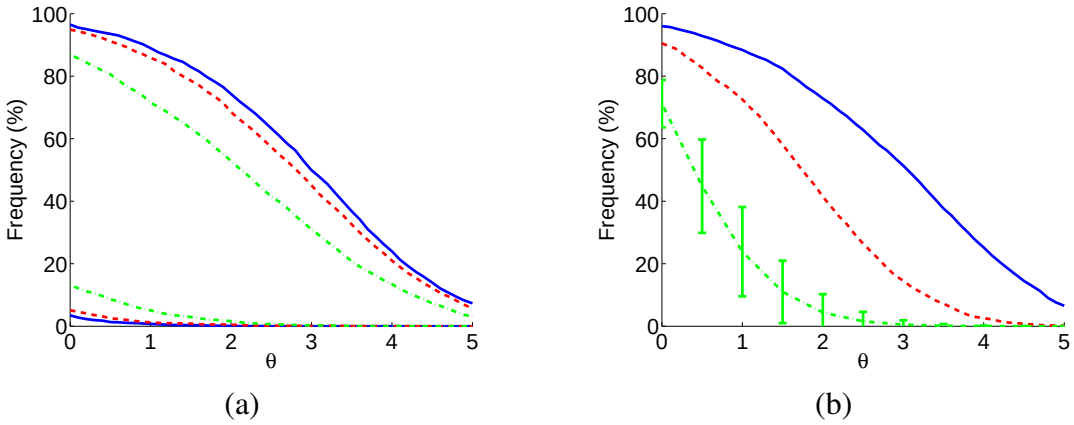


Fig. 4.9: Simulation of discriminating three pTfa models. (a) Frequency of correct decision (top three curves) and incorrect decision (bottom three; the remainder are rejections) for the top plan (Fig. 4.8), with experimental error of 0.3 (blue solid), 0.6 (red dash), and 1.2 (green dot-dash) kcal/model. (b) Frequency of correct decision for the top plan (blue solid) vs. 1000 randomly chosen 6-experiment plans (green dot-dash shows the mean, with bars indicating one sigma variation; red dash shows the best out of the 1000).

1000 random plans by selecting from the set of 192 allowed mutations, and computed the average and best discrimination results for all plans using the same simulated data. The planned experiment is always significantly more effective than even the best random plan. In additional simulations (not shown), we found that it takes 24–30 random mutations to provide results of the same quality as the optimal 6-mutation plan. We note here that we have already improved the experiment over completely random by considering 192 mutations; a truly random method would be selecting from the entire set of 2052 mutations, and thus would be expected to do much worse.

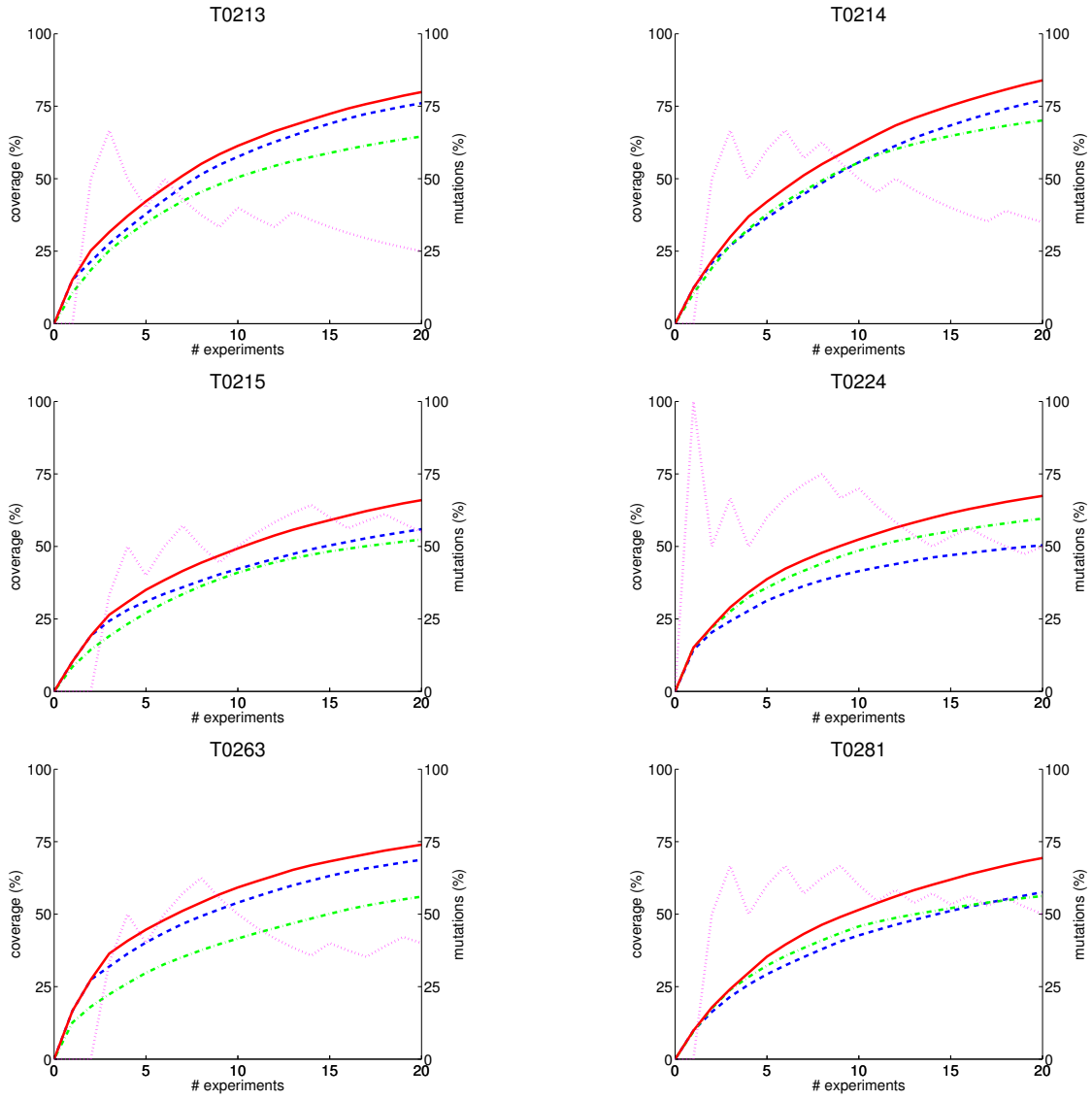


Fig. 4.10: Coverage of greedy plans ($\Delta = 1$) on CASP targets, using cross-linking (blue dashed line), mutagenesis (green dash-dotted line), or both (red solid line). The magenta dotted line (y-axis on the right) shows the percentage of mutations among selected experiments in the combined approach.

4.5.4 Multimodal PRAXIS

We have demonstrated that both cross-linking and mutagenesis can be employed to discriminate predicted protein structure models. It is interesting to see which experimental technique, or combining both, is more informative for model discrimination. Our analysis in Chapter 3 shows that disulfide trapping is probably more valuable than residue-specific cross-linking for model discrimination, because it has higher capture rate, lower noise rate and more freedom to choose experiments (residue pairs). It is more difficult to compare mutagenesis and cross-linking directly because it is hard to determine the relative significance of the information provided by these two experiments. If we use the same parameter values for cross-linking as in Fig. 3.6 (*i.e.* $H = 0.9$, $L = 0.1$, $\kappa = 0.95$ and $\nu = 0.05$), the expected posterior ratio contributed by one cross-link is 5.31. On the other hand, the expected ratio contributed by one mutation is by construction at least 10^T, where T is the discriminability threshold (see Sec. 4.3). If we set $T = 0.7$, we also have an expected ratio around 5. In other words, under these parameters, single model-pair coverage provided by cross-linking and mutagenesis are roughly equivalent, at least in terms of expected posterior ratio. Thus we can use the coverage as a consistent criterion to evaluate plans including cross-links and mutations.

Fig. 4.10 presents the coverage of greedy plans on CASP targets (Tab. 4.2), using cross-linking or mutagenesis alone, or combined. We allow at most one mutation to be selected from each position. Restrictions about cross-linking sites can also be applied although we allow free selection in this analysis. As expected, the combined plans always outperform

single-mode plans, either cross-linking or mutagenesis alone. A combined plan has more available degrees of freedom to choose experiments and delays the "diminishing returns" seen under either experimental type alone. Furthermore, all the plans are well mixed, *i.e.* comparable numbers of cross-links and mutations are selected, implying that cross-linking and mutagenesis are both important in reaching high coverage, at least under the current parameters.

We also observed that cross-linking provides higher coverage than does mutagenesis on most of the targets, except for T0224. This is probably due to the fact that cross-linking has more candidate experiments (residue pairs) than mutagenesis (point mutations). However, this result does depend on the parameters; *e.g.* if the capture rate and noise rate are set as $\kappa = 0.85$ and $\nu = 0.15$, the expected ratio contributed by one cross-link would be around 3, comparable to mutagenesis with $T = 0.5$. In this case, the coverage of mutations would be significantly increased, and mutagenesis would outperform cross-linking on all targets (not shown).

In order to simulate the real performance of the selected plans in model discrimination, rather than the *a priori* coverage, we conducted the same simulation as in Fig. 4.7, for each of cross-linking, mutagenesis, and combined. To simulate the cross-linking data, we evaluated the $C_\beta - C_\beta$ distances in the x-ray/NMR structures, and considered the cross-link to be observed for a distance less than 11 Å, not observed for a distance greater than 17 Å, and discarded for a distance within the range of $11, 17$ Å, *i.e.* that cross-link is not used to support or penalize any model. Fig. 4.11 presents the average GDT_TS z-score of selected models by each type of plan. The average z-score consistently increases

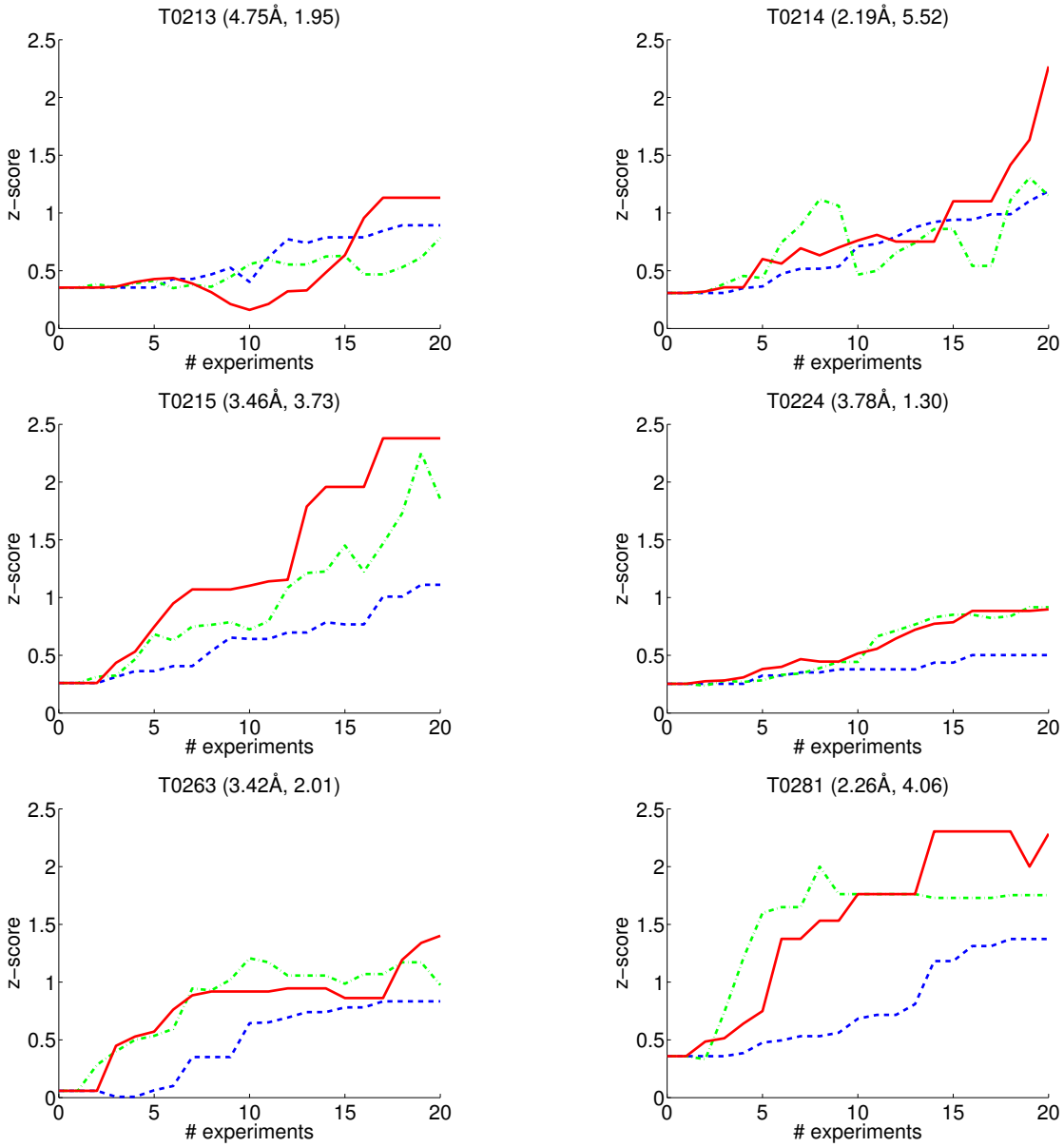


Fig. 4.11: Average GDT_TS z-score of selected models w.r.t. the number of experiments: cross-linking (dashed blue), mutagenesis (green dash-dotted), and combined (red solid).

with the number of experiments in all three approaches. Mutagenesis tends to outperform cross-linking, although the coverage is lower (see Fig. 4.10). This is probably due to the partial information in continuous mutagenesis data that were not taken into account in the calculation of coverage. This motivates our development in the next chapter of criteria to evaluate continuous data. The more important observation in Fig. 4.11 is that, with a sufficient number of experiments, the combined approach almost always exceeds single-mode approaches.

The choice of parameters is only illustrative and we have not take into account the experimental cost. However, these results nevertheless demonstrate the merit of a multi-modal approach, *i.e.* it provides more information with the same number of experiments. Furthermore, the consistency of multiple modes of experimental data (*e.g.* cross-linking and mutagenesis) with a single model also enhances confidence in its correctness. See Chapter 7 for more discussion of a multimodal approach for model discrimination.

4.6 Discussion

Employing a variation of the framework developed for cross-linking, we demonstrated with retrospective testing on two proteins and simulation on CASP targets, the effectiveness of stability mutagenesis for model discrimination. We also demonstrated with simulated data on CASP targets and illustrative parameters, that a combined approach is valuable for model discrimination.

Because of the limited state of methods for predicting changes in protein stability upon

mutation, we have applied a number of restrictions to select those mutants that are most likely to be accurately predicted by such methods. In the future, residues which are invariant in sequence alignments might also be usefully avoided. Such residues presumably have special roles in the structure or function of the protein, thus prediction methods, which are inherently based on average properties, might give misleading predictions.

In the current formulation of our methods, structural models are independently derived by purely computational methods before experimentation. In this case we can clearly only test models that are part of the initial prediction set (the "closed world" assumption in the language of Bayesian analysis). A more correct model would be missed if it were not included in this set. If the correct model were significantly different from the one we put forward, we would expect that the experimental data would agree at best only partially with the predicted features being tested. In such a case, we would still note that the model we put forward is the best among the alternatives, but it would also not be congruent with all the expected experimental outcomes. However, this hypothesis was not supported by our simulation on other CASP targets where no model is within 5 Å of the x-ray/NMR structure (not shown). Models with poor GDT_TS z-scores were found highly consistent with experimental mutagenesis data for most of these targets. This suggests that either the GDT_TS z-score is not a good criterion for model quality, or stability mutagenesis is unreliable for model discrimination when no high-quality model is included.

Of course, the more extensively the models are probed to test their congruence with expectation, the more likely we are to notice serious deviations between the best model and the experimental outcomes. Methods for predicting protein structure can be wrong

both locally and globally; both need to be confirmed and/or corrected. Out of practical concerns, our planning mechanisms attempt to limit the number of experiments conducted; however, our effort at spatial balance in planning seeks to spread the tests over the entire structure so that deviations in any part are more likely to be noticed.

5. MODEL DISCRIMINATION BY CONTINUOUS $\Delta\Delta G^\circ$ DATA

We have demonstrated that the information in stability mutagenesis is sufficient for discriminating predicted protein structure models (Chapter 4). However, we discretized the continuous $\Delta\Delta G^\circ$ data in order to fit into the framework previously developed for discrete cross-linking data (Chapter 3). This discretization threw away some useful information. Partial information provided by mutagenesis (how close the predicted and experimental $\Delta\Delta G^\circ$ values are, not just whether they are within a threshold) was used in data interpretation but was not taken into account in the calculation of coverage. This chapter develops effective planning metrics (Sec. 5.1) and a multi-phase planning algorithm (Sec. 5.2) that take full advantage of the information content in continuous $\Delta\Delta G^\circ$ data. The new approach selects a set of mutations that are most informative and robust, based on their combined ability to effectively discriminate all the models.

In our multi-phase planning algorithm, an initial plan is first selected by a greedy approach, then a large number of good plans is generated by a branch-and-bound search, followed by robustness analysis that significantly reduces the number of good plans. If multiple equivalently good plans are left, as usually happens, one of them will be selected either by other criteria such as dispersion of selected mutations, or simply by manual in-

spection. However, we observe that good plans overlap heavily, which implies that the last step of selection among good plans is probably not critical to the success of our approach, at least not as important as the previous steps.

We present prospective experiment plans for discriminating the three high-quality threading models of the bacteriophage lambda pTfa chaperone protein. We then studied the trade off between the optimality of selected plan and the speed of the algorithm, on a larger number of models of the same CASP targets as in Chapter 4.

5.1 Experiment Planning Metrics

Our goal is to select the optimal subset of mutations that is most informative for discriminating a given set of protein structure models. A natural criterion measuring the quality of plans is the average probability of choosing a wrong model, *i.e.* the Bayes error as it is known in the pattern recognition field [35].

Let $S = \{s_1, s_2, ...s_n\}$ be the given set of predicted protein structure models, and X be a vector of random variables representing the $\Delta\Delta G^\circ$ values with Normal errors. Then each model can be represented as a conditional density distribution in the multi-dimensional $\Delta\Delta G^\circ$ space $\mathcal{X}$, which is assumed to have the following form as discussed in Sec. 4.1.

$$p(X|s_i) = \mathcal{N}(\mu_i, \sigma^2 I) \ . \tag{5.1}$$

In other words, X is a multivariate Normal distribution with mean μ, and its variance is mutation independent and model independent.

Once the experimental $\Delta\Delta G^\circ$ values have been measured, we will choose the model

with the maximum posterior probability. Since experiment planning is done beforehand, we consider the Bayes error (or the expected error probability) as follows:

$$\varepsilon = \sum_{i=1}^{n} P(s_i)\varepsilon_i \tag{5.2}$$

$$\varepsilon_i = \int_{\mathcal{X}} p(X|s_i) \cdot I\{P(s_i)p(X|s_i) < \max_{j \neq i}(P(s_j)p(X|s_j))\}dX \tag{5.3}$$

where $P(s_i)$ is the prior probability of model s_i, and ε_i is the conditional error given that model s_i is correct. By "correct" we mean that the distributions of X w.r.t. the "true" protein structure and this model are very similar. The indicator function $I\{e\}$ returns 1 if Boolean expression e is true or 0 if false. In Eq. 5.3, e indicates if a wrong model is selected because the experimental data X is misclassified. In the rest of this chapter, almost all probabilities will be calculated as the integral of the product of $p(X|s_i)$ and such an indicator function. To simplify the notation, we define the probability P_i of a Boolean expression e w.r.t. model s_i:

$$P_i\{e\} = \int_{\mathcal{X}} p(X|s_i) \cdot I\{e\}dX \ . \tag{5.4}$$

The goal of experiment planning is to select a subset of mutations from all candidate mutations minimizing the Bayes error (Eq. 5.2). However, it is hard to calculate with multiple distributions in multi-dimensional space. Even with our assumptions about $p(X|s_i)$ in Eq. 5.1, we must resort to numerical techniques [35]. In the following sections, we develop upper and lower bounds on the Bayes error, along with criteria for robustness. These criteria together will allow us to develop planning algorithms (Sec. 5.2) to find (near) optimal plans that are most informative and robust for discriminating a given set of protein structure models.

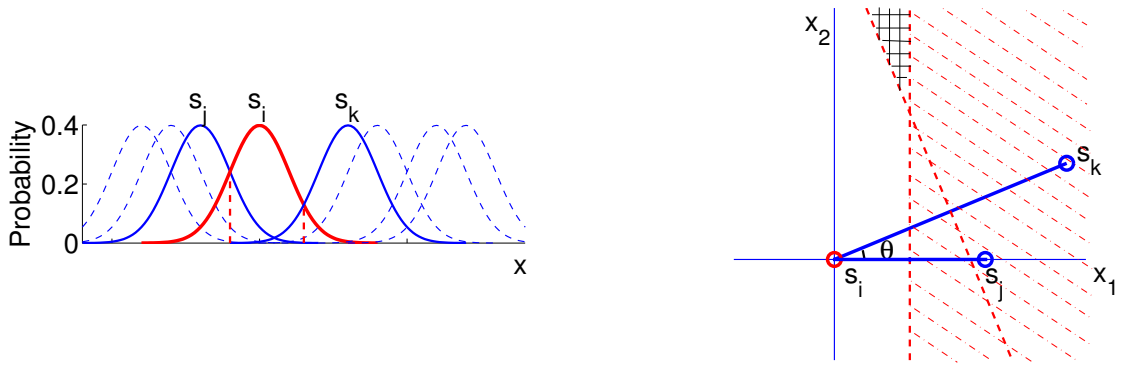
5.1.1 Bounds on Bayes Error

The problem of estimating and bounding the Bayes error has received considerable attention [37, 115, 36, 65]. These techniques have been demonstrated to be applicable to a variety of problems in pattern recognition and other fields. However, we take advantage of the particular structure of our mutagenesis planning problem in order to derive a tighter bound on Bayes error. In our experiment planning problem, numerical methods involving integrals in high-dimensional space are not practical due to the large number of combinations, and simple analytical bounds (such as the union bound) are not tight enough to achieve a high pruning rate in the branch-and-bound search. In this section, we develop tight upper and lower bounds on the Bayes error that are specific to our problem, under the Normal and independence assumptions discussed above. Tighter bounds on Bayes error will help us plan experiments that are most informative and least expensive for our application. Our method does involve numerical integration but it is limited to 2D, so that it is efficient. In order to employ the branch-and-bound algorithm, we also develop a lower bound on the Bayes error of the optimal plan that can be selected from a given set of candidate mutations.

For simplicity, we assume a uniform prior for models. All discussion in this section applies to the case of non-uniform priors unless stated otherwise. Under this assumption, and employing Eq. 5.4, the conditional error in Eq. 5.3 can be rewritten as

$$\varepsilon_i = P_i\{p(X|s_i) < \max_{j \neq i} p(X|s_j)\} \; . \tag{5.5}$$

We start our discussion from the Bonferroni inequalities [27] that provide straightforward



(a) (b)

Fig. 5.1: Tighter upper bound of ε_i with Normal distributions of a common variance. (a) In the 1D case, the conditional error ε_i (given that s_i is correct) is determined by the closest neighbors to s_i on each side, s_j and s_k. Other models (dashed curves) have no effect on ε_i. (b) In higher-dimensional cases, multiple models are unlikely to be collinear. However, if the angle between $\overrightarrow{s_i s_j}$ and $\overrightarrow{s_i s_k}$ is small and s_k is not closer to s_i than s_j is, adding s_k will only increase ε_i by a small amount (integral of $p(X|s_i)$ over the "#" shaded area).

initial bounds on ε_i.

$$\varepsilon_i \leq \sum_{j \neq i} P_i\{p(X|s_i) < p(X|s_j)\} \tag{5.6}$$

$$\varepsilon_i \geq \sum_{j \neq i} P_i\{p(X|s_i) < p(X|s_j)\} - \sum_{j<k \neq i} P_i\{p(X|s_i) < \min(p(X|s_j), p(X|s_k))\} \tag{5.7}$$

Eq. 5.6 is just a union bound, or Boole's inequality, which means that the probability that at least one of the wrong models beats the correct one is not greater than the sum of the probabilities of each individual wrong model beating the correct one. The union bound is easy to calculate and provides a simple minimization criterion for experiment planning; unfortunately it is often too loose for a large number of models.

Tighter Bounds on a Particular Plan

As we discussed (Sec. 4.1), we assume a common variance for all mutations in all models; hence the error probability is completely determined by the relative distances among the distribution means. Fig. 5.1 provides some intuition about the error probability. In the one-dimensional case, the conditional error probability ε_i is determined by the closest neighbor on each side of the distribution $p(X|s_i)$. This is true in higher-dimensional space if the model distributions are collinear, although that is very unlikely to happen. To relax the collinear requirement, consider a small angle between two vectors as shown in Fig. 5.1 (b). As we can see, in such an almost collinear situation (small θ), ε_i is much less than the sum of the individual error probabilities (union bound).

Inspired by the above intuition, we now do a rigorous derivation. Assuming model s_i is correct, we can shift the coordinate system so that μ_i is at the origin and the rest of the models are represented as vectors from the origin. We cluster these vectors into disjoint clusters C_t for $t = 1, 2, ...$ Regardless of how the clustering is done, we always have the following inequality:

$$\varepsilon_i \leq \sum_t P_i\{p(X|s_i) < \max_{j \in C_t} p(X|s_j)\} . \tag{5.8}$$

The difference between Eq. 5.8 and Eq. 5.6 is that the Bonferroni inequality is applied on clusters instead of individual models. Choosing a representative model s_{j_t} from each cluster C_t, we have

$$\begin{aligned} P_i\{p(X|s_i) < \max_{j \in C_t} p(X|s_j)\} &= P_i\{p(X|s_i) < p(X|s_{j_t})\} + P_i\{p(X|s_{j_t}) < p(X|s_i) < \max_{j \in C_t, j \neq j_t} p(X|s_j)\} & (5.9) \\ &\leq P_i\{p(X|s_i) < p(X|s_{j_t})\} + \sum_{j \in C_t, j \neq j_t} P_i\{p(X|s_{j_t}) < p(X|s_i) < p(X|s_j)\} & (5.10) \end{aligned}$$

Eq. 5.9 is just a rewriting of the probability; either model s_{j_t} beats s_i or some other models

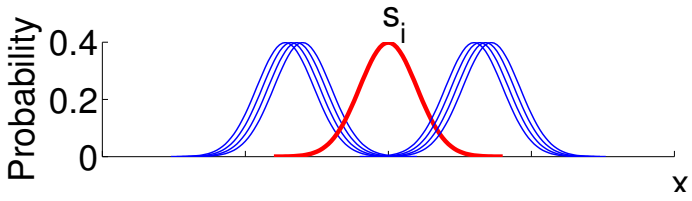


Fig. 5.2: Tighter lower bound of ε_i with Normal distributions of a common variance. In this 1D case, four wrong models that are very close to each other reside on each side of the correct model s_i. Suppose that $P_i\{p(X|s_i) < p(X|s_j)\} \approx \epsilon$ for all wrong models s_j, $j = 1, 2, ..., 8$. The lower bound from Eq. 5.7 is about -4ϵ. If we choose one representative model from each side, as in Eq. 5.12, the lower bound becomes about 2ϵ, which is much tighter.

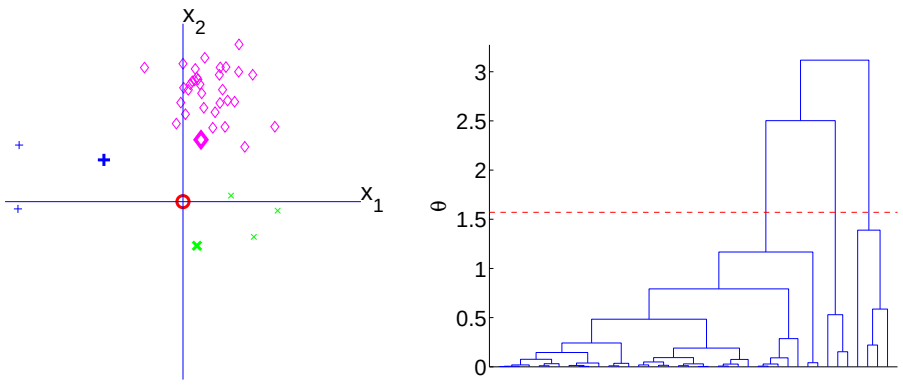
in cluster C_t beat it. Eq. 5.10 is obtained by applying the union bound on the second term of Eq. 5.9, where the first and second terms correspond to the integral of $p(X|s_i)$ over the stripe shaded area and the "#" shaded area in Fig. 5.1 (b), respectively.

If two events are highly dependent, the joint probability will be comparable to the individual probability of either one. Hence Eq. 5.7 could be negative if models are highly dependent, because the number of pairwise joint probabilities could be much larger than the number of individual probabilities. This is especially a concern for a large number of models. The same technique we discussed for the upper bound can also be employed to derive a tighter lower bound. Fig. 5.2 provides some intuition about obtaining a tighter lower bound by using a subset of models that are highly independent. In fact, the following discussion is more general. Still assuming s_i is correct, let $S' \subset S - \{s_i\}$ be a subset of the remaining models. Then we have

$$\varepsilon_i \geq P_i\{p(X|s_i) < \max_{j \in S'} p(X|s_j)\} \tag{5.11}$$

$$\geq \sum_{j \in S'} P_i\{p(X|s_i) < p(X|s_j)\} - \sum_{j<k \in S'} P_i\{p(X|s_i) < \min(p(X|s_j), p(X|s_k))\} \; . \tag{5.12}$$

Eq. 5.11 is true because the probability that any model in $S - \{s_i\}$ beats s_i is always greater



(a) (b)

Fig. 5.3: Model clustering. Assuming one model is correct and placed at the origin (red circle in (a)), the remaining models are represented as vectors from the origin. These vectors are hierarchically clustered w.r.t. their angles. A cutoff $\pi/2$ (red dashed line in (b)) gives three clusters (different markers in (a)). The vector with the shortest length is selected as the representative model for each cluster (bold markers in (a)).

than or equal to the probability that any model in a subset S' beats it. Eq. 5.12 is just the Bonferroni inequality applied to S'.

Eq. 5.8 and Eq. 5.10 hold for any clustering of models and any selection of representative models, and Eq. 5.12 holds for any subset S'. However, the tightness of these bounds depends critically on the clustering method and the selected representative models. We employ an agglomerative approach to cluster models, with distance between two clusters defined as the maximum angle between any two vectors in them, *i.e.*

$$d(C_a, C_b) = \max_{j \in C_a, k \in C_b} \angle \mu_j \mu_i \mu_k \tag{5.13}$$

See Fig. 5.3 for an example of hierarchical clustering of models. After the hierarchical tree is constructed, a cutoff θ is used to determine the number of clusters, and then the model with the smallest distance to s_i in each cluster is selected as the representative models. We also use the representative models as S' for the lower bound in Eq. 5.12, because

these models are likely to be highly independent, so that the pairwise joint probabilities are smaller and hence the lower bound is tighter. The optimal value of θ is the one that gives the tightest upper bound or lower bound. This optimal value could be model specific and different for the upper bound and the lower bound. While it is hard to determine the optimal θ value, it is easy to calculate the upper and lower bounds with several θ values and choose the tightest. In our implementation, we try three values of θ, $\pi/4$, $\pi/3$, and $\pi/2$. The running time is only three times that of using a fixed cutoff, and we found that the result is significantly improved in practice.

Lower Bound on the Optimal Plan

If the number of candidate or selected mutations is small, we can enumerate all possible plans, calculate the upper and lower bounds, and choose a good plan. In terms of Bayes error, plan A is better than plan B if the upper bound for A is less than the lower bound for B. In practice, the computational complexity of such a brute force method becomes prohibitive for even a modest number of mutations (*e.g.* choosing 10 from 100 mutations will yield on the order of 10^{13} possible combinations). In such cases, we can still use a greedy approach to minimize the upper bound on the Bayes error. A tight upper bound will allow us to identify a high quality set of selected mutations.

However, we still do not know how close a plan is to the optimal one. In order to evaluate the optimality of a given plan M, we define the Bayes error of the optimal plan of size m from a set of candidate mutations, M_c, as follows:

$$\varepsilon(M_c, m) = \min_{M' \subset M_c, |M'|=m} \varepsilon(M') \,, \tag{5.14}$$

and the optimality of plan M as

$$\text{Optimality}(M, M_c) = \frac{\varepsilon(M_c, |M|)}{\varepsilon(M)} \ . \tag{5.15}$$

Since both the numerator and denominator in Eq. 5.15 are hard to calculate, we compute a lower bound on the optimality as

$$\text{Optimality}(M, M_c) \geq \frac{lb(M_c, |M|)}{ub(M)} \ . \tag{5.16}$$

where $ub(M)$ is the upper bound we previously discussed (Eq. 5.8, Eq. 5.10) and we develop below $lb(M_c, |M|)$, the lower bound of Bayes error on the optimal plan. If M_c is the set of all candidate mutations, Eq. 5.16 provides a way to evaluate the quality of plan M. The larger the score, the better the plan is guaranteed to be. A score close to 1 indicates a plan that is guaranteed to be near optimal (A plan with a lower score may still be good, but we just cannot prove it with our bounds). If M_c is a subset of mutations, Eq. 5.16 provides a criterion to prune subtrees in a branch-and-bound algorithm: we can ignore all combinations in M_c if the score in Eq. 5.16 is greater than 1, because no plan of the same size from M_c can be better than the plan M that we already have. Let us first prove the following lemma, which will allow us to derive the lower bound on the optimal plan, $lb(M_c, |M|)$.

Lemma 5.1 *Let*

$$d^2 = \sum_{i=1}^{n} d_i^2 \tag{5.17}$$

be the sum of squares of n positive real numbers d_i, $i = 1, 2, ..., n$; *and let*

$$\varepsilon_i = \int_{-\infty}^{-d_i/2} \frac{1}{\sqrt{2\pi}\,\sigma} e^{-x^2/2\sigma^2} dx \ . \tag{5.18}$$

be the cumulative density of Normal distribution $N(0, \sigma)$ *at point* $-d_i/2$*. For a fixed value of* d^2*,* $\sum_{i=1}^{n} \varepsilon_i$ *is minimized when*

$$d_i = d_j \tag{5.19}$$

for all $1 \leq i, j \leq n$*.*

Proof: Suppose we can find $d_i = b$ and $d_j = a$ that are not equal, say $0 < b < a$, and let $c = \sqrt{\frac{a^2+b^2}{2}}$ be new equal values for d_i and d_j, so that the sum of squared values d^2 is not changed. The changes in ε_i and ε_j are

$$\Delta\varepsilon_i = -\frac{1}{\sqrt{2\pi}} \int_{b/2}^{c/2} e^{-\frac{x^2}{2}} dx \tag{5.20}$$

$$\Delta\varepsilon_j = +\frac{1}{\sqrt{2\pi}} \int_{c/2}^{a/2} e^{-\frac{x^2}{2}} dx \tag{5.21}$$

It is easy to show that

$$c - b > a - c \,, \tag{5.22}$$

i.e. the integral region of $\Delta\varepsilon_i$, $b/2, c/2$, is larger than that of $\Delta\varepsilon_j$, $c/2, a/2$. Furthermore, the density is higher on the first region because it is closer to the mean ($b < c < a$) (see Fig. 5.4 for an illustration). Therefore, we have

$$\Delta\varepsilon_i + \Delta\varepsilon_j < 0 \,, \tag{5.23}$$

i.e. $\sum_{i=1}^{n} \varepsilon_i$ is reduced by making d_i and d_j equal. We can continue this process until $d_i = d_j$ for all $1 \leq i, j \leq n$ and $\sum_{i=1}^{n} \varepsilon_i$ is minimized. □

A lower bound on the Bayes error based on pairwise risk functions developed for multi-hypothesis testing [37] is as follows:

$$\varepsilon \geq \left(\frac{2}{n}\right)^2 \cdot \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \varepsilon_{ij} \tag{5.24}$$

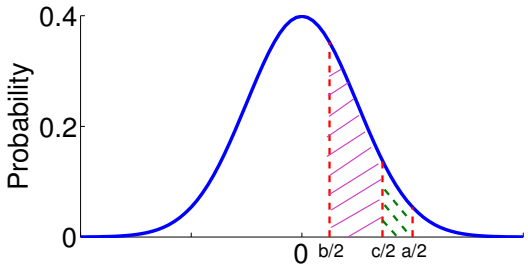


Fig. 5.4: Illustration of the proof of Lemma 5.1. The shaded areas indicate the decrement (upward solid lines) and increment (downward dashed lines) of pairwise Bayes errors by replacing distances a and b with $c = \sqrt{\frac{a^2+b^2}{2}}$. Because region $\frac{b}{2}, \frac{c}{2}$ is larger and closer to the mean, the sum of pairwise errors must be decreased.

Combining Lemma 5.1 and Eq. 5.24, we have

$$\varepsilon \geq \frac{2(n-1)}{n} \int_{-\infty}^{-r} \frac{1}{\sqrt{2\pi}\,\sigma} e^{-x^2/2\sigma^2} dx \tag{5.25}$$

where $r = \frac{1}{2}\sqrt{\frac{d^2}{\binom{n}{2}}}$ and d^2 is the sum of squared distances among all model distribution means. When d^2 is maximized, the lower bound in Eq. 5.25 is minimized and becomes a lower bound for any plan of the same size, including the optimal plan. The sum of squared distances can be rewritten as

$$d^2 = \sum_{i<j} \sum_{k} (\mu_{ki} - \mu_{kj})^2 \tag{5.26}$$

$$= \sum_{k} \sum_{i<j} (\mu_{ki} - \mu_{kj})^2 \tag{5.27}$$

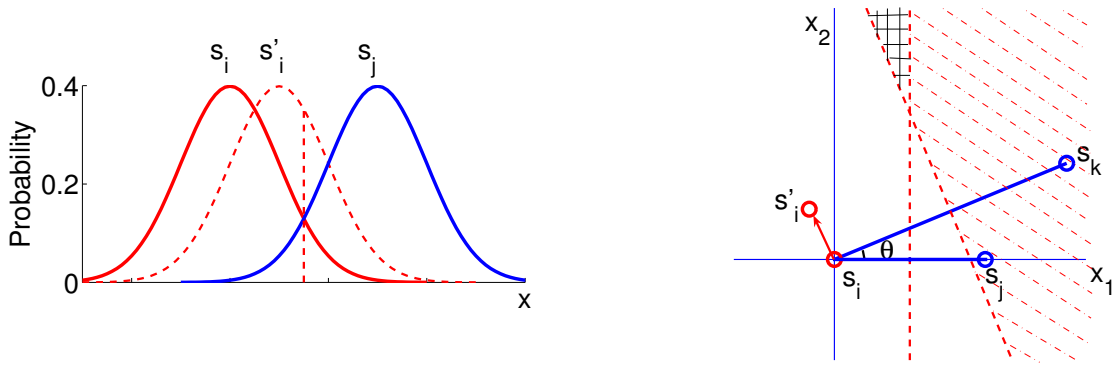
where μ_{ki} is the mean of the distribution of model s_i in the k^{th} dimension, *i.e.* the predicted $\Delta\Delta G^\circ$ value of the k^{th} mutation according to model s_i. Since the inner sum of Eq. 5.27 is for only one mutation, we can easily maximize d^2 by independently choosing mutations according to their sums of squared distances over all models.

In summary, to calculate the lower bound for the optimal plan of size k, we first choose k mutations so that the sum of squared distances d^2 is maximized; then the average distance is used to calculate the lower bound by Eq. 5.25.

5.1.2 Robustness w.r.t. the Inaccuracy of $\Delta\Delta G^\circ$ Prediction

As we discussed, discrimination is based primarily on relative destabilization. We run a risk, however, of the experimental data having a consistent bias in the difference between experimental and predicted $\Delta\Delta G^\circ$ values, due to some protein-specific properties. Such a bias could arise from a protein that is relatively easier or harder to destabilize than the proteins against which the predictor is trained. If we selected mutations in which one model were predominantly predicted to be more destabilized than the others, that model would be favored if the protein were relatively easy to destabilize. If we knew the bias for a protein, as a single number or a distribution, we could incorporate it into the prediction distribution $p(X|s_i)$. We assume, however, that we only know the range of bias (*i.e.* the bias could be anywhere in the range) because that is probably a more realistic situation in practice.

A good experiment plan needs to be robust w.r.t. such "unknown" bias. In Chapter 4, we achieved a balance in the direction of relative destabilization by separating each pair of models $\{r, s\}$ into two directed pairs $\langle r, s\rangle$ and $\langle s, r\rangle$. The framework developed in this chapter allows us to evaluate the robustness of plans in a more systematic way and obtain a balanced design naturally. Since we do not know whether the bias is present, we will still use the unbiased distribution for data interpretation. Therefore, the error bounds cal-



(a) (b)

Fig. 5.5: The effect of systematic bias on $P_i\{p(X|s_i) < p(X|s_j)\}$, $P_i\{p(X|s_i) < \min(p(X|s_j), p(X|s_k))\}$ and $P_i\{p(X|s_j) < p(X|s_i) < p(X|s_k)\}$. $p'(X|s_i)$ is the projection of the biased distribution onto the line $s_i s_j$ or the plane $s_i s_j s_k$, which replaces $p(X|s_i)$ in the integrals for calculating bounds of Bayes error. However, the integration areas (shaded areas) are not changed because we do not know where $p'(X|s_i)$ is and still use $p(X|s_i)$ for data interpretation.

culation (Sec. 5.1.1) is still valid except that we need to redefine the probability of Boolean expression e w.r.t. model s_i (Eq. 5.4) as follows

$$P'_i\{e\} = \int_{\mathcal{X}} p'(X|s_i) \cdot I\{e\} dX \ , \tag{5.28}$$

where $p'(X|s_i)$ is the unknown biased distribution of the correct model. Fig. 5.5 illustrates the effect of bias on error bounds calculation (Eq. 5.10, Eq. 5.12). In order to calculate the error probabilities, we need to project the biased distribution onto the line $\mu_i\mu_j$ or plane $\mu_i\mu_j\mu_k$. Regardless of the direction of bias (*i.e.* the protein is relatively easy or hard to destabilize), the vector $\overrightarrow{\mu_i\mu'_i}$ can be decomposed into two perpendicular vectors, one parallel and the other orthogonal to line $\mu_i\mu_j$ or plane $\mu_i\mu_j\mu_k$. Since the orthogonal vector does not provide any information for model discrimination (distributions $p(X|s_i)$, $p(X|s_j)$ and $p(X|s_k)$ have the identical projection in this dimension), such projections lose

no information for discrimination. In our implementation, we try bias values within the range $-2, 2$ kcal/mol at a resolution of 0.1 and use the worst case (maximum upper bound of Bayes error) as the robustness measurement of a plan. A robust plan will have a biased error bound close to the unbiased one. We choose a plan with small error probability in both unbiased and biased cases.

5.1.3 Top Group Selection

For a large number of models, simultaneous discrimination of all models may require a prohibitive number of mutations. In such cases, a sequential approach may be a better alternative: plan a small set of mutations, measure the experimental $\Delta\Delta G^{\circ}$ values, choose a subset of top models and repeat until one or a few models stand out. If some models are very close to each other, our planning metrics may also be problematic. The error probability may never be made small enough no matter how many mutations are selected. In order to address these issues, we may decide to choose a "top group" of models rather than a single best model after the experimental $\Delta\Delta G^{\circ}$ values have been measured. Accordingly, we can optimize a plan so that the correct model will be included in the top group with a high probability. The error bounds we discussed above (Eq. 5.10 and Eq. 5.12) can be slightly modified for this purpose. If we will choose a top group of size t, we should ignore the closest $t-1$ neighbors in calculating the error bounds, because these models are hardest to distinguish from the correct one. The error bound calculated in this way will bound the probability that more than $t-1$ models beat the correct one, *i.e.* the probability that the

```
MUTPLAN(m, λ1, λ2, η, Mc)
  ub, M ← MUTPLAN-Greedy(m, Mc)                    # Fig. 5.7
  ub*, Ψ ← MUTPLAN-BnB(m, λ1, ub, {M}, ∅, Mc)      # Fig. 5.8
  Ψ ← {M ∈ Ψ|ub(M)/ub* ≤ λ2}                       # test optimality
  sort Ψ in ascending order of biased upper bounds  # robustness
```

Fig. 5.6: Mutagenesis planning algorithm. The inputs include the desired size of plan (m), cutoffs for subtree pruning in branch-and-bound algorithm (λ_1) and good plan selection in post-processing (λ_2), bias range (η) and the set of candidate mutations (M_c).

```
MUTPLAN-Greedy(m, Mc)
  M ← ∅
  while |M| < m
     e ← arg min_{e∈Mc} ub(M ⋃{e})          # find next mutation
     M ← M ⋃{e}
     Mc ← Mc − {e}
     Mc ← {e ∈ Mc|e satisfies all constraints}  # optional
  return ub(M), M
```

Fig. 5.7: Greedy algorithm for mutagenesis planning. The inputs are the desired size of plan (m) and the set of candidate mutations (M_c).

correct model is not included in the top group of size t. Choosing the best model is a special case for $t = 1$.

5.2 Experiment Planning Algorithms

Employing the error bounds and robustness analysis discussed in Sec. 5.1, we develop a multi-phase algorithm for mutagenesis planning (Fig. 5.6). We first find a greedy solution by selecting mutations one by one minimizing the upper bound on Bayes error at each step (Fig. 5.7). A branch-and-bound algorithm (Fig. 5.8) then finds a sufficient number of plans with small error probability in the unbiased case. These potentially good plans are then

```
MUTPLAN-BnB(m, λ, ub*, Ψ, M_s, M_c)
  if |M_s| + |M_c| = m                                  # only one possible plan
     M_s ← M_s ⋃ M_c
     M_c ← ∅
  if constraintsSatisfied(M_s) and lb(M_s ⋃ M_c, m)/ub* ≤ λ   # subtree worth exploring
     if |M_s| = m                                       # a complete plan
        Ψ ← Ψ ⋃{M_s}
        if ub(M_s) < ub*                                # a tighter upper bound
           ub* ← ub(M_s)
     else for i from 1 to m − |M_s| + 1                 # discard M_c i at the i^th child
        M'_s ← M_s ⋃ M_c 1..i − 1                       # select mutations before that
        M'_c ← M_c i + 1..|M_c|                         # update candidate mutations
        ub*, Ψ ← MUTPLAN-BnB(m, λ, ub*, Ψ, M'_s, M'_c)
  return [ub*, Ψ]
```

Fig. 5.8: Branch and bound algorithm for mutagenesis planning. The inputs include the desired size of plan (m), pruning cutoff (λ), the best upper bound (ub^*) and good plans (Ψ) so far, and sets of selected and candidate mutations (M_s and M_c) at the current node.

subjected to post-processing and robustness analysis. If there is more than one good robust plan, the selection of the final plan is subjective. Other criteria such as the dispersion of selected mutations in the sequence or 3D structure can also be used to evaluate plans and help select the best one.

The heart of our algorithm is a branch-and-bound search (Fig. 5.6), which finds all plans with guaranteed optimality in the unbiased case, according to user-specified parameters. Fig. 5.9 shows an example of a branch-and-bound search tree for choosing two from six mutations (at four positions). We enumerate the subsets of discarded mutations [35]. The path from the root to the current node specifies discarded mutations; all mutations after the current one are still to be considered; and those before the current node but not shown on the path are selected.

Fig. 5.8 presents the branch-and-bound algorithm. The function *constraintsSatisfied*

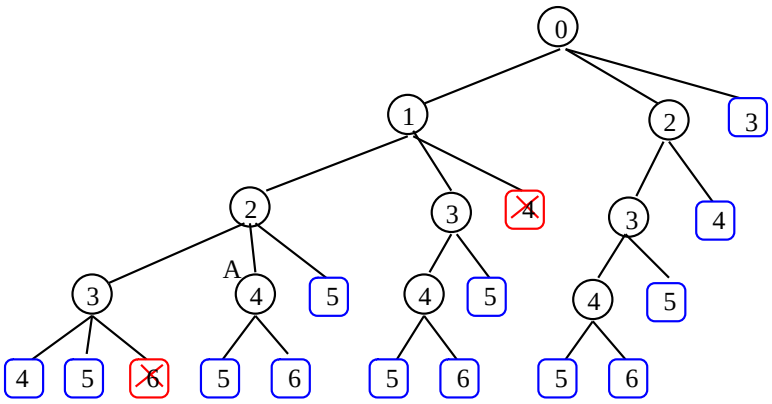


Fig. 5.9: A complete search tree of branch-and-bound algorithm for choosing two from six mutations at four positions, {A2G, F3A, F3L, R4A, R4G, M5A}, indexed from 1 to 6. Circles are internal nodes and squares leaf nodes. Red crosses indicate violation of the constraint requiring at most one mutation per position. The mutations on a path from the root are discarded. The sets of candidate and selected mutations can be derived from the path: all mutations indexed after the current one are candidate mutations, those indexed before the current one but not shown in the path are selected. For example, at node A, mutations A2G, F3A and R4A (indices 1, 2 and 4) have been discarded, mutation F3L (index 3) has been selected and mutations R4G and M5A (indices 5 and 6) are still to be considered.

checks satisfaction of user-specified constraints, *e.g.* at most one mutation per position can be selected. The width of the search tree is limited by $m - |M_s| + 1$ because $|M_s| \leq m$ and hence the number of mutations selected at current level cannot exceed $m - |M_s|$. The output is the best upper bound (ub^*) and a list of potentially good plans (Ψ), which will be subjected to postprocessing and robustness analysis. We need to calculate the lower bound of the optimal plan ($lb(M_s \bigcup M_c, m)$) for each visited internal node, but the upper bound ($ub(M_s)$) only for each potentially good plan (*i.e.* those that cannot be eliminated under the user-specified parameters). The lower bound is much more efficient to calculate than the upper bound because it only requires a table lookup of the Normal cumulative density function (Eq. 5.25), while the upper bound involves numerical integration in 2D

space (Eq. 5.10). Fortunately, the upper bound does not need to be computed for most nodes.

The pruning cutoff λ is used to control the speed of the algorithm and the output size. If $\lambda > 1.0$, sub-optimal plans will also be listed. If $\lambda < 1.0$, some good plans might be missed in order to speed up the algorithm. However, plans that are "really" good are guaranteed to be kept. For example, if $\lambda = 0.5$, any plan with Bayes error lower than half of ub^*, the best upper bound found in the branch-and-bound search, cannot be missed. Since ub^* bounds the Bayes error of the optimal plan, any plan with Bayes error lower than half of that of the the optimal plan cannot be missed. The default value of λ is 1.0, so that all good plans will be listed. At the end of the branch-and-bound algorithm, all identified plans are filtered by the best upper bound ub^* (see parameter λ_2 in Fig. 5.6).

In our branch-and-bound algorithm, we assume that user-specified constraints are monotonic. A monotonic constraint is one that is violated by a superset of mutations if it is violated by any subset. For example, a constraint requiring at most one mutation per position is monotonic. In fact, we can avoid visiting right siblings if any monotonic constraint is violated (not shown in Fig. 5.8). We can slightly modify the algorithm in Fig. 5.8 to handle non-monotonic constraints: only check the satisfaction of constraints on complete plans.

In order to increase the pruning rate, we initially sort all mutations in ascending order of upper bound on Bayes error (easy to calculate in the 1D case). The branch-and-bound algorithm (Fig. 5.8) is then applied to this ordered list of mutations. The heuristic here is to exclude good mutations first, so that the error of the remaining mutations is larger, as is the chance of pruning left subtrees, which are larger (Fig. 5.9) [35]. This reordering

improves the pruning rate significantly. Although we can reorder mutations at each level of the search tree, the cost of the sorting may not worth the benefit, which is not likely to be as significant as the initial sorting.

5.3 Results

5.3.1 Prospective Experiment Planning for pTfa

We put our new planning mechanism into practice on the three high-quality model of the pTfa protein of bacteriophage lambda (see Chapter 4). In this three-model case, the upper bound and the lower bound converge to the exact Bayes error. Therefore all results for this test case show the Bayes error instead of the bounds. Fig. 5.10 shows the Bayes error and optimality of the plans selected by the greedy algorithm (Fig. 5.7).

A good initial plan can provide a higher pruning rate and hence improve the efficiency of our branch-and-bound algorithm. Since we use the greedy plan as the initial plan, the optimality of the greedy plan provides a clue about the expected efficiency of the branch-and-bound algorithm. As we can see in Fig. 5.10, the optimality score is about 0.6 for the greedy plan for the three Tfa models, which means that the Bayes error of the greedy plan is within a factor of two of the optimal value. Therefore, we expect a high pruning rate in the branch-and-bound algorithm using this greedy plan as initial solution. Since all plans are subjected to robustness analysis, we would like to have a sufficient number of candidate plans selected by branch-and-bound search. If not enough plans are selected, we

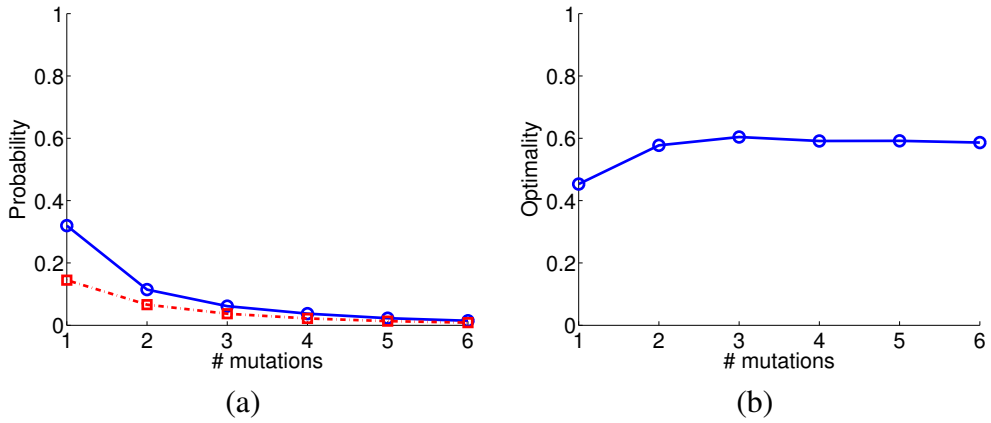
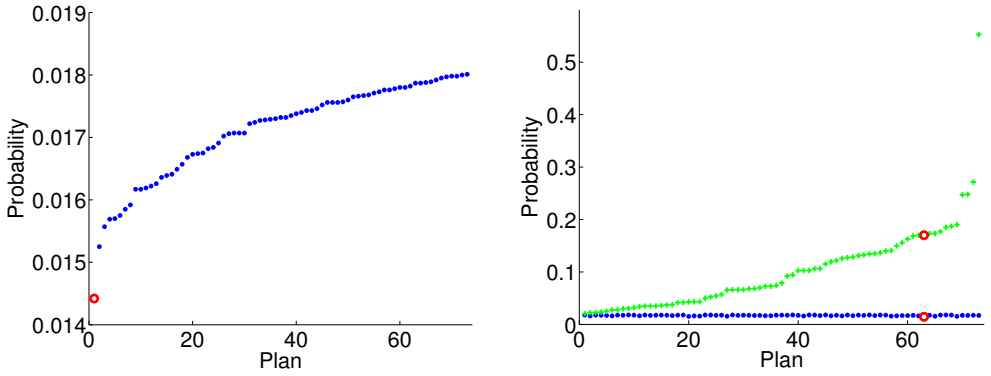


Fig. 5.10: Greedy plan for three Tfa models. (a) Bayes error of greedy plans (blue solid line, circles) and lower bound of the optimal plan of the same size (red dash-dotted line, squares). (b) Optimality of greedy plans as defined in Eq. 5.15.

can increase the value of λ_1 and λ_2 to obtain more.

The greedy plan is good in the unbiased case, with a Bayes error of 1.4%. However, with a bias range of $-2, 2$ kcal/mol, the Bayes error goes up to 17%. In order to get a better plan, we first use our branch-and-bound search to generate a larger number of plans that are good in the unbiased case, and then apply the robustness analysis on these plans. Fig. 5.11 (a) shows the plans of six mutations for the three Tfa models selected by our branch-and-bound search at $\lambda_1 = 1$ and $\lambda_2 = 1.25$. A total of 15942 nodes were visited in about 2 hours, and 73 plans were selected, with a total of 24 unique mutations. As we can see, the greedy plan happens to have the smallest Bayes error in this case. Fig. 5.11 (b) shows the result of robustness analysis with a bias range of $-2, 2$ kcal/mol. Good plans in the unbiased case are not necessarily good in the biased case, and the greedy plan is no longer the best. Also, the biased Bayes errors are more distinguishable than the unbiased ones.



(a) (b)

Fig. 5.11: Six-mutation plans for three Tfa models selected by MUTPLAN at $\lambda_1 = 1$ and $\lambda_2 = 1.25$. With a total of 192 candidate mutations at 77 positions, there are about 5.7×10^{10} possible combinations of six mutations. Plans are shown in ascending order of Bayes error in (a) unbiased and (b) biased cases. The red circles indicate the Bayes error of the greedy plan in both cases.

Tab. 5.1 presents three plans — the best, the greedy and the worst among all plans selected by our branch-and-bound search ("worst-of-bb"). As we can see, these three plans are comparable without bias (similar Bayes errors). However, the best plan stands out in the presence of bias (significantly smaller Bayes error). In order to see if the good plans selected by the automatic robustness analysis are consistent with our heuristics, *i.e.* achieving a balance in the direction of relative destabilization, we define a directed distance between two models as follows.

$$d\langle s_i, s_j \rangle = \sqrt{\sum_k I\{\mu_{ki} < \mu_{kj}\} \cdot (\mu_{ki} - \mu_{kj})^2}\,, \tag{5.29}$$

where the indicator function I returns 1 if a mutation is more destabilizing (or less stabilizing) in model s_i than in s_j, and 0 otherwise. Tab. 5.1 lists the directed distances for all model pairs. As we can see, the best plan (smallest Bayes error in the biased case) is also the most balanced one, *i.e.* the two directed distances are comparable for each pair

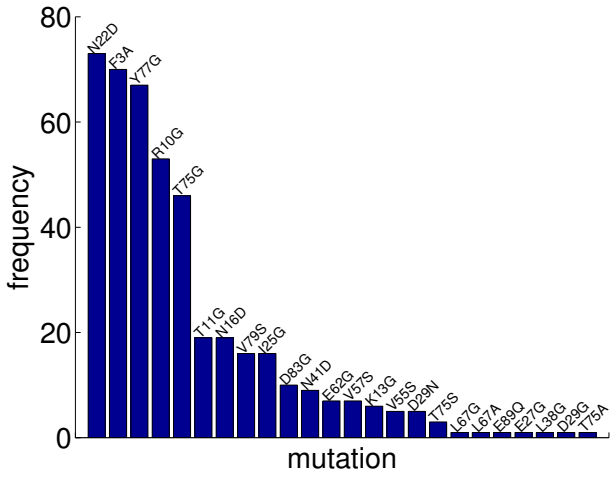


Fig. 5.12: Frequencies of 24 unique mutations involved in all 73 plans in Fig. 5.11.

of models. The worst-of-bb plan is very unbalanced and the greedy one is in the middle. Therefore, we conclude that our robustness evaluation is consistent with our heuristics and can choose good robust plans automatically.

Note that only two mutations are different between the greedy and the best plans in Tab. 5.1. In fact, all the good plans selected by the branch-and-bound search overlap heavily. Fig. 5.12 presents the frequencies of 24 unique mutations involved in all 73 plans selected by the branch-and-bound search. We also observed that the best plan shares four (of six) mutations with the plan selected in Chapter 4 (Fig. 4.8). The Bayes error for the previous plan in unbiased and biased cases are 1.9% and 2.5%, respectively. It is close to the Bayes error of the best plan in Tab. 5.1, 1.8% and 2%.

Tab. 5.1: Three plans for three Tfa models: the best, the greedy and the worst plan among those selected. The Bayes errors in both unbiased case (ε) and biased case with a bias range of $-2, 2$ kcal/mol (ε_{biased}) are listed. The following three tables show the details of the three plans. The distances between model means are shown as directed pairs, where more destabilizing and more stabilizing mutations are separated.

plan	ε	ε_{biased}
best	0.018	0.020
greedy	0.014	0.170
worst	0.017	0.553

The best plan

mutation	$\Delta\Delta G_p^\circ$			$d\langle s_i, s_j\rangle$					
N22D	0.68	-3.26	0.02	0.00	3.94	0.00	0.66	3.28	0.00
Y77G	-3.26	0.23	-0.34	3.49	0.00	2.92	0.00	0.00	0.56
T75G	-2.98	-0.75	0.27	2.24	0.00	3.25	0.00	1.02	0.00
F3A	0.15	-0.09	-2.71	0.00	0.24	0.00	2.86	0.00	2.62
D83G	1.73	-1.29	-0.35	0.00	3.02	0.00	2.08	0.94	0.00
T11G	-0.56	0.19	-2.34	0.74	0.00	0.00	1.79	0.00	2.53
total				4.21	4.98	4.37	4.02	3.56	3.68

The greedy plan

mutation	$\Delta\Delta G_p^\circ$			$d\langle s_i, s_j\rangle$					
N22D	0.68	-3.26	0.02	0.00	3.94	0.00	0.66	3.28	0.00
Y77G	-3.26	0.23	-0.34	3.49	0.00	2.92	0.00	0.00	0.56
T75G	-2.98	-0.75	0.27	2.24	0.00	3.25	0.00	1.02	0.00
F3A	0.15	-0.09	-2.71	0.00	0.24	0.00	2.86	0.00	2.62
R10G	-1.35	-0.84	1.50	0.51	0.00	2.84	0.00	2.33	0.00
N16D	-0.71	-2.77	-0.17	0.00	2.06	0.54	0.00	2.60	0.00
total				4.18	4.46	5.25	2.93	4.90	2.68

The worst-of-bb plan

mutation	$\Delta\Delta G_p^\circ$			$d\langle s_i, s_j\rangle$					
N22D	0.68	-3.26	0.02	0.00	3.94	0.00	0.66	3.28	0.00
Y77G	-3.26	0.23	-0.34	3.49	0.00	2.92	0.00	0.00	0.56
T75G	-2.98	-0.75	0.27	2.24	0.00	3.25	0.00	1.02	0.00
R10G	-1.35	-0.84	1.50	0.51	0.00	2.84	0.00	2.33	0.00
N16D	-0.71	-2.77	-0.17	0.00	2.06	0.54	0.00	2.60	0.00
V79S	-2.45	-2.93	-0.50	0.00	0.48	1.96	0.00	2.44	0.00
total				4.18	4.48	5.60	0.66	5.47	0.56

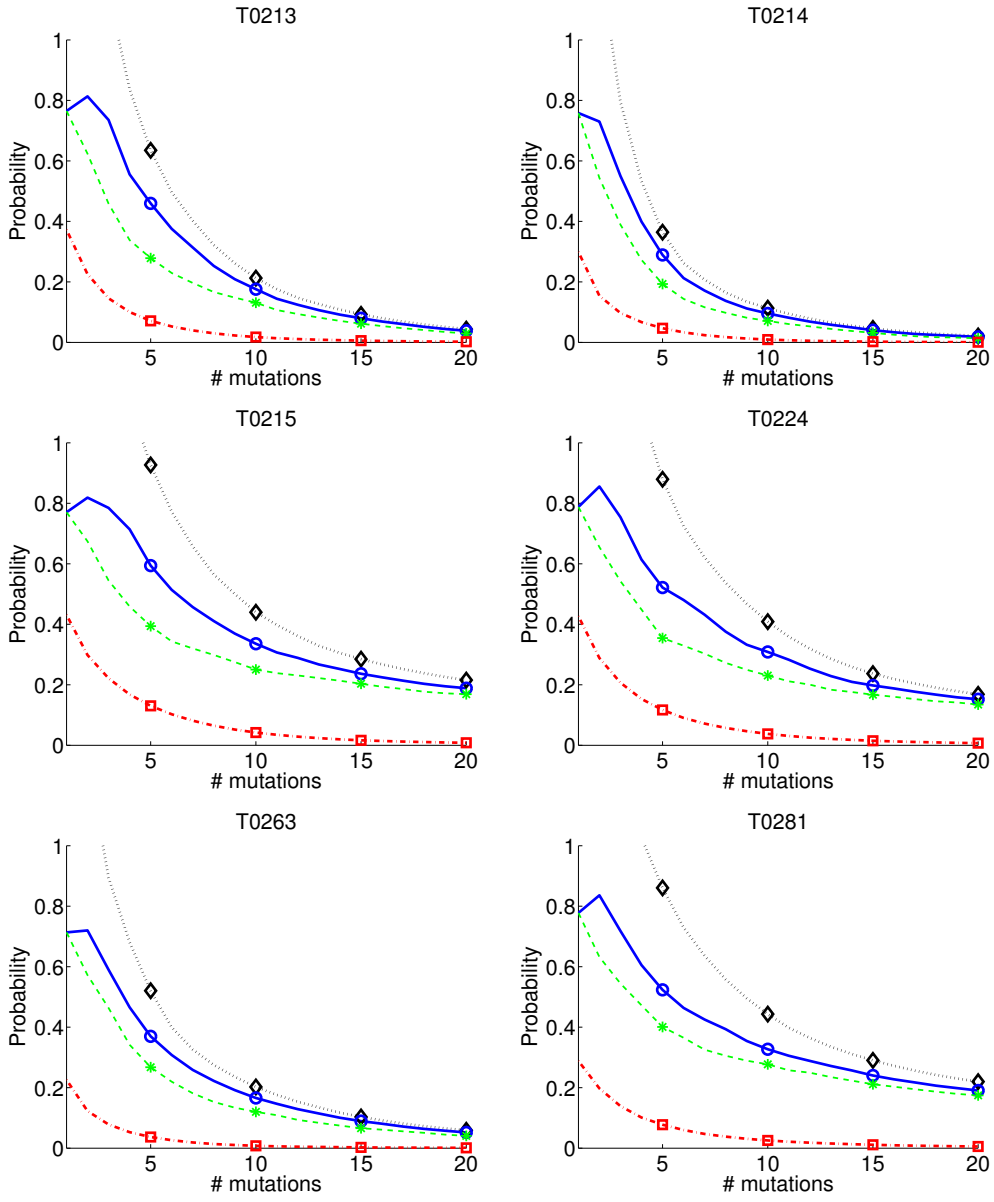


Fig. 5.13: Error probabilities of greedy plans on the 10 models with highest GDT_TS z-score for each CASP target: union bound (black dotted), tight upper bound (blue solid), tight lower bound (green dashed) and lower bound for the optimal plan of the same size (red dash-dotted).

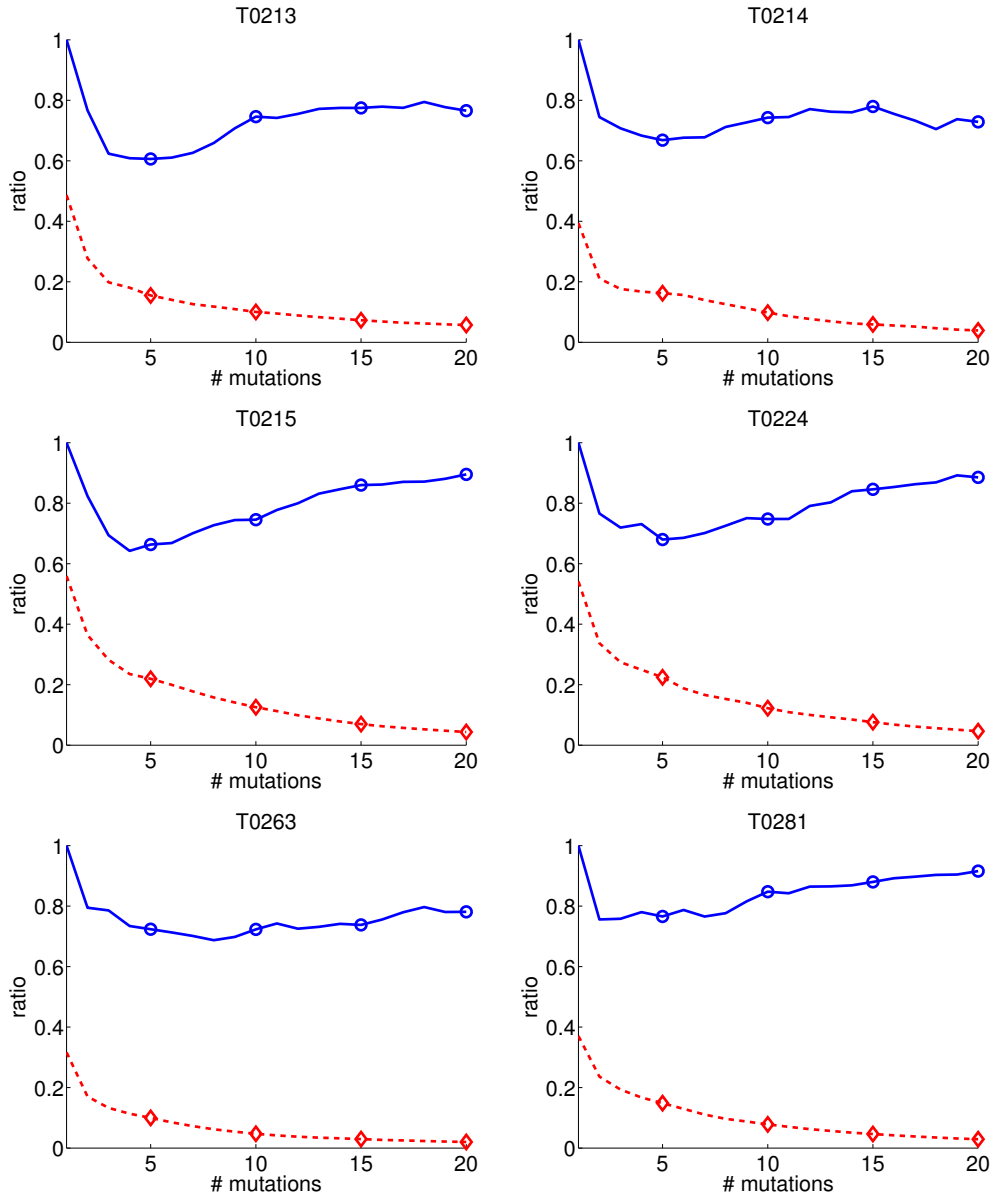


Fig. 5.14: Tightness (blue solid) and lower bound of optimality (red dashed) of the greedy plans on the 10 models with highest GDT_TS z-score for each CASP target.

5.3.2 Optimality vs. Speed

The branch-and-bound search is the most time consuming step in our approach. The user-specified parameter λ in Fig. 5.8 (λ_1 in Fig. 5.6) is used to control how much effort should be put in the branch-and-bound search to find good plans. Although we used the default value $\lambda = 1.0$ for the three Tfa models, the branch-and-bound search with the default λ value may become prohibitive for a larger number of mutations and models. In order to determine an appropriate value for λ, we first examine the other two factors that determine the pruning rate of branch-and-bound together with λ: the upper bound for a plan and the lower bound for the optimal plan (denominator and numerator in Eq. 5.16). We define the tightness of bounds on a plan M as the ratio between the lower bound and the upper bound.

$$\text{Tightness}(M) = \frac{lb(M)}{ub(M)} \,. \tag{5.30}$$

The larger the value, the tighter the bounds and the better they bound the Bayes error. This value approaches 1 as the upper and lower bounds converge to the exact Bayes error.

Fig. 5.13 presents the error bounds of the greedy plans on ten high-quality models (with the highest GDT_TS scores) for each selected CASP target. Fig. 5.14 presents the corresponding tightness and optimality (actually the lower bound of optimality, Eq. 5.16) for these greedy plans. As we can see, the tightness score is high for most of the targets, which means that the upper bound is close to the lower bound, and hence the exact Bayes error. However, the optimality is low, which may result in a poor pruning rate in branch-and-bound if the default value of λ (1.0) is used. Since the upper bound is tight, the poor optimality indicates that either the greedy plan is bad or the lower bound for the optimal

plan is loose. We also observed that the greedy plan is usually very good in the unbiased case (Fig. 5.11). Therefore, the most likely reason for the poor optimality is that the lower bound for the optimal plan is loose.

In order to improve the pruning rate in the branch-and-bound search, we specify a small value for λ. If we choose $\lambda = 0.1$, theoretically we can only guarantee that no plan with Bayes error lower than $ub^*/10$ will be missed (see Fig. 5.8), while in practice it may be much better due to the loose lower bound for the optimal plan. Furthermore, if we take into account the uncertainties that are not modeled, such as the outliers of $\Delta\Delta G^\circ$ values (see Fig. 4.2) and experimental errors, error probabilities cannot be too small. A plan with Bayes error (or its upper bound) of 1% is probably not that different, in practice, from another plan with Bayes error of 0.1%. Therefore, it might be a good idea to sacrifice some theoretical optimality for real speed. We claim that the main goal of the branch-and-bound search is to provide a sufficient number of good plans, but not to find the complete set of good plans or the best one. Since the worst running time of the branch-and-bound algorithm could be exponential, it can take an unacceptably long time if an inappropriate λ value is specified. A good way to avoid this situation is to start from a conservative value of λ, which should be a value close to the optimality of the greedy plan. Then we can increase it iteratively until we get enough good plans or a robust good plan.

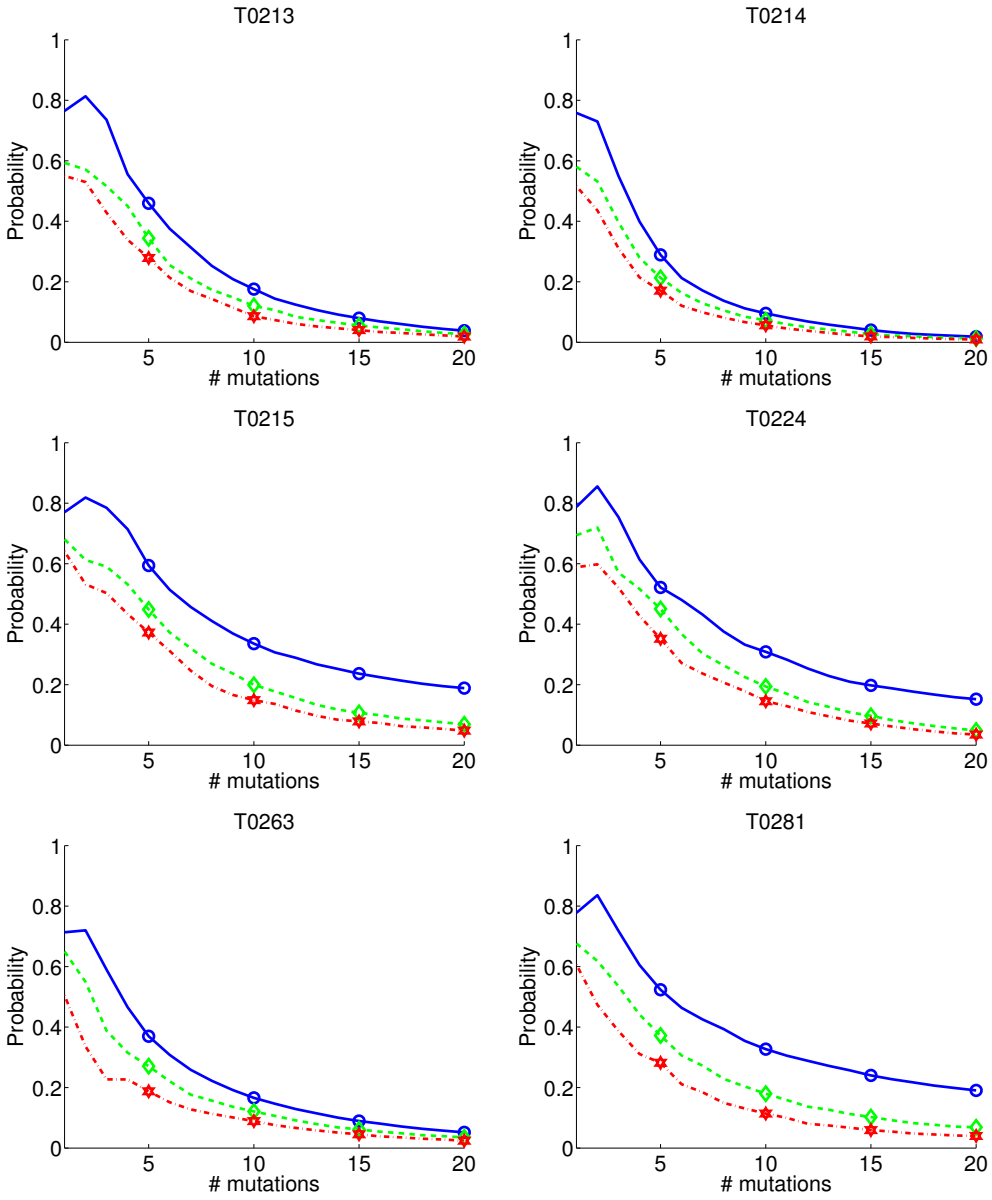


Fig. 5.15: Upper bound of Bayes error for the greedy plans in Fig. 5.13 w.r.t. top group of size 1 (blue solid), 2 (green dashed) and 3 (red dash-dotted).

5.3.3 Top Group Selection

Fig. 5.15 presents the upper bound of Bayes error for the greedy plans in Fig. 5.13, w.r.t. top group of size 1, 2 and 3. The error bound was significantly improved for targets T0215, T0224 and T0281 by choosing the two top models instead of one. That is because these targets contain identical models: models 026 and 027 are identical for T0215 and T0281; models 501 and 506 are identical for T0224. This example is a little extreme because we probably should have excluded duplicates of models before experiment planning. However, it demonstrates the merit of top group selection in the presence of extremely similar models.

5.4 Discussion

In this chapter, we have developed criteria and algorithms that take full advantage of the information provided by stability mutagenesis. Bayes error provides a natural criterion to evaluate the quality of plans for continuous data. We develop tight error bounds for Bayes error and a multi-phase algorithm to choose high-quality plans. In support of the robustness analysis, the plan selected by minimizing the Bayes error is also heuristically balanced, and hence justifies the idea of directed model pair in our previous discrete version of planning algorithm.

Our criteria also handle missing data naturally. If the $\Delta\Delta G^{\circ}$ prediction of a mutation is missing in some models, it should not be used to discriminate these models from others. In the calculation of error bounds, what matters is the differences between predictions in different models. We define the difference between a missing value and any value as zero,

so that the missing data are not for or against any model regardless of the experimental value of $\Delta\Delta G^{\circ}$.

In the discrete version, we need to specify a cutoff T to determine if a model-pair is discriminable by a mutation. As we discussed in Chapter 4, the choice of T has significant effect on the coverage of mutations. If T is too high, we underestimate the discrimination information provided by mutagenesis. If T is too low, we overestimate it. Furthermore, mutations with utility smaller than T also provide some information for model discrimination. Such partial information is not taken into account by the coverage criterion. These problems are no longer present in the continuous version. No user-specified cutoff is required and all information is taken into account.

Although the continuous version of planning has advantages over the discrete one, we may need the discrete version for the multimodal approach. If mutagenesis is discretized as in Chapter 4, cross-linking and mutagenesis can be combined easily and the combined plan can be optimized. For continuous mutagenesis and discrete cross-linking, we can plan separately and combine them in data interpretation. However, it is not so straightforward to optimize the combined plan because different criteria are used in the two methods.

6. SITE-DIRECTED PROTEIN RECOMBINATION

Chapter 3 to Chapter 5 were dedicated to our PRAXIS approach for high-throughput protein structure elucidation. This chapter addresses the other experiment planning challenge introduced in Chapter 1: site-directed protein recombination, an important application in protein engineering. We developed a probabilistic hypergraph model (Fig. 6.1(a)) in which edges represent pairwise and higher-order residue interactions, while edge weights represent the degree of "hyperconservation" of the interacting residues (Sec. 6.1). Then we developed efficient algorithm to choose the optimal breakpoint locations in recombination that minimize the total perturbation to these interactions.

Our hypergraph model generalizes the traditional representations of sequence information in terms of single-position conservation and structural interactions in terms of pairwise contacts. Hyperconservation can reveal significant residue interactions both within members of the family (arising from structural and functional constraints) and generally common to all proteins (arising from general properties of the amino acids). We then combine family-specific and database-wide statistics with suitable weighting (Sec. 6.1.1), ensure non-redundancy of the information in super- and sub-edges with a multi-order potential score (Sec. 6.1.2), and derive edge weights by mean potential scores (Sec. 6.1.3). We also

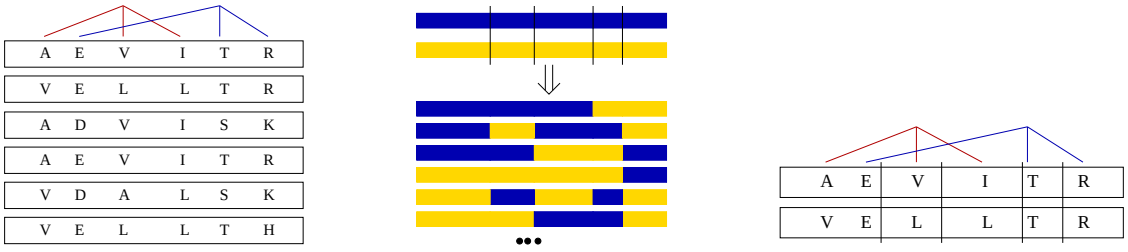


Fig. 6.1: Hypergraph model of evolutionary interactions, and effects of site-directed protein recombination. (a) Higher-order evolutionary interactions (here, order-3) determining protein stability and function are observed in the statistics of "hyperconservation" of mutually interacting positions. The left edge is dominated by Ala,Val,Ile and Val,Leu,Leu interactions, while the right is dominated by Glu,Thr,Arg and Asp,Ser,Lys ones. The interactions are modeled as edges in a hypergraph with weights evaluating the degree of hyperconservation of an interaction, both generally in the protein database and specific to a particular family. (b) Site-directed recombination experiments mix and match sequential fragments from homologous parents to construct a library of hybrids with the same basic structure but somewhat different sequences and thus different functions. (c) Site-directed recombination experiments perturb edges that cross one or more recombination breakpoints. The difference in edge weights derived for the parents and those derived for the hybrids indicates the effect of the perturbation on maintenance of the evolutionarily preserved interactions.

evaluate the significance of hyperconservation by calculating their p-values (Sec. 6.1.5). Testing on previous recombination data of beta-lactamases (Sec. 6.3) shows that the effect of non-redundant higher-order terms is significant and can be effectively handled by our model.

Site-directed recombination experiments (Fig. 6.1(b)) seek to create hybrids with the same basic structure but different functions, by mixing and matching sequential fragments from homologous parents. Optimizing retainment of multi-order interactions after recombination (Fig. 6.1(c)) should help identify the best recombinants and thus the best locations for breakpoints. In support of this optimization, we develop criteria to evaluate the quality of hybrid libraries by considering the effects of recombination on edge weights (Sec. 6.1.4). We then formulate the optimal selection of breakpoint locations as a sequentially-constrained hypergraph partitioning problem (Sec. 6.2), and prove it to be NP-hard in general (Sec. 6.2.1). We develop exact and heuristic algorithms for a number of important cases (Secs. 6.2.2–6.2.5), and demonstrate their practical effectiveness in design of recombination experiments for members of the beta-lactamase family (Sec. 6.3).

6.1 A Hypergraph Model of Evolutionary Interactions

Previous work on breakpoint selection for site-directed recombination (Sec. 2.5) focused on retaining pairwise residue-residue relationships. In order to more completely model statistical interactions in a protein, it is necessary to generalize single-position sequence conservation and pairwise structural contact. We model a protein and its reference struc-

ture with a weighted hypergraph $G = (V, E, w)$, where vertices $V = \{v_1, v_2, \cdots, v_{|V|}\}$ represent residue positions in sequential order on the backbone, edges $E \subseteq 2^V$ represent mutually interacting sets of vertices, and weight function $w \quad E \rightarrow \mathbb{R}$ represents the relative significance of edges. We construct an order-c edge $e = \langle v_1, v_2, \cdots, v_c \rangle$ for each set of residues (listed in sequential order for convenience) that are in mutual contact; this construction can readily be extended to capture other forms of interaction, *e.g.* long-range interaction of non-contacting residues due to electrostatics. Note that subsets of vertices associated with a higher-order edge form lower-order edges.

The definition of the edge weight is key to effective use of the hypergraph model. In the case where the protein is a member of a family with presumed similar structures, edge weights can be evaluated from the general database or a specific family. There are many observed residue values (across the family or database) for the vertices of any given edge. We thus build up to an edge weight by first estimating the probability of the residue values, then decomposing the probability to ensure non-redundant information among multi-order edges for the same positions. Finally we determine the effect on the pattern of these values due to recombination according to a set of chosen breakpoint locations.

6.1.1 Distribution of Hyperresidues in Database and Family

Let $R = \langle r_1, r_2, \cdots, r_c \rangle$ be a "hyperresidue," a c-tuple of amino acid types (*e.g.* $\langle$Ala, Val, Ile$\rangle$). Intuitively speaking, the more frequently a particular hyperresidue occurs in functional proteins, the more important it is expected to be for their folding and function. We can estimate

the overall probability p of hyperresidues from their frequencies in the database $\mathcal{D}$ of protein sequences and corresponding structures:

$$p(R) = \frac{\#R \text{ in } \mathcal{D}}{|\mathcal{D}|} , \tag{6.1}$$

where $|\mathcal{D}|$ represents the total number of tuples of the same order in the database. When considering a specific protein family $\mathcal{F}$ with a multiple sequence alignment (MSA) and shared structure, we can estimate position-specific (*i.e.*, for edge e) probability of a hyperresidue:

$$p_e(R) = \frac{\#R \text{ at } e \text{ in } \mathcal{F}}{|\mathcal{F}|} , \tag{6.2}$$

where $|\mathcal{F}|$ is the total number of tuples at positions forming edge e in the family MSA, *i.e.* the total number of sequences in the family MSA.

Estimation of probabilities from frequencies is valid only if the frequencies are large. Thus the general probability estimated from the whole database (Eq. 6.1) is more robust than the position-specific one from a single family (Eq. 6.2). However, family-specific information is more valuable as it captures the evolutionarily-preserved interactions in that family. To combine these two aspects, we adopt the treatment of sparse data sets proposed by Sippl [102]:

$$q_e(R) = \omega_1 \cdot p(R) + \omega_2 \cdot p_e(R) , \tag{6.3}$$

but employing weights suitable for our problem:

$$\omega_1 = \frac{1}{1 + |\mathcal{F}|\rho} \quad \text{and} \quad \omega_2 = 1 - \omega_1 , \tag{6.4}$$

where ρ is a user-specified parameter that determines the relative contributions of database and family. Note that when $\rho = 0$, $q_e(R) = p(R)$ and the family-specific information is

ignored; whereas when $\rho = \infty$, $q_e(R) = p_e(R)$ and the database information is ignored. Using a suitable value of ρ, we will obtain a probability distribution that is close to the overall database distribution for a small family but approximates the family distribution for a large one.

6.1.2 Multi-order Potential Score for Hyperresidues

Since we have multi-order edges, with lower-order subsets included alongside their higher-order supersets, we must ensure that these edges are not redundant. In other words, a higher-order edge should only include information not captured by its lower-order constituents. The inclusion-exclusion principle ensures non-redundancy in a probability expansion, as demonstrated in the case of protein structure prediction [100]. We define an analogous multi-order potential score for hyperresidues at edges of orders 1, 2, and 3, respectively, as follows:

$$\phi_{v_i}(r_\alpha) = \log q_{v_i}(r_\alpha) \,, \tag{6.5}$$

$$\phi_{v_i v_j}(r_\alpha r_\beta) = \log \frac{q_{v_i v_j}(r_\alpha r_\beta)}{q_{v_i}(r_\alpha) \cdot q_{v_j}(r_\beta)} \,, \tag{6.6}$$

$$\phi_{v_i v_j v_k}(r_\alpha r_\beta r_\gamma) = \log \frac{q_{v_i v_j v_k}(r_\alpha r_\beta r_\gamma) \cdot q_{v_i}(r_\alpha) \cdot q_{v_j}(r_\beta) \cdot q_{v_k}(r_\gamma)}{q_{v_i v_j}(r_\alpha r_\beta) \cdot q_{v_i v_k}(r_\alpha r_\gamma) \cdot q_{v_j v_k}(r_\beta r_\gamma)} \,. \tag{6.7}$$

Here, $\phi_{v_i}(r_\alpha)$ captures residue conservation at v_i; $\phi_{v_i v_j}(r_\alpha r_\beta)$ captures pairwise hyperconservation and is zero if v_i and v_j are not in contact or their residue types are completely independent; $\phi_{v_i v_j v_k}(r_\alpha r_\beta r_\gamma)$ captures 3-way hyperconservation and is zero if v_i, v_j, and v_k are not in mutual contact or their residue types are completely independent. The poten-

tial score of higher-order hyperresidues can be defined similarly. The potential score of a higher-order hyperresidue contains no information redundant with that of its lower-order constituents.

An alternative understanding of the hyperconservation score is as a measurement of over/underrepresentation of hyperresidues. Let $q'_e(R)$ be the probability of hyperresidue R at edge e assuming no order-$|R|$ conservation (there might be lower-order conservation). Then we can write the general definition of the hyperconservation score as follows,

$$\phi_e(R) = \log \frac{q_e(R)}{q'_e(R)} \,, \tag{6.8}$$

which includes Eq. 6.5 – 6.7 as special cases.

6.1.3 Edge Weights

In the hypergraph model, edge weights measure evolutionary optimization of higher-order interactions. For a protein or a set of proteins $\mathcal{S} \subseteq \mathcal{F}$, we can evaluate the significance of an edge as the average potential score of the hyperresidues appearing at the positions forming the edge:

$$w(e) = \sum_R \frac{\#R \text{ at } e \text{ in } \mathcal{S}}{|\mathcal{S}|} \cdot \phi_e(R) \,. \tag{6.9}$$

6.1.4 Edge Weights for Recombination

We can view recombination as a two-step process: *decomposing* followed by *recombining*. In the decomposing step, each protein sequence is partitioned into $n+1$ intervals according

to the breakpoints, and the hypergraph is partitioned into $n+1$ disjoint subgraphs by removing all edges spanning a breakpoint. The impact of this decomposition can be individually assessed for each edge, using Eq. 6.9 for the parents $\mathcal{S}$.

In the recombining step, edges removed in the decomposing step are reconstructed with new sets of hyperresidues according to all combinations of parent fragments. The impact of this reconstruction can also be individually assessed for each edge, yielding a breakpoint-specific weight:

$$w(e, X) = \sum_{R} \frac{\#R \text{ at } e \text{ in } \mathcal{L}}{|\mathcal{L}|} \cdot \phi_e(R) \ . \tag{6.10}$$

$X = \{x_1, x_2, \cdots, x_n\}$ is a set of breakpoints at which parent sequences $\mathcal{S}$ are recombined, where $x_t = v_i$ indicates that breakpoint x_t is between residues v_i and v_{i+1}. In this case, the potential score of hyperresidue R is weighted by the amount of its representation in the library $\mathcal{L}$. Note that we need not actually enumerate the set of hybrids (which can be combinatorially large) in order to determine the weight, as the frequencies of the residues at the positions are sufficient to compute the frequencies of the hyperresidues.

The combined effect of the two-step recombination process on an individual edge, the *edge perturbation*, is then defined as the change in edge weight:

$$\Delta w(e, X) = w(e) - w(e, X) \ . \tag{6.11}$$

If all vertices of e are in one fragment, we have $w(e) = w(e, X)$ and $\Delta w(e, X) = 0$. The edge perturbation thus integrates essential information from the database, family, parent sequences, and breakpoint locations, and serves as a guide for breakpoint selection in site-directed recombination (Sec. 6.2).

6.1.5 Significance of Multi-order Hyperconservation

We estimated the probabilities of hyperresidues from their frequencies (Eq. 6.1, Eq. 6.2), and the risk of doing so is that the estimation is valid only if the frequencies are sufficiently large. In this section, we evaluate the statistical significance of hyperresidues by calculating their p-values. The p-value of an order-c ($c = 2, 3, 4$) hyperresidue is defined as the probability of obtaining the exact or a more extreme number of occurrences, assuming that there is no order-c hyperconservation but that the hyperconservation of its lower-order constituents (if any) is retained. By retaining the hyperconservation of lower-order constituents in the calculation of p-values, we are able to separate multi-order hyperconservation and avoid spurious higher-order hyperconservation that is merely a combination of lower-order hyperconservation. Therefore, we can identify higher-order hyperconservation that is significant.

The p-value of a hyperresidue is determined by its number of occurrences and those of its lower-order constituents, independent of the particular residue types. For example, if we are calculating the p-value of hyperresidue $\langle$Ala, Val, Ile$\rangle$ at positions $\langle i, j, k\rangle$ of a MSA, we only care that Ala is at position i, Val is at position j, and Ile is at position k. Therefore, without losing any generality, we use "A" to represent the residue type of interest and at each position and "O" to represent other residue types. We also use "X" as a wild card, either "A" or "O". For example, in this representation, there are four possible types for order-2 hyperresidues, "AA", "AO", "OA" and "OO"; "AX" represents either "AA" or "AO", and "XX" represents all four types. Higher-order hyperresidue types can be represented

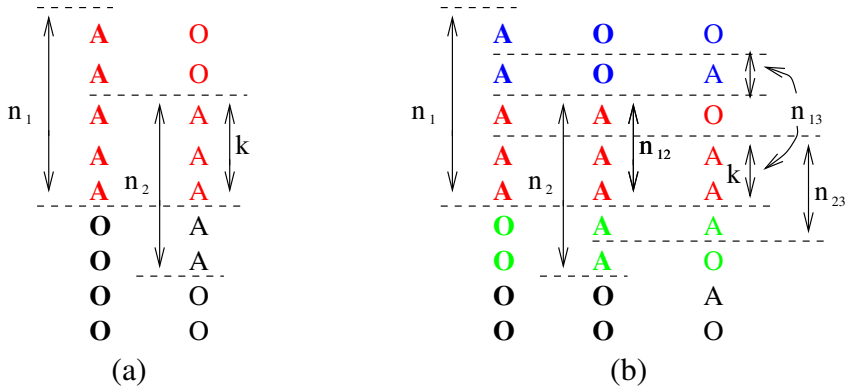


Fig. 6.2: Illustration of random permutation of residues. (a) Order-2: given any permutation of the first column, randomly permute residues in the second column. The residues within each group (red and black) are free to re-permute at the second column without violating the constraint $x_{12} = k$. (b) Order-3: given any permutation of the first two column, randomly permute residues in the third column. The red, blue and green groups correspond to three terms in Eq. 6.15. The residues within each group (red, blue, green and black) are also free to re-permute as long as all constraints are enforced.

similarly.

Let n be the size of the MSA (*i.e.* the total number of residues in each column) and let n_e be the number of occurrences of a hyperresidue of interest at hyperedge e (a set of position indices). We also define a random variable x_e as the number of occurrences of the hyperresidue assuming that there is no order-$|e|$ hyperconservation but that the hyperconservation of lower-order constituents (if any) is retained. In order to retain the hyperconservation of lower-order constituents, we derive the conditional probability distribution of x_e. For example, the conditional probability of x_{123} is $p(x_{123}|n, n_1, n_2, n_3, n_{12}, n_{13}, n_{23})$, where the conditions indicate the size of the MSA and the number of occurrences of lower-order constituents. For simplicity, the conditions are made implicit in the rest of this section.

Let us start from order-2 hyperconservation. The distribution of x_{12} is determined by

permutations of the residues in these two columns. Hence we have

$$p(x_{12} = k) = \frac{n\binom{n_2}{k}\binom{n-n_2}{n_1-k}n_1(n-n_1)}{nn} = \frac{\binom{n_2}{k}\binom{n-n_2}{n_1-k}}{\binom{n}{n_1}} \tag{6.12}$$

The numerator in Eq. 6.12 is the number of permutations with $x_{12} = k$, and the denominator is the total number of possible permutations. In fact the number of permutations of the first column (n) cancels out in the numerator and denominator. For any particular permutation of the first column, k "A"s and $n_1 - k$ "O"s are chosen from the second column $\left(\text{yielding } \binom{n_2}{k}\binom{n-n_2}{n_1-k} \text{ possibilities}\right)$ to match the n_1 "A"s in the first column, so that the number of "AA"s is $x_{12} = k$. The residues within each group (n_1 and $n - n_1$) can be permuted ($n_1(n - n_1)$ possibilities) without violating this constraint. Fig. 6.2 (a) illustrates the calculation of $p(x_{12} = k)$. Since the observed value of x_{12} is n_{12}, the p-value is calculated as $p(x_{12} \geq n_{12})$ or $p(x_{12} \leq n_{12})$, whichever is smaller. The former indicates over-representation of hyperresidues and the latter under-representation, and they are not independent for the same hyperedge (see Sec. 6.3).

As the reader may have noticed, Eq. 6.12 is a hypergeometric distribution [32]. The hypergeometric distribution arises when a random selection without replacement is made among objects of two distinct types ("A" and "O" in our case). If there are p good objects among n total and we take q samples, the probability of obtaining r good ones is

$$h(n, p, q, r) = \frac{\binom{p}{r}\binom{n-p}{q-r}}{\binom{n}{q}} . \tag{6.13}$$

Now we can rewrite Eq. 6.12 as an h function with $p = n_2$, $q = n_1$ and $r = k$, *i.e.* by randomly taking n_1 residues from the second column (n_2 of n are "A"s) to match the n_1

"A"s in the first column, the probability of obtaining k "AA"s is

$$p(x_{12} = k) = h(n, n_2, n_1, k) \tag{6.14}$$

The calculation of p-value for order-3 edges is slightly more complicated because we need to retain the hyperconservation of its order-2 constituents. As seen in Eq. 6.12, the permutation of the first column has no effect on $p(x_{12})$ — $p(x_{123})$ is independent of the permutation of the first two columns as long as we have $x_1 = n_1$, $x_2 = n_2$ and $x_{12} = n_{12}$. We first derive the following joint probability by randomly permuting the third column.

$$\begin{aligned} p(x_{123} = k, x_{13} = n_{13}, x_{23} = n_{23}) &= h(n, n_3, n_{12}, k) \\ &= \cdot h(n - n_{12}, n_3 - k, n_1 - n_{12}, n_{13} - k) \\ &\quad \cdot h(n - n_1, n_3 - n_{13}, n_2 - n_{12}, n_{23} - k) \end{aligned} \tag{6.15}$$

i.e. the probability of obtaining k "AAA"s, n_{13} "AXA"s and n_{23} "XAA"s. Fig. 6.2 (b) illustrates how this joint probability is derived. The three terms in Eq. 6.15 enforce three constraints, $x_{123} = k$, $x_{13} = n_{13}$ and $x_{23} = n_{23}$. The last two constraints must be satisfied in order to retain the potential hyperconservation of lower-order constituents. In fact, these three constraints are just variants of Eq. 6.12 that each enforces one constraint and all adopt the hypergeometric distribution. The first term of Eq. 6.15 enforces the order-3 constraint $x_{123} = k$, *i.e.* choosing n_{12} residues from the third column (n_3 of n are "A"s) to match the n_{12} "AA"s in the first two columns; the probability of obtaining k "A"s, and thus k "AAA"s, is $h(n, n_3, n_{12}, k)$. The second and third terms enforce the two order-2 constraints, $x_{13} = n_{13}$ and $x_{23} = n_{23}$, where the argument values of the hypergeometric functions are updated after each constraint is applied. These four parameters are: number of residues

Tab. 6.1: Calculation of parameters in Eq. 6.15. Starting from any permutation of the first two columns with $x_1 = n_1$, $x_2 = n_2$ and $x_{12} = n_{12}$, satisfy the constraints one by one by permuting residues in the third column. The remaining columns of the table represent the number n of residues to choose from, the number p of type "A", the number q to be selected, and the number r of type "A" among the selected ones. Each row corresponds to parameters of one hypergeometric function and the probability $p(x_{123} = k)$ is the product of all these hypergeometric functions.

constraint	n	p	q	r
$x_{123} = k$	n	n_3	n_{12}	k
$x_{13} = n_{13}$	$n - n_{12}$	$n_3 - k$	$n_1 - n_{12}$	$n_{13} - k$
$x_{23} = n_{23}$	$n - n_1$	$n_3 - n_{13}$	$n_2 - n_{12}$	$n_{23} - k$

Tab. 6.2: Extension of Tab. 6.1 to order-4.

constraint	n	p	q	r
$x_{1234} = k$	n	n_4	n_{123}	k
$x_{124} = n_{124}$	$n - n_{123}$	$n_4 - k$	$n_{12} - n_{123}$	$n_{124} - k$
$x_{134} = n_{134}$	$n - n_{12}$	$n_4 - n_{124}$	$n_{13} - n_{123}$	$n_{134} - k$
$x_{234} = n_{234}$	$n - n_{12} - n_{13} + n_{123}$	$n_4 - n_{124} - n_{134} + k$	$n_{23} - n_{123}$	$n_{234} - k$
$x_{14} = n_{14}$	$n - n_{12} - n_{13} - n_{23} + 2n_{123}$	$n_4 - n_{124} - n_{134} - n_{234} + 2k$	$n_1 - n_{12} - n_{13} + n_{123}$	$n_{14} - n_{124} - n_{134} + k$
$x_{24} = n_{24}$	$n - n_1 - n_{23} + n_{123}$	$n_4 - n_{14} - n_{234} + k$	$n_2 - n_{12} - n_{23} + n_{123}$	$n_{24} - n_{124} - n_{234} + k$
$x_{34} = n_{34}$	$n - n_1 - n_2 + n_{12}$	$n_4 - n_{14} - n_{24} + n_{124}$	$n_3 - n_{13} - n_{23} + n_{123}$	$n_{34} - n_{134} - n_{234} + k$

remaining, number of "A"s remaining in the third column, number of samples to take, and number of "A"s to take at each step, corresponding to the four parameters n, p, q, and r in Eq. 6.13.

We tabulate the calculation of parameters of Eq. 6.15 in Tab. 6.1. The method for filling entries in Tab. 6.1 is straightforward. The first row has n as the size of the MSA, n_3 as the number of type "A" residues in the third column, n_{12} as the number of "AA"s in the first two columns, and k as the number of "AAA"s desired. Then the last two columns can be filled by employing the inclusion-exclusion principle because the k "AAA"s also count as "AXA"s and "XAA"s, and the number of "A"s remaining in the first two columns need to

subtract the number of "AA"s. Finally, the first two entries of each row (other than the first) can be obtained from the numbers in the previous row, by subtracting the last two entries from the first two.

Once we have the conditional probability in Eq. 6.15, we can calculate the probability $p(x_{123} = k)$ by normalization.

$$p(x_{123} = k) = \frac{p(x_{123} = k, x_{13} = n_{13}, x_{23} = n_{23})}{\sum_{i=0}^{\min(n_{12},n_{13},n_{23})} p(x_{123} = i, x_{13} = n_{13}, x_{23} = n_{23})} . \tag{6.16}$$

Then the p-value can be calculated as $p(x_{123} \geq n_{123})$ or $p(x_{123} \leq n_{123})$, whichever is smaller. The summation range used here, $0, \min(n_{12}, n_{13}, n_{23})$, may not be tight, *i.e.* not all values in this range are possible in practice. For Eq. 6.13 to be meaningful, we have to satisfy constraints $0 \leq r \leq p$ and $0 \leq q - r \leq n - p$, and these constraints have to be satisfied for all rows in Tab. 6.1 and Tab. 6.2. Tight ranges can be derived by solving series of inequalities. Alternatively, we can artificially define $h(n, p, q, r) = 0$ if any of these constraints is violated, so that unrealistic numbers of occurrences have no effect on the probability distribution of x_e.

The probability $p(x_e)$ and hence the p-value of higher-order hyperresidues can be calculated in the same way. Tab. 6.1 can be extended to higher-order hyperresidues and the same method for filling entries applies. Tab. 6.2 presents the table for order-4 hyperresidues. Starting from any permutation of the first three columns such that $x_e = n_e$, where $e \in \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$, we randomly permute the fourth column ($x_4 = n_4$) to satisfy seven constraints one by one. The joint probability $p(x_{1234} = k, x_{124} = n_{124}, x_{134} = n_{134}, x_{234} = n_{234}, x_{14} = n_{14}, x_{24} = n_{24}, x_{34} = n_{34})$ is the product of seven hypergeometric functions whose argument values correspond to the seven rows

in Tab. 6.2. Similarly, $p(x_{1234})$ can be calculated by normalization and the p-value can be calculated as $p(x_{1234} \geq n_{1234})$ or $p(x_{1234} \leq n_{1234})$, whichever is smaller.

A hyperresidue is considered significant if its p-value is less than a user specified significance level α. A hyperedge is considered significant if it contains at least one significant hyperresidue. Since there are multiple hyperresidues at each hyperedge, the testing of hyperedge significance may be subject to the Bonferonni correction for multi-hypothesis testing [12] in order to avoid spurious positives. In other words, we should adjust the significance level based on the number of tests.

$$\alpha_e \approx \frac{\alpha}{N} \tag{6.17}$$

where N is the number of possible hyperresidues at the current hyperedge. However, we also know that hyperresidues for the same hyperedge are highly dependent (see Fig. 6.6 in Sec. 6.3), and thus the Bonferonni correction may be too conservative and some significant hyperedges may be missed.

The computational complexity of p-value calculation is as same as that of the hypergeometric function $h(n, p, q, r)$, which involves factorial functions, making it hard to obtain the exact value for large parameters. The Gamma function $g(n) = \int_0^{+\infty} x^{n-1} e^{-x} dx$ is used to interpolate the factorial n. We use the Matlab function *hygepdf* in the *stats* toolbox to calculate $h(n, p, q, r)$. This function obtains the logarithm of $g(n)$ without computing $g(n)$, and also avoids the underflow and overflow that may occur if it is computed directly [26]. An order-c hyperresidue is considered if and only if all its order-$(c-1)$ constituents have non-zero occurrences. Note that a considered hyperresidue may have zero occurrences itself. The number of order-c hyperresidues that must be considered is usually much less

than 20^c, so that the computation is not very time-consuming in practice.

6.2 Optimization of Breakpoint Locations

As we mentioned (Sec. 6.1.4), the edge perturbation serves as a guide for breakpoint selection in site-directed recombination. Given parent sequences, a set of breakpoints determines a hybrid library. The quality of this hybrid library can be measured by the total perturbation to all edges due to the breakpoints. The hypothesis is that the lower the perturbation, the higher the representation of folded and functional hybrids in the library. We formulate the breakpoint selection problem as follows.

Problem 6.1 *c-RECOMB. Given* $G_c = (V, E_c, w)$ *and a positive integer n, choose a set of breakpoints* $X = \{x_1, x_2, \cdots, x_n\}$ *minimizing* $\sum_{e \in E_c} \Delta w(e, X)$.

We use notation G_c to indicate a hypergraph with edge order uniformly c. Since lower-order edges can be regarded as a special kind of higher-order ones, G_c includes "virtual" lower-order edges.

This hypergraph partitioning problem is significantly more specific than general hypergraph partitioning, so it is interesting to consider its algorithmic difficulty. As we will see in Sec. 6.2.1, *c-RECOMB* is NP-hard for $c = 4$ (and thus also for $c > 4$), although we provide polynomial-time solutions for $c = 2$ in Sec. 6.2.2 and $c = 3$ in Sec. 6.2.4.

A special case of *c-RECOMB* with even more structure provides an efficient heuristic approach to minimize the overall perturbation. By minimizing the total weight of all edges removed in the decomposing step, fewer interactions need to be recovered in the

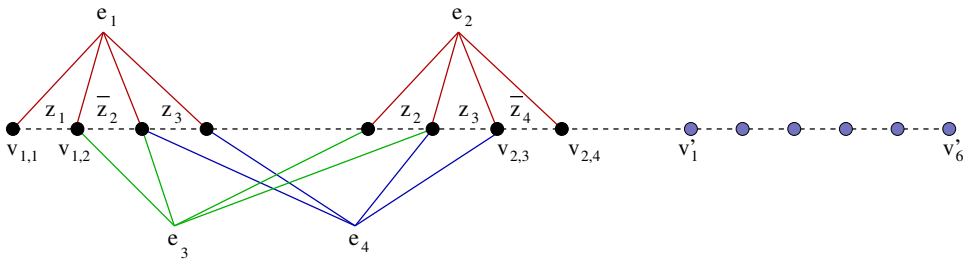


Fig. 6.3: Construction of hypergraph $G_4 = (V, E_4, w)$ from an instance of *3SAT* $\phi = (z_1 \vee \bar{z}_2 \vee z_3) \wedge (z_2 \vee z_3 \vee \bar{z}_4)$. Type 1 edges e_1 and e_2 ensure the satisfaction of clauses (-1 perturbation iff there is a breakpoint iff the literal is true and the clause is satisfied), while type 3 edge e_3 and type 2 edge e_4 ensure the consistent use of literals (-1 perturbation iff the breakpoints are identical or complementary iff the variable has a single value).

recombining step.

Problem 6.2 *c-DECOMP. Given $G_c = (V, E_c, w)$ and a positive integer n, choose a set of breakpoints $X = \{x_1, x_2, \cdots, x_n\}$ minimizing $\sum_{e \in E_X} w(e)$, where $E_X \subseteq E_c$ is the set of edges spanning X.*

c-DECOMP could also be useful in identifying modular units in protein structures, in which case there is no recombining step.

6.2.1 NP-hardness of *4-RECOMB*

4-RECOMB is combinatorial in the set X of breakpoints and the possible configurations they can take relative to each edge. The number of possible libraries could be huge even with a small number of breakpoints (*e.g.* choosing 7 breakpoints from 262 positions for beta-lactamase results in combinations on the order of 10^{13}). The choices made for breakpoints are reflected in whether or not there is a breakpoint between each pair of sequentially-ordered vertices of an edge, and thus in the perturbation to the edge. We first

give a decision version of *4-RECOMB* as follows and then prove that it is NP-hard. Thus the related optimization problem is also NP-hard. Of course edges are not arbitrary in real protein structures; it remains interesting future work to determine if the problem is still NP-hard in such "geometrically-constrained" situation.

Problem 6.3 4-*RECOMB-DEC. Given* $G_4 = (V, E_4, w)$*, a positive integer* n*, and an integer* W*, does there exist a set of breakpoints* $X = \{x_1, x_2, \cdots, x_n\}$ *such that* $\sum_{e \in E_4} \Delta w(e, X) \leq W$.

Theorem 6.4 *4-RECOMB-DEC is NP-hard.*

Proof: We reduce from *3SAT*. Let $\phi = C_1 \wedge C_2 \wedge \cdots \wedge C_k$ be a boolean formula in 3-CNF with k clauses. We shall construct a hypergraph $G_4 = (V, E_4, w)$ such that ϕ is satisfiable iff there is a *4-RECOMB-DEC* solution for G_4 with $n = 3k$ breakpoints and $W = -|E_4|$ (see Fig. 6.3). For clause $C_i = (l_{i,1} \vee l_{i,2} \vee l_{i,3})$ in ϕ, add to V four vertices in sequential order $v_{i,1}$, $v_{i,2}$, $v_{i,3}$, and $v_{i,4}$. Elongate V with $3k$ trivial vertices (v'_j in Fig. 6.3), where we can put trivial breakpoints that cause no perturbation. Let us define predicate $b(i, s, X) = v_{i,s} \in X$ for $s \in \{1, 2, 3\}$, indicating whether or not there is a breakpoint between $v_{i,s}$ and $v_{i,s+1}$. We also use indicator function I to convert a boolean value to 0 or 1. We construct E_4 with three kinds of edges: (1) For the 4-tuple of vertices for clause C_i, add an edge $e = \langle v_{i,1}, v_{i,2}, v_{i,3}, v_{i,4} \rangle$ with $\Delta w(e, X) = -I\{b(i, 1, X) \vee b(i, 2, X) \vee b(i, 3, X)\}$. (2) If two literals $l_{i,s}$ and $l_{j,t}$ are identical, add an edge $e = \langle v_{i,s}, v_{i,s+1}, v_{j,t}, v_{j,t+1} \rangle$ with $\Delta w(e, X) = -I\{b(i, s, X) = b(j, t, X)\}$. (3) If two literals $l_{i,s}$ and $l_{j,t}$ are complementary, add an edge $e = \langle v_{i,s}, v_{i,s+1}, v_{j,t}, v_{j,t+1} \rangle$ with $\Delta w(e, X) = -I\{b(i, s, X) \neq b(j, t, X)\}$.

There are $7k$ vertices and at most $k + 3\binom{k}{2} = O(k^2)$ edges, so the construction takes polynomial time. It is also a reduction. First, if ϕ has a satisfying assignment, choose breakpoints $X = \{v_{i,s} | l_{i,s}$ is TRUE$\}$ plus additional breakpoints between the trivial vertices to reach $3k$ total. Since each clause is satisfied, one of its literals is true, so there is a breakpoint in the corresponding edge e and its perturbation is -1. Since literals must be used consistently, type 2 and 3 edges also have -1 perturbation. Thus *4-RECOMB-DEC* is satisfied with $n = 3k$ and $W = -|E_4|$. Conversely, if there is a *4-RECOMB-DEC* solution with breakpoints X, then assign truth values to variables such that $l_{i,s} = b(i, s, X)$ for $s \in \{1, 2, 3\}$ and $i \in \{1, 2, \cdots, k\}$. Since perturbation to type 1 edges is -1, there must be at least one breakpoint in each clause vertex tuple, and thus a true literal in the clause. Since perturbation to type 2 and 3 edges is -1, literals are used consistently. □

We note that *4-RECOMB-DEC* is in NP, since given a set of breakpoints X for parents $\mathcal{S}$ we can compute $\Delta w(e, X)$ for all edges in polynomial time (see Sec. 6.2.6), and then must simply sum and compare to a provided threshold.

6.2.2 Dynamic Programming Framework

Despite the NP-hardness of the general sequentially-constrained hypergraph partitioning problem *c-RECOMB*, the structure of the problem (*i.e.* the sequential constraint) leads to efficient solutions for some important cases. Suppose we are adding breakpoints one by one from left to right (N- to C-terminal) in the sequence. Then the additional perturbation to an edge e caused by adding breakpoint x_t given previous breakpoints $X_{t-1} =$

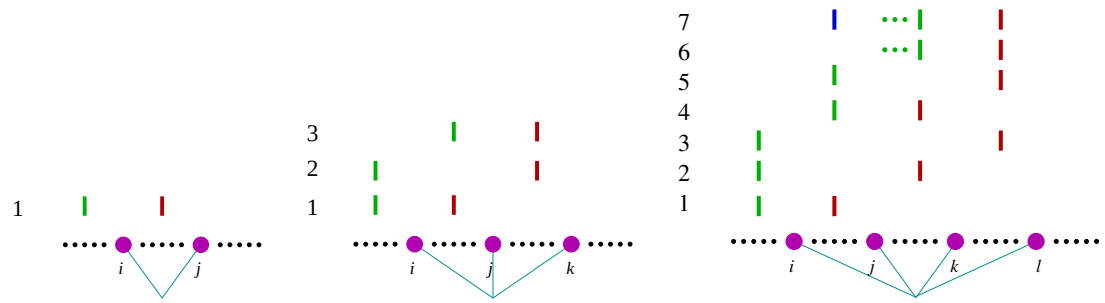


Fig. 6.4: All breakpoint configurations that cause additional perturbation to an edge as breakpoints are added one by one from left to right in the sequence. The dynamic programming formulation requires that we be able to distinguish these configurations from each other and from configurations with no additional perturbation. For an order-2 edge $\langle v_i, v_j \rangle$, there is additional perturbation if and only if the current breakpoint (right bar) is added between v_i and v_j and the previous breakpoint (left bar) is to the left of v_i. Similarly, the configurations on an order-3 edge $\langle v_i, v_j, v_k \rangle$ can be distinguished by the positions of the current breakpoint and the preceding one with respect to the intervals v_i, v_j and v_j, v_k. However, for an order-4 edge, configurations 6 and 7 are ambiguous with respect to the intervals of $\langle v_i, v_j, v_k, v_l \rangle$. We cannot be certain about the (non-)existence of a breakpoint between v_i and v_j without potentially looking back at all previous breakpoints (ellipsis).

$\{x_1, x_2, \cdots, x_{t-1}\}$ can be written:

$$\Delta\Delta w(e, X_{t-1}, x_t) = \Delta w(e, X_t) - \Delta w(e, X_{t-1}) \,, \tag{6.18}$$

where $X_0 = \emptyset$ and the additional perturbation caused by the first breakpoint is $\Delta\Delta w(e, X_0, x_1) = \Delta w(e, X_1)$. Reusing notation, we use $\Delta\Delta w(E, X_{t-1}, x_t)$ to indicate the total additional perturbation to all edges. Now, if the value of $\Delta\Delta w(E, X_{t-1}, x_t)$ can be determined by the positions of x_{t-1} and x_t, independent of previous breakpoints, then we can adopt the dynamic programming approach shown below. When the additional perturbation depends only on x_{t-1} and x_t, we write it as $\Delta\Delta w(E, x_{t-1}, x_t)$ to indicate the restricted dependence.

Let dt, τ be the minimum perturbation caused by t breakpoints with the rightmost at position τ. If, for simplicity, we regard the right end of the sequence as a trivial breakpoint that causes no perturbation, then $dn + 1, |V|$ is the minimum perturbation caused by n

breakpoints plus this trivial one, *i.e.* the objective function for Problem 6.1. We can compute d recursively:

$$dt, \tau = \begin{cases} \Delta w(E, \{\tau\}), & \text{if } t = 1 \ ; \\ \min\limits_{\lambda \leq \tau - \delta} \{dt - 1, \lambda + \Delta\Delta w(E, \lambda, \tau)\}, & \text{if } t \geq 2 \ . \end{cases} \tag{6.19}$$

where δ is a user-specified minimum sequential distance between breakpoints. The recurrence can be efficiently computed bottom-up in a dynamic programming style, due to its optimal substructure. In the following, we instantiate this dynamic programming formulation with different forms of $\Delta\Delta w$ for different cases of *c-RECOMB* and *c-DECOMP*.

The special case of *2-DECOMP* (disruption of pairwise interactions) has been previously solved as a shortest path problem [31]. A complexity analysis accounting for both the edge weight calculation and dynamic programming shows that the total time is $O(\mathcal{S}E + VE + nV^2)$ (see Sec. 6.2.6).

The instantiation for *2-RECOMB* is as follows. Each order-2 edge $\langle v_i, v_j \rangle$ has two states: either there is breakpoint between v_i and v_j or not (Fig. 6.4). The state of e is changed by adding breakpoint x_t iff $x_{t-1} < v_i < x_t < v_j$. Thus the additional perturbation caused by adding x_t can be determined by the positions of x_{t-1} and x_t, and is independent of previous breakpoints. Our dynamic programming framework Eq. 6.19 is therefore applicable to 2-*RECOMB*; the time complexity is $O(\mathcal{S}^2E + VE + nV^2)$ (see Sec. 6.2.6).

6.2.3 Reduction from *c-DECOMP* to *2-DECOMP*

A significant property of our multi-order potential score (Sec. 6.1.2) is that the score of a higher-order edge captures only higher-order hyperconservation and contains no information about its lower-order constituents. Thus in the decomposition phase, a higher-order edge is broken if there is a breakpoint *anywhere* in the set of residue positions it spans. The lack of breakpoints between any adjacent pair of its vertices will be captured by the weight of the appropriate lower-order constituent edge. By this reasoning, we can reduce the *c-DECOMP* problem to the *2-DECOMP* problem: given hypergraph $G_c = (V_c, E_c, w_c)$, construct graph $G_2 = (V_2, E_2, w_2)$ such that $V_2 = V_c$ and each edge $e_c = \langle v_1, v_2, \cdots, v_c \rangle \in E_c$ is mapped to an edge $e_2 = \langle v_1, v_c \rangle \in E_2$ connecting the first and last vertex of e_c, putting weight $w_c(e_c)$ on $w_2(e_2)$. There is a breakpoint decomposing e_c in G_c iff there is one decomposing e_2 in G_2. G_2 can be constructed in $O(V + E)$ time, and optimal solutions for *c-DECOMP* on G_c correspond to optimal solutions for *2-DECOMP* on G_2. Under this reduction (which adds only $O(E)$ computation), the total time complexity for *c-DECOMP* is $O(\mathcal{S}E + VE + nV^2)$ (see Sec. 6.2.6). Thus protein modules can be computed under *c-DECOMP* in polynomial time for any order of edge.

6.2.4 Dynamic Programming for *3-RECOMB*

We have seen that the *c-RECOMB* problem is NP-hard when $c \geq 4$ (Sec. 6.2.1) and solvable in polynomial time when $c = 2$ (Sec. 6.2.2). In this section, we instantiate our dynamic programming framework to give a polynomial-time solution when $c = 3$.

An order-3 edge has four possible states, according to whether or not there is at least one breakpoint between each pair of its vertices listed in sequential order. As Fig. 6.4 illustrates, given only x_{t-1} and x_t, all breakpoint configurations that cause additional perturbation can be uniquely determined, and the additional perturbation can be computed as in Eq. 6.18. This edge perturbation calculation meets the restriction required for our dynamic programming framework, and Eq. 6.19 and be used to optimize *3-RECOMB* in $O(\mathcal{S}^3 E + VE + nV^2)$ time (see Sec. 6.2.6).

6.2.5 Stochastic Dynamic Programming for *4-RECOMB*

Tetrahedra are natural building blocks of 3D structures, and Delaunay tetrahedra in the protein core have been shown to capture interactions important for protein folding [20]. Our potential scores show significant information in general order-4 hyperconservation (Sec. 6.3). Although the p-value analysis (Sec. 6.1.5) identified few significant hyperedges of order-4 in one protein family, that could be due to the small size of the MSA. With more information available, we would expect more significant order-4 hyperedges. In order to solve *4-RECOMB* problems, we develop here a heuristic approach based on stochastic dynamic programing. Unlike *2-RECOMB* and *3-RECOMB*, the additional perturbation of a breakpoint cannot always be determined by reference just to the current and previous breakpoint locations. As Fig. 6.4 shows, given x_{t-1} and x_t, there is ambiguity only between configurations 6 and 7.

We can still employ the dynamic programming framework if we move from a deter-

ministic version, in which both the additional perturbation and next state are known, to a stochastic version, in which they are predicted as expected values. In the ambiguous case of configurations 6 and 7 with $t \geq 2$, let us assume that breakpoints before x_{t-1} are uniformly distributed in the sequence. Then the probability of finding no breakpoint between v_i and v_j, *i.e.* being in configuration 6 rather than 7, is

$$p = \left(1 - \frac{v_j - v_i}{x_{t-1}}\right)^{t-2} , \tag{6.20}$$

since $\frac{v_j - v_i}{x_{t-1}}$ is the probability of a breakpoint being located between v_i and v_j and $t-2$ is the number of breakpoints before position x_{t-1}. Therefore, for the ambiguous cases, the expected additional perturbation to e caused by adding x_t is

$$\Delta\Delta w(e, x_{t-1}, x_t, t) = p \cdot \Delta\Delta w_6(e, X_{t-1}, x_t) + (1-p) \cdot \Delta\Delta w_7(e, X_{t-1}, x_t) , \tag{6.21}$$

where the subscript of $\Delta\Delta w$ indicates the configuration. Note that, unlike our previous formulations, the additional perturbation depends on the number of previous breakpoints. Thus the time complexity of this stochastic dynamic programming is increased to $O(\mathcal{S}^4 E + nVE + nV^2)$ (see Sec. 6.2.6). This stochastic dynamic programming technique can also be applied to $c > 4$ *c-RECOMB* problems, but the effectiveness of the approximation is expected to decrease with an increasing number of ambiguous states.

6.2.6 Time Complexity Analysis

Since *c-DECOMP* is a special case of *c-RECOMB* and is always easier, we focus on the time complexity of *c-RECOMB*. The hyperresidue potential score $\phi_e(R)$ needs to be com-

puted only once for each family and requires time polynomial in the size of the database and family. Thus we assume that $\phi_e(R)$ is precomputed before breakpoint selection. Then the time complexity of the dynamic programming algorithms includes three parts:

1. Computation of $\Delta w(e, X)$ for all edges.

   Although $w(e, X)$ is defined over the whole library, we do not really need to enumerate all hybrids because the number of hyperresidues is bounded by $O(\mathcal{S}^c)$, when all vertices in an order-c edge are separated by breakpoints and can combine freely. Since each combination shows up exactly the same number of times in the library, the time complexity of computing $w(e, X)$ is $O(\mathcal{S}^c)$. In fact, this bound can be improved if $|\mathcal{S}| > 20$. We first count the frequencies of residues at each position in $O(\mathcal{S})$ time, and then compute the frequencies of hyperresidues in $O(20^c)$ time, thereby improving the time complexity of computing $w(e, X)$ to a total of $O(\mathcal{S} + 20^c)$. Doing either of these computations for all edges results in total time of $O(\min\{\mathcal{S}^c E, (\mathcal{S} + 20^c)E\})$. For *2-DECOMP*, we only need to compute $w(e)$, which takes $O(\mathcal{S}E)$ time.

2. Computation of $\Delta\Delta w(E, \lambda, \tau)$ for $\lambda \in \{1, 2, \cdots, |V| - \delta\}$ and $\tau \in \{\delta + 1, \delta + 2, \cdots, |V|\}$.

   In a naïve approach, we can check which edges are broken for each combination of λ and τ, and compute the additional perturbation; this takes time $O(V^2E)$. This time complexity can be improved to $O(VE)$ as follows. For each position of λ, first compute $\Delta\Delta w(E, \lambda, \lambda + \delta)$, in $O(E)$ time. Then in sweeping from $\lambda + \delta$ to $|V|$, the state of an edge e will be changed at most $2^{c-1} - 1$ times. Thus $\Delta\Delta w(E, \lambda, \tau)$

for $\tau \in \{\lambda + \delta, \cdots, |V|\}$ can be computed in total $O(E)$ time in an incremental manner by adjusting it at each step for those edges whose weight changes from τ to $\tau + 1$. Thus the total time complexity is $O(VE)$ for *2-RECOMB* and *3-RECOMB*, and $O(nVE)$ for the stochastic dynamic programming of *4-RECOMB* since we need to compute the additional perturbation for $t \in \{2, 3, \cdots, n\}$.

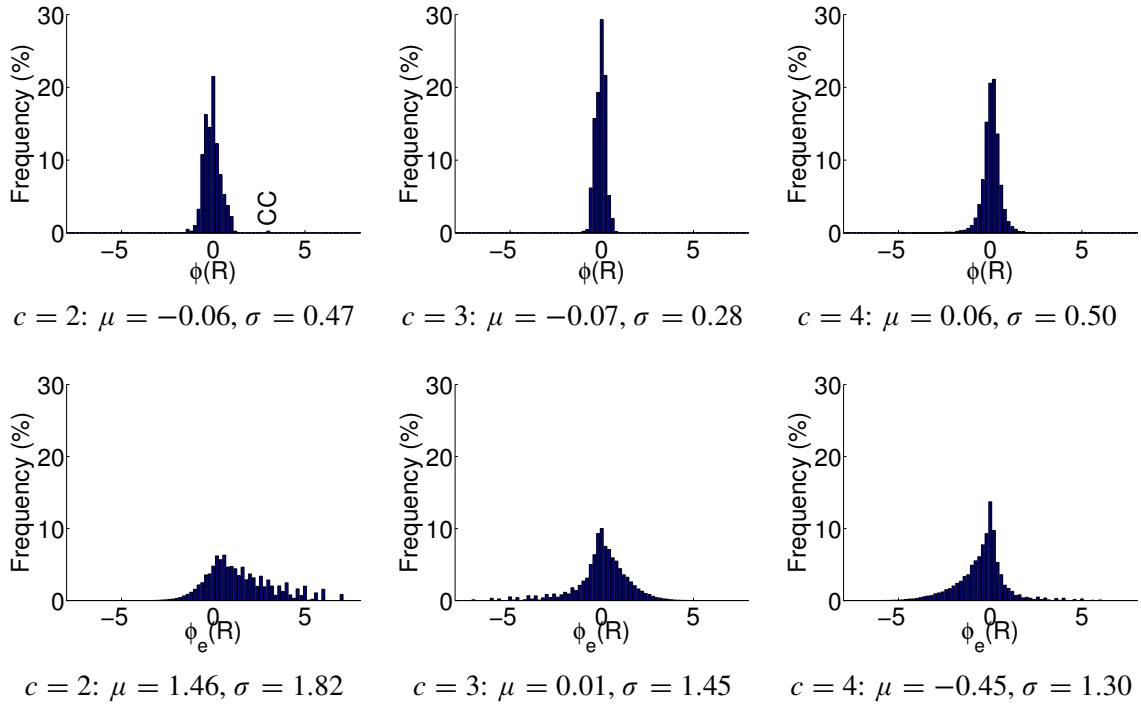
3. Computation of $dn + 1, |V|$ in Eq. 6.19.

   We must compute dt, τ for $t \in \{1, 2, \cdots, n, n + 1\}$ and $\tau \in \{\delta + 1, \delta + 2, \cdots, |V|\}$. Each evaluation takes $O(V)$ time to choose the minimum, so the time complexity is $O(nV^2)$.

Therefore, the time complexity of dynamic programming for *c-RECOMB* is $O(\mathcal{S}^c E + VE + nV^2)$ for $2 \leq c \leq 3$ and $O(\mathcal{S}^c E + nVE + nV^2)$ for $c = 4$, where the first term in each can be improved to $O((\mathcal{S} + 20^c)E)$ for large $\mathcal{S}$.

6.3 Results

We demonstrate our hypergraph model and recombination planning algorithms in analysis of the beta-lactamase protein family, since previous site-directed recombination experiments have employed beta-lactamase parents TEM-1 and PSE-4 [72]. We identified 136 beta-lactamases for $\mathcal{F}$, including TEM-1 and PSE-4, with no more than 80% sequence identity, and constructed a multiple sequence alignment with at most 20% gaps in any sequence. PDB file 1BTL was used as the representative family structure. Vertices were considered as located at the average position of non-hydrogen side-chain atoms (C^{α} atoms



$c = 2$: $\mu = -0.06$, $\sigma = 0.47$ $c = 3$: $\mu = -0.07$, $\sigma = 0.28$ $c = 4$: $\mu = 0.06$, $\sigma = 0.50$

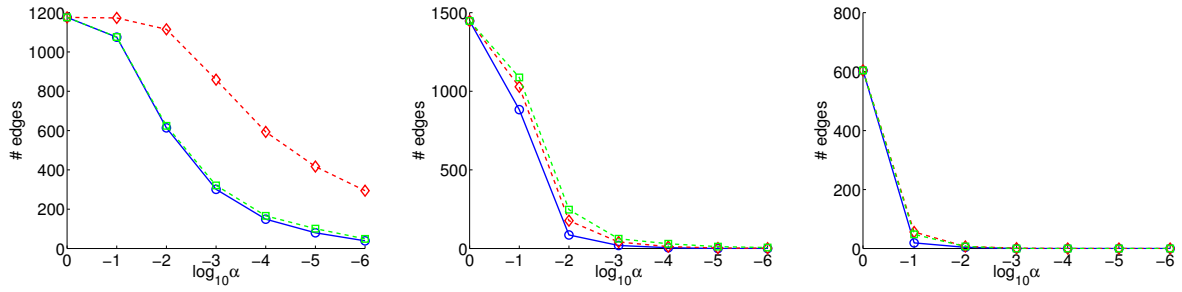
$c = 2$: $\mu = 1.46$, $\sigma = 1.82$ $c = 3$: $\mu = 0.01$, $\sigma = 1.45$ $c = 4$: $\mu = -0.45$, $\sigma = 1.30$

Fig. 6.5: Multi-order potential scores, derived from the database (top) and the beta-lactamase family (bottom). For each order c of hyperresidues, the distribution of potential scores among bins of size 0.2 is shown (pooled over all edges for the family version). Base 2 logarithm is used for computing potential scores.

not included except for Glycines), and edges formed for sets of vertices whose positions were within 8 Å of each other.

For the database $\mathcal{D}$, we started with a subset of sequences culled from the protein data bank according to structure quality (R-factor less than 0.25) and mutual sequence identity (at most 60%) by *PISCES* [123]. To minimize the effect of structural errors on statistical results, chains with nonconsecutive residue numbers, gaps (C^α-C^α distance greater than 4.2 Å between consecutive residues), or incorrect atom composition of residues were excluded [20]. This left 687 chains. Contact maps were constructed as with the family.

We first considered the information content in higher-order interactions. Fig. 6.5 shows



(a) $c = 2$, $N = 1176$ (b) $c = 3$, $N = 1447$ (c) $c = 4$, $N = 604$

Fig. 6.6: Number of significant edges with respect to various significance levels. c is the order of an edge, and N is the total number of order-c edges. The significance level is shown in a logarithmic scale. The numbers of edges with significant over-represented (red dashed, diamonds), under-represented (green dashed, squares), and both (blue solid, circles) hyperresidues are shown.

the distributions of hyperresidue potential scores in both the database and family, for increasing hyperresidue order. By the non-redundant decomposition, a higher-order potential score would be 0 if the lower-order terms were independent. Non-zero scores represent positive and negative correlation. The figure shows that there is clearly information in the higher-order constraints. Note that the family distribution is biased (μ not at zero) because many sets of amino acid types are not observed in the MSA. It is also more informative (larger σ, with those for higher-order interactions comparable to that for pairwise interactions). Dicysteine pairs are expected to be informative (*i.e.*, cysteines are not independent), and they are clear outliers marked in the $c = 2$ database potential.

We next considered the significance of hyperconservations in the beta-lactamase family. Fig. 6.6 shows the number of hyperedges containing at least one significant hyperresidue with respect to various significance levels. Over/under-represented hyperresidues are separated and the number of hyperedges containing both types is also shown. As expected,

the higher the order, the smaller the number of significant edges. Among all hyperedges (1176, 1447 and 604 for order 2, 3 and 4), the numbers of significant edges at $\alpha = 0.01$ are 1125, 338 and 10, respectively. These numbers become 879, 84 and 1 at $\alpha = 0.001$. If we considered a Bonferonni correction for multiple hypothesis testing [12], the number of significant edges would be even smaller. Another observation from Fig. 6.6 (a) is that the set of edges containing under-represented hyperresidues is roughly a subset of those containing over-represented ones. One possible explanation of this phenomenon is that under-representation of a hyperresidue for an edge is just a side effect of over-represention of another hyperresidue for that edge.

A limited amount of data is currently available for evaluating the experimental effectiveness of the hyperconservation score for a recombination plan. Here, we use the beta-lactamase hybrid libraries of [72, 53]. For each hybrid in a library, we computed both the total potential score and the mutation level. The total potential score is the sum, over all edges up to order-4, of the edge potential (Eq. 6.5 – 6.7) for the residues in the hybrid sequence. The mutation level is the number of residues in the hybrid that are different from the closest parent. Hybrids with small mutation levels are expected to be functional. Fig. 6.7 shows that, especially for high mutation levels, the hybrids with better potential scores are enriched in measured functional activity. Similar performance can be obtained by using only the significant hyperedges (Fig. 6.7 (c, d)), although the number of edges is much smaller (Fig. 6.6).

Next we applied our dynamic programming algorithms to optimize 7-breakpoint sets for different beta-lactamase parents (Fig. 6.8), using minimum effective fragment length

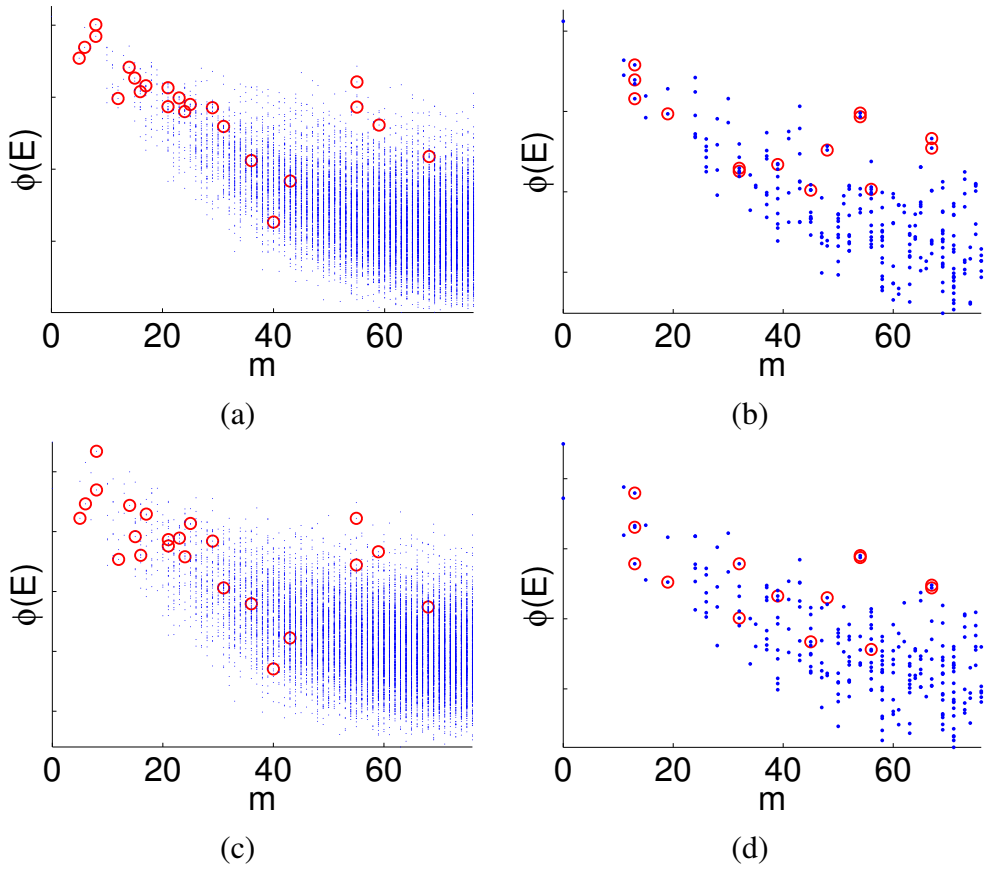


Fig. 6.7: Potential score $\phi(E)$ (sum over all interactions up to order-4) vs. mutation level m (to the closest parent) for all hybrids in a beta-lactamase library with (left) 13 breakpoints and (right) 7 breakpoints. Dots indicate hybrids, and circles those determined to be functional [72, 53]. The potential score is shown when (a, b) using all hyperedges and (c, d) only significant ($\alpha = 0.01$) hyperedges.

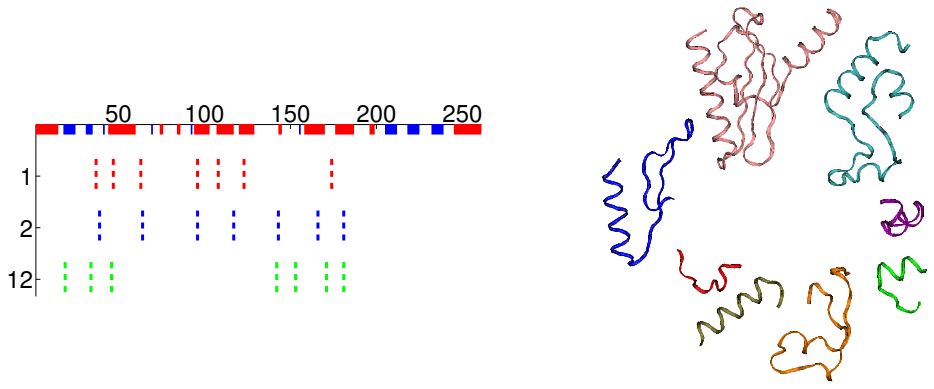


Fig. 6.8: (Left) Optimized breakpoint locations for beta-lactamase when planning with 1, 2, or 12 parents. The sequence is labeled with residue index, with helices in black and β-sheets in gray. (Right) Fragments of beta-lactamase in 3D structure (PDB id: 1BTL) according to optimized breakpoint locations for the 1-parent case.

$\delta = 10$, database/family weight $\rho = 0.01$, and all edges until order $c = 3$. We found the results to be insensitive to ρ, beyond very small values placing all the emphasis on the database (not shown). In the 1-parent case, the plan amounts to decomposing the protein (PDB file 1BTL as representative family structure) into modules preserving multi-order interactions. The 2-parent and 12-parent cases illustrated here would be useful in site-directed recombination experiments. We note that some locations can "float" due to parent sequence identity (*e.g.* in positions 17–20 with 2 parents). These all represent viable experiment plans, optimizing multi-order interactions according to sequence characteristics of different parents.

Finally, we considered the error that could be caused by the stochastic approximation

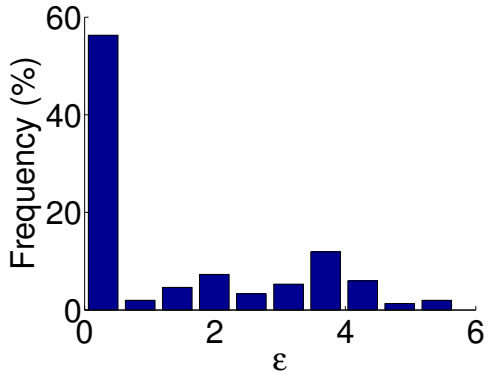


Fig. 6.9: Distribution of differences in edge perturbations in ambiguous *4-RECOMB* cases. The differences are expressed in terms of perturbation standard deviations ε.

in solving *4-RECOMB*. Fig. 6.9 shows the distribution, over all order-4 edges, of differences in perturbations between the ambiguous states. The differences are expressed in terms of perturbation standard deviations $\varepsilon = \frac{|\Delta\Delta w_6 - \Delta\Delta w_7|}{(std(\Delta\Delta w_6)+std(\Delta\Delta w_7))/2}$. Edges with identical residues at v_i or v_j are excluded, since the perturbation is necessarily the same. Even so, in a majority of cases the heuristic would lead to no or very small error. Thus the stochastic dynamic programming will provide a near optimal solution, which makes it reasonable to include 4-way interactions in practice.

6.4 Discussion

We have developed a general hypergraph model of multi-order residue interactions in proteins, along with algorithms that use the model to optimize site-directed recombination experiments. Our multi-order potential provides a natural means for identifying and representing conservation across sets of residues. Our experiment planning algorithms take advantage of the structure of this potential, along with the sequential constraint imposed

by recombination experiments, in order to efficiently determine optimal sets of breakpoints maintaining important interactions.

The design of an optimal library for recombination must account for and balance multiple criteria of optimality. Various approaches have been explored such as making trade-off between the diversity and activity of all hybrids and the best hybrid in a library [79], or minimizing the average number of clashes per hybrid [90]. Our approach focuses on obtaining a high representation of folded and functional hybrids, by preserving significant interactions observed in the family and database. We take the diversity of hybrids into account only indirectly, by limiting the minimum effective fragment length. To more directly account for diversity, it may be helpful to weight our potential to provide varying amounts of freedom to maintain or perturb different interactions. Once we have learned the rules, we know how to break them. Alternatively, a planning algorithm may keep these aspects separate in a multi-dimensional optimization.

Finally, after the planned recombination experiments have been conducted, we may desire to improve the model according to consistency with experimental data. Some interactions determined to be important from the database and family information may prove to be highly conserved in the folded, functional hybrids, while some may have more flexibility. An improved model can then be used in subsequent rounds of planning.

7. SUMMARY AND FUTURE WORK

This thesis has developed effective criteria and efficient algorithms to plan experiments for two significant applications in protein science, predicted protein structure model discrimination and site-directed protein recombination. In order to conduct the experiments that are in some sense optimal, either for most effectively selecting among a given set of protein structure models or for increasing the probability of obtaining folded and functional hybrids in recombination, it is necessary to do careful planning. Efficient planning algorithms are demanded by the combinatorial number of possible experiment plans in both cases. The experiment planning algorithms developed in this thesis not only can choose the most informative and least expensive experiments efficiently, but also allow experimenters to make explicit trade-offs among key properties of practical importance such as information gain, robustness and cost.

The PRAXIS (Planned RApid eXperimental Investigation of Structure) approach, emphasizing the importance of planning, was developed in order to close the gap between protein structure prediction and evaluation. Associated data interpretation frameworks have also been developed for the model discrimination problem in order to handle noisy and sparse experimental data. We applied two complementary experimental techniques,

cross-linking and mutagenesis, to investigate protein structure characterization in terms of residue-residue distance and local residue environment, respectively. We demonstrated that each method can provide sufficient information to discriminate predicted protein models. We also illustrated the value of combining cross-linking and mutagenesis for model discrimination, although it remains future work to determine their relative weights.

The methods described clearly cannot reveal a structure in the same deterministic way as the "direct" methods of EM and x-ray crystallography. Nor can they severely restrict the range of possible models in the same way as the "indirect" method of NMR generating a large number of structural restraints evaluated by distance geometry and molecular dynamics. Thus we claim only that the "inverse" PRAXIS methods of cross-linking and stability mutagenesis are capable of elucidating important features of a protein structure in solution (*e.g.* its fold) as revealed in the predicted structural models that are being tested.

Many factors limit the elucidation of protein structure no matter the method employed, including the limited resolution of deterministic data, the limited number of experimental restraints in indirect and inverse methods, the conformational differences between experimental states, and the dynamics of molecules. When all of these factors are considered, it is clear that no biochemical, biophysical, or computational method can provide a perfectly correct and complete model of a protein in solution. Even if a perfectly correct model of a protein in solution were available, it is not required for many purposes. Short of structures used to understand chemical mechanisms, small variations don't impede many of the efforts of biologists and biochemists in understanding protein function and interaction with other molecules. Thus we expect the methods we describe to be useful for the practical

task of confirming predicted models to guide future experiments with greater confidence. Applications span a wide range of biological investigations, including investigations of homology and protein family relationships, site-directed mutagenesis to probe function, and bioinformatics-driven investigations into the associations of molecules into complexes.

The second part of this thesis addressed the breakpoint location selection problem in site-directed protein recombination. A probabilistic hypergraph model was developed to represent the evolutionary relationships among amino acids that determine protein stability and functionality, with edge weight representing the significance of these relationships. After validating the effectiveness of this edge weight by showing its ability to distinguishing functional hybrids from non-functional ones, the total edge weight after recombination was used as the quality measurement for possible sets of breakpoint locations. The breakpoint selection problem was formulated as a sequentially-constrained hypergraph partitioning problem and proved NP-hard in general. However, dynamic programming was applied to develop a polynomial time algorithm when the edge order is up to three. When order-4 (or higher) edges are taken into account, stochastic dynamic programming was applied to develop a polynomial time algorithm that is expected to produce good plans.

The breakpoint locations selected by our algorithm are optimal with respect to the computational criteria. Although the effectiveness of such criteria has been validated using the limited amount of experimental data that is currently available, the optimality of selected breakpoint locations has not been fully validated with a sufficient amount of experimental data. An ideal benchmark would be comparing the experimental output of two sets of breakpoint locations, one designed to be good and the other designed to be bad. Alter-

natively, the optimal set could be further compared with sets of randomly selected breakpoints, but that would require a substantial amount of experimental effort.

7.1 Future Work for PRAXIS

7.1.1 Multimodal PRAXIS

We have demonstrated preliminary results for a multimodal PRAXIS by combining cross-linking and stability mutagenesis for model discrimination (Sec. 4.5.4). The merits of a multimodal approach include the additional information provided with the same number (or cost) of experiments, the delay of diminishing returns from additional experiments, and the enhanced confidence in the absolute correctness of selected model(s) provided by the consistency of multiple modes of complementary experimental data. Although we have tried to choose the set of experiments that is most robust to experimental errors, the wrong model(s) can also be selected if significant errors are present. For example, compact models may be favored in cross-linking since such models tend to have high feasibilities for all cross-links and will receive more positive support from the experimental data. Wrong models may also be favored in the stability mutagenesis approach if the protein is substantially easier (or harder) to destabilize than the proteins used for training the $\Delta\Delta G^{\circ}$ predictor. A multimodal approach is expected to reduce such risks because we may decide not to choose any model and plan more experiments until some models receive consistent support from multiple modes of experimental data. Although combining cross-linking and

stability mutagenesis under illustrative parameters was not very successful for this purpose (not shown), we expect more promising results from multimodal PRAXIS when the parameters can be estimated more accurately from accumulated experimental data, and when extra experimental techniques, such as solution scattering [45] and hydrogen-deuterium exchange [2], are introduced.

7.1.2 Model Improvement

Where disagreement between prediction and experimental outcomes is noted, we can begin a process of "model improvement." The best model would be used as the basis for developing additional models (*e.g.* by modification of sequence alignment and template conformation in models derived by fold recognition). These revised models would be included in the set for further planning and experimentation. Planning would focus on the new models. If this process is successful in achieving good congruence between expected and experimental outcomes after one or more iterations, then much enhanced confidence in the correctness of the improved model would result. If the process does not result in better agreement, then the most likely conclusion would be that the correct structure is significantly different from the one originally put forward as best, and attempts to find a better initial model would be launched. In some sense each succeeding model is a "structural hypothesis" that is tested and then refined further.

7.2 Future Work for Site-directed Recombination

7.2.1 Data interpretation

One of the main advantages of an integrated computational-experimental mechanism is to refine criteria and algorithms and optimize experiments iteratively. For example, further experiments can be planned focusing on the models that are ranked high by the output of the previous experiments. Similarly, previous recombination data should provide some information about the quality of breakpoints and help select better breakpoint locations for next experiment. For example, if breakpoints observed in functional hybrids are strongly biased [119, 72], it might suggest that breakpoint selection needs to be improved or too many breakpoints were selected (Sec. 7.2.2).

However, unlike the experimental data in PRAXIS that is usually sparse, the experimental data in recombination is overwhelming for a modest number of parent sequences and breakpoint locations. A library of n parent sequences and k breakpoints contains n^{k+1} different hybrids. Furthermore, the number of amino acid interactions, *i.e.* the number of edges in the hypergraph model, is also large. It remains a challenging problem to derive explicit relationships between folded, functional, or novel hybrids and the perturbation of these interactions. It is thus difficult to improve further selection of breakpoints using previous experimental data. One possible approach is to treat all functional hybrids as a new MSA and derive new hyperconservation scores from it. It would be interesting to compare the new score with the one derived from the original MSA.

7.2.2 Optimal Number of Breakpoints

Intuitively, with increasing library size, we will obtain more folded and functional hybrids. From this point of view, larger libraries are always better since they offer more opportunity to find interesting variants. However, increasing the library size also increases the experimental cost. Due to the diminishing return effect, it is advantageous to determine the optimal size of library, *i.e.* the optimal number of breakpoints for given parent sequences. Furthermore, an excess of breakpoints can also overwhelm the good breakpoint locations and make them hard to identify even by effective criteria, because most hybrids in the library will include some bad breakpoint locations and be non-functional. An average score of all hybrids in a library will be dominated by non-functional ones and lose its effectiveness for distinguishing different sets of breakpoint locations.

Average-based criteria such as average perturbation and average mutation level of all hybrids in a library are widely used to measure the quality (stability or diversity) of a library. Unfortunately, such average-based criteria could be misleading. An observation from existing experimental data is that only a very small fraction of hybrids in a library are folded and functional [119, 72, 31]. While we are interested in folded and functional hybrids, average-based criteria could easily be dominated by unfolded or nonfunctional ones, which are usually the majority in a library. A deeper look into hybrid libraries will probably reveal more meaningful criteria for measuring the quality of such libraries, hence provide us new criteria for optimizing breakpoint locations.

7.2.3 Diversity of Hybrids

Designing an optimal library for recombination is a complicated task and must consider multiple criteria of optimality and make trade-offs among them. Ostermeier discussed the trade-off between diversity and average activity of libraries and the best hybrid in a library [79]. Saraf *et al.* tried to optimize a library by minimizing the average number of clashes per hybrid [90].

While focusing on perturbation minimization in the breakpoint selection problem here, we take diversity into account indirectly, by limiting the minimum effective fragment length. In order to improve the frequency of obtaining hybrids with novel functionality, it is necessary to explicitly plan for diversity in site-directed recombination. We have recently developed effective metrics for characterizing the diversity of a planned hybrid library and efficient algorithms for optimizing experiments accordingly [133]. It remains interesting future work to optimize stability and diversity simultaneously, along with selecting the optimal parent sequences (Sec. 7.2.4). Most likely we need to make an appropriate trade-off between these two aspects: obtaining more folded and functional hybrids (stability) and obtaining hybrids with improved or novel functionality (diversity).

7.2.4 Parent Sequence Selection

We have assumed that parent sequences are given, and have focused on breakpoint selection. Endelman *et al.* have shown the influence of parent sequences on the library quality based on a simple pairwise potential [31]. They compute a *RASPP* curve for ev-

ery possible set of parents by varying the minimum effective fragment length and solving the 2-DECOMP problem multiple times, and then compare choices of parents at multiple mutation levels. This brute force approach becomes infeasible even for a medium sized problem. For example, choosing 10 parent sequences from 100 candidates will result in 1.7×10^{13} combinations. More sophisticated approaches such as greedy, branch and bound, or heuristic algorithms are demanded for parent sequence selection.

Since our potential score integrates essential information from the database, family, parent sequences, and breakpoint locations, it can be used as a criterion for parent sequence selection. Parent sequences and breakpoint locations can be selected simultaneously in order to optimize the average stability of hybrids. Other factors such as diversity also need to be taken into account while selecting parent sequences.

7.2.5 Other Applications of the Hypergraph Model

Our model of multi-order interactions may be productively applied to other problems, after suitable parameterization. For example, our multi-order potential generalizes the four-body interactions employed in prediction of ΔG° of unfolding [20], and may prove useful in prediction of stability mutagenesis. To apply our approach to functionality mutagenesis (mutation followed by functionality measurement), it may be necessary to separate and appropriately weight the contributions to stability and functionality from the multi-residue interactions (*e.g.* accounting for relationships with known functional sites). The hyperedges and their weights are by no means limited to spatially proximate sets of residues,

and non-contacting interactions may also be usefully incorporated for these contexts. Indeed, studies of pairwise residue coupling have identified many important non-contacting relationships [68, 110]. Finally, multi-order interactions may also be applied to identify modular units of protein structure (finer-grained than domains). Optimization of breakpoints for modularity may require a potential that appropriately balances intra-fragment interactions with inter-fragment interactions.

Bibliography

[1] A. M. Aguinaldo and F. H. Arnold. Staggered extension process (StEP) in vitro recombination. *Methods in Molecular Biology*, 231:105–110, 2003.

[2] G. S. Anand, D. Law, J. G. Mandell, A. N. Snead, I. Tsigelny, S. S. Taylor, L. F. Ten Eyck, and E. A. Komiv. Identification of the protein kinase a regulatory rialpha-catalytic subunit interface by amide H/2H exchange and protein docking. *Proceedings of the National Academy of Sciences (PNAS)*, 100:13264–13269, 2003.

[3] R. Apweiler, A. Bairoch, and C. H. Wu. Protein sequence databases. *Current Opinion in Chemical Biology*, 8:76–80, 2004.

[4] F. H. Arnold. Combinatorial and computational challenges for biocatalyst design. *Nature*, 409:253–257, 2001.

[5] J. W. Back, M. A. Sanz, L. De Jong, L. J. De Koning, L. G. Nijtmans, C. G. De Koster, L. A. Grivell, H. Van Der Spek, and A. O. Muijsers. A structure for the yeast prohibitin complex: Structure prediction and evidence from chemical crosslinking and mass spectrometry. *Protein Science*, 11(10):2471–2478, 2002.

[6] J. Balbach, V. Forge, N. A. J. van Nuland, S. L. Winder, P. J. Hore, and C. M. Dobson. Following protein folding in real time using NMR spectroscopy. *Nature Structural Biology*, 2:865–870, 1995.

[7] R. B. Bass and J. J. Falke. The aspartate receptor cytoplasmic domain: *in situ* chemical analysis of structure, mechanism and dynamics. *Structure with Folding & Design*, 7(7):829–840, 1998.

[8] W. Baumeister, R. Grimm, and J. Walz. Electron tomography of molecules and cells. *Trends in Cell Biology*, 9:81–85, 1999.

[9] K. A. Bava, M. M. Gromiha, H. Uedaira, K. Kitajima, and A. Sarai. Protherm, version 4.0: Thermodynamic database for proteins and mutants. *Nucleic Acids Research*, 32:D120–D121, Database issue, 2004.

[10] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Research*, 28:235–242, 2000.

[11] M. R. Betancourt and D. Thirumalai. Pair potentials for protein folding: choice of reference states and sensitivity of predicted native states to variations in the interaction schemes. *Protein Science*, 8:361–369, 1999.

[12] C. E. Bonferroni. Il calcolo delle assicurazioni su gruppi di teste. *n Studi in Onore del Professore Salvatore Ortu Carboni*, Rome: Italy:13–60, 1935.

[13] J. U. Bowie, R. Luthy, and D. Eisenberg. A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, 253(5016):164–170, 1991.

[14] C. Branden and J. Tooze. *Introduction to Protein Structure*. Garland Publishing Inc., 1998.

[15] S. H. Bryant. Evaluation of threading specificity and accuracy. *Proteins: Structure, Function, and Genetics*, 26 (2):172–185, 1998.

[16] E. Capriotti, P. Fariselli, and R. Casadio. A neural-network-based method for predicting protein stability changes upon single point mutations. *Bioinformatics*, 20:I63–I68, 2004.

[17] E. Capriotti, P. Fariselli, and R. Casadio. I-mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Research*, 33:W306–W310, 2005.

[18] C. L. Careaga and J. J. Falke. Thermal motions of surface alpha-helices in the D-galactose chemosensory receptor. detection by disulfide trapping. *Journal of Molecular Biology*, 226:1219–1235, 1992.

[19] P. Carter. *Directed Mutagenesis: A Practical Approach*. In M. J. McPherson (Ed.), Oxford University Press, NY, 1991.

[20] C. W. Carter Jr., B. C. LeFebvre, S. A. Cammer, A. Tropsha, and M. H. Edgell. Four-body potentials reveal protein-specific correlations to stability changes caused by hydrophobic core mutations. *Journal of Molecular Biology*, 311:621–638, 2001.

[21] T. Chen, J. D. Jaffe, and G. M. Church. Algorithms for identifying protein cross-links via tandem mass spectrometry. *Journal of Computational Biology*, 8:571–583, 2001.

[22] J. Cheng, A. Randall, and P. Baldi. Prediction of protein stability changes for single-site mutations using support vector machines. *Proteins: Structure, Function, and Genetics*, 62:1125–1132, 2005.

[23] W. Chiu, M. L. Baker, W. Jiang, M. Dougherty, and M. F. Schmid. Electron cryomicroscopy of biological machines at subnanometer resolution. *Structure*, 13:363–372, 2005.

[24] L. Ciobanu, D. A. Jayawickrama, X. Zhang, A. G. Webb, and J. V. Sweedler. Measuring reaction kinetics by using multiple microcoil NMR spectroscopy. *Angew Chem Int Ed Engl*, 42(38):4669–4672, 2003.

[25] W. M. Coco. RACHITT: Gene family shuffling by random chimeragenesis on transient templates. *Methods in Molecular Biology*, 231:111–127, 2003.

[26] J. Cody. An overview of software development for special functions. *Lecture Notes in Mathematics, Numerical Analysis Dundee*, 506, 1976.

[27] L. Comtet. *Advanced Combinatorics: The Art of Finite and Infinite Expansions*. Springer, 1 edition, 2001.

[28] R. D. Cook and S. Weisberg. *Residuals and Influence in Regression*. London: Chapman & Hall, 1982.

[29] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, second edition, 2001.

[30] A. Crameri, S. A. Raillard, E. Bermudez, and W. P. C. Stemmer. DNA shuffling of a family of genes from diverse species accelerates directed evolution. *Nature*, 391:288–291, 1998.

[31] J. B. Endelman, J. J. Silberg, Z. G. Wang, and F. H. Arnold. Site-directed protein recombination as a shortest-path problem. *Protein Engineering, Design and Selection*, 17:589–594, 2004.

[32] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. New York: Wiley, 3 edition, 1968.

[33] J. Frank. Single-particle imaging of macromolecules by cryo-electron microscopy. *Annual Review of Biophysics & Biomolecular Structure*, 31:303–319, 2002.

[34] K. J. French, D. A. Rock, J. I. Manchester, B. M. Goldstein, and J. P. Jones. Active site mutations of cytochrome p450cam alter the binding, coupling, and oxidation of the foreign substrates (R)- and (s)-2-ethylhexanol. *Archives of Biochemistry and Biophysics*, 398 (2):188–197, 2002.

[35] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Morgan Kaufmann, second edition, 1990.

[36] K. Fukunaga and T. E. Flick. Classification error for a very large number of classes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:779–788, 1984.

[37] F. D. Garber and A. Djouadi. Bounds on the bayes classification error based on pairwise risk functions. *IEEE Transactions on Pattern Analysis and Machine Intellegence*, 10:281–288, 1988.

[38] D. Gilis and M. Rooman. Predicting protein stability changes unpon mutation using database-derived potentials: solvent accessibility determins the importance of local versus non-local interactions along the sequence. *Journal of Molecular Biology*, 272:276–290, 1997.

[39] D. Gilis and M. Rooman. PoPMuSiC, an algorithm for predicting protein mutant stability changes. application to prion proteins. *Protein Engineering, Design and Selection*, 12:849–856, 2000.

[40] U. Gobel, C. Sander, R. Schneider, and A. Valencia. Correlated mutations and residue contacts in proteins. *Proteins: Structure, Function, and Genetics*, 18:309–317, 1994.

[41] A. Godzik. Fold recognition methods. *Methods of Biochemical Analysis*, 44:525–546, 2003.

[42] S. Govindarajan, R. Recabarren, and R. A. Goldstein. Estimating the total number of protein folds. *Proteins: Structure, Function, and Genetics*, 35:408–414, 1999.

[43] N. S. Green, E. Reisler, and K. N. Houk. Quantitative evaluation of the lengths of homobifunctional protein cross-linking reagents used as molecular rulers. *Protein Science*, 10(7):1293–1304, 2001.

[44] J. Greer, J. W. Erickson, J. J. Baldwin, and M. D. Varney. Application of the three-dimensional structures of protein target molecules in structure-based drug design. *Journal of Medicinal Chemistry*, 37 (8):1035–1054, 1994.

[45] J. G. Grossmann, M. Neu, E. Pantos, F. J. Schwab, R. W. Evans, E. Townes-Andrews, P. F. Lindley, H. Appel, W. G. Thies, and S. S. Hasnain. X-ray solution scattering reveals conformational changes upon iron uptake in lactoferrin, serum and ovo-transferrins. *Journal of Molecular Biology*, 225(3):811–819, 1992.

[46] R. Guerois, J. E. Nielsen, and L. Serrano. Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *Journal of Molecular Biology*, 320:369–87, 2002.

[47] H. Matsui H., S. Lazareno, and N. J. M. Birdsall. Application of site-directed mutagenesis to the location of the gallamine binding site on muscarinic receptors. *Life Sciences*, 56(11):1011–1011, 1995.

[48] M. Haniu, L. O. Narhi, T. Arakawa, S. Elliott, and M. F. Rohde. Recombinant human erythropoietin (rHuEPO): cross-linking with disuccinimidyl esters and identification of the interfacing domains in EPO. *Protein Science*, 9:1441–1451, 1993.

[49] S. Hashemolhosseini, D. Montag, L. Kramer, and U. Henning. Determinants of receptor specificity of coliphages of the T4 family. A chaperone alters the host range. *Journal of Molecular Biology*, 241(4):524–533, 1994.

[50] S. Hashemolhosseini, Y. D. Stierhof, I. Hindennach, and U. Henning. Characterization of the helper proteins for the assembly of tail fibers of coliphages T4 and lambda. *Journal of Bacteriology*, 178(21):6258–6265, 1996.

[51] R. Henderson. Realizing the potential of electron cryomicroscopy. *Quarterly Reviews of Biophysics*, 37:3–13, 2004.

[52] R. W. Hendrix and R. L. Duda. Bacteriophage lambda PaPa: not the mother of all lambda phages. *Science*, 258(5085):1145–1148, 1992.

[53] K. Hiraga and F. H. Arnold. General method for sequence-independent site-directed chimeragenesis. *Journal of Molecular Biology*, 330 (2):287–296, 2003.

[54] S. J. Hubbard and J. M. Thornton. NACCESS, 1993.

[55] R. E. Hughes, P. A. Rice, T. A. Steitz, and N. D. Grindley. Protein-protein interactions directing resolvase site-specific recombination: a structure-function analysis. *EMBO Journal*, 12(4):1447–1458, 1993.

[56] D. S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9:256–278, 1974.

[57] W. Kabsch and C. Sander. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22:2577–2637, 1983.

[58] D. Kihara, H. Lu, A. Kolinski, and J. Skolnick. TOUCHSTONE: an *ab initio* protein structure prediction method that uses threading-based tertiary restraints. *Proceedings of the National Academy of Sciences (PNAS)*, 98:10125–10130, 2001.

[59] P. Koehl and M. Levitt. A brighter future for protein structure prediction. *Nature Structural Biology*, 6:108–111, 1999.

[60] J. Kosinski, I. A. Cymerman, M. Feder, M. A. Kurowski, J. M. Sasin, and J. M. Bujnicki. A "FRankenstein's monster" approach to comparative modeling: merging the finest fragments of Fold-Recognition models and iterative model refinement aided by 3D structure evaluation. *Proteins: Structure, Function, and Genetics*, S6:369–79, 2003.

[61] B. Krishnamoorthy and A. Tropsha. Development of a four-body statistical pseudo-potential to discriminate native from non-native protein conformations. *Bioinformatics*, 19:1540–1548, 2003.

[62] G. H. Kruppa, J. Schoeniger, and M. M. Young. A top down approach to protein structural studies using chemical cross-linking and Fourier transform mass spectrometry. *Rapid Communications in Mass Spectrometry*, 17(2):155–62, 2003.

[63] M. A. Kurowski and J. M. Bujnicki. Genesilico protein structure prediction meta-server. *Nucleic Acids Research*, 31(13):3305–3307, 2003. `http://genesilico.pl/meta`.

[64] I. Kwaw, J. Sun, and H. R. Kaback. Thiol cross-linking of cytoplasmic loops in lactose permease of *Escherichia coli*. *Biochemistry*, 39:3134–3140, 2000.

[65] D. G. Lainiotis. A class of upper bounds on the probability of error for multihypothesis pattern recognition. *IEEE Transactions on Information Theory*, IT-15:730–731, 1969.

[66] C. W. Lawrence. Classical mutagenesis techniques. *Methods in Enzymology*, 350:189–199, 2002.

[67] K. Liolios, N. Tavernarakis, P. Hugenholtz, and N. C. Kyrpides. The genomes on line database (gold) v.2: a monitor of genome projects worldwide. *Nucleic Acids Research*, 34:D332–334, 2006.

[68] S. W. Lockless and R. Ranganathan. Evolutionarily conserved pathways of energetic connectivity in protein families. *Science*, 286:295–299, 1999.

[69] S. Lutz, M. Ostermeier, G. L. Moore, C. D. Maranas, and S. J. Benkovic. Creating multiple-crossover DNA libraries independent of sequence identity. *Proceedings of the National Academy of Sciences (PNAS)*, 98:11248–53, 2001.

[70] V. N. Maiorov and G. M. Crippen. Contact potential that recognizes the correct folding of globular proteins. *Journal of Molecular Biology*, 227:876–88, 1992.

[71] B. W. Matthews. Studies on protein stability with T4 lysozyme. *Advances in Protein Chemistry*, 46:249–278, 1995.

[72] M. M. Meyer, J. J. Silberg, C. A. Voigt, J. B. Endelman, S. L. Mayo, Z. G. Wang, and F. H. Arnold. Library analysis of SCHEMA-guided protein recombination. *Protein Science*, 12:1686–93, 2003.

[73] S. Miyazawa and R. L. Jernigan. Estimation of effective interresidue contact energies from protein crystal structures: Quasi-chemical approximation. *Macromolecules*, 18:531–552, 1985.

[74] D. Montag and U. Henning. An open reading frame in the *Escherichia coli* bacteriophage lambda genome encodes a protein that functions in assembly of the long tail fibers of bacteriophage T4. *Journal of Bacteriology*, 169(12):5884–5886, 1987.

[75] J. Moult, K. Fidelis, A. Zemla, and T. Hubbard. Critical Assessment of methods of protein Structure Prediction (CASP): Round IV. *Proteins: Structure, Function, and Genetics*, S5:2–7, 2001.

[76] J. Moult, J. T. Pedersen, R. Judson, and K. Fidelis. A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Genetics*, 3:ii–iv, 1995.

[77] J. E. Ness, M. Welch, L. Giver, M. Bueno, J. R. Cherry, T. V. Borchert, W. P. C. Stemmer, and J. Minshull. DNA shuffling of subgenomic sequences of subtilisin. *Nature Biotechnology*, 17:893–896, 1999.

[78] P. Novak, G. H. Kruppa, M. M. Young, and J. Schoeniger. A top-down method for the determination of residue-specific solvent accessibility in proteins. *Journal of Mass Spectrometry*, 39(3):322–328, 2004.

[79] M. Ostermeier. Synthetic gene libraries: in search of the optimal diversity. *Trends in Biotechnology*, 21:244–247, 2003.

[80] M. Ostermeier, J. H. Shim, and S. J. Benkovic. A combinatorial approach to hybrid enzymes independent of DNA homology. *Nature Biotechnology*, 17:1205–9, 1999.

[81] C. R. Otey, J. J. Silberg, C. A. Voigt, J. B. Endelman, G. Bandara, and F. H. Arnold. Functional evolution and structural conservation in chimeric cytochromes p450: Calibrating a structure-guided approach. *Chemistry & Biology*, 11:309–318, 2004.

[82] B. Parka and M. Levitta. Energy functions that discriminate x-ray and near-native folds from well-constructed decoys. *Journal of Molecular Biology*, 258:367–392, 1996.

[83] V. Parthiban, M. M. Gromiha, and D. Schomburg. Cupsat: prediction of protein stability upon point mutations. *Nucleic Acids Research*, 34:W239–W242, 2006.

[84] C. J. Penington and G. S. Rule. Application of site-directed mutagenesis in nuclear magnetic resonance spectroscopy. *Biophysical Journal*, 62 (1):116–118, 1992.

[85] S. Potluri, A. A. Khan, A. Kuzminykh, J. M. Bujnicki, A. M. Friedman, and C. Bailey-Kellogg. Geometric analysis of cross-linkability for protein fold discrimination. In *Pacific Symposium on Biocomputing*, pages 447–458, January 2004.

[86] P. Rajagopal, E. B. Waygood, J. Reizer, M. H. Saier Jr, and R. E. Klevit. Demonstration of protein-protein interaction specificity by NMR chemical shift mapping. *Protein Science*, 6 (12):2624–2627, 1997.

[87] M. G. Rossmann, M. C. Morais, P. G. Leiman, and W. Zhang. Combining x-ray crystallography and electron microscopy. *Structure*, 13:355–362, 2005.

[88] B. Rupp and J. Wang. Predictive models for protein crystallization. *Methods*, 34 (3):390–407, 2004.

[89] L. Saftalov, P. A. Smith, A. M. Friedman, and C. Bailey-Kellogg. Site-directed combinatorial construction of chimaeric genes: General method for optimizing assembly of gene fragments. *Proteins: Structure, Function, and Genetics*, 64:629–642, 2006.

[90] M. C. Saraf, A. Gupta, and C. D. Maranas. Design of combinatorial protein libraries of optimal size. *Proteins: Structure, Function, and Bioinformatics*, 60:769–777, 2005.

[91] M. C. Saraf, A. R. Horswill, S. J. Benkovic, and C. D. Maranas. Famclash: A method for ranking the activity of engineered enzymes. *Proceedings of the National Academy of Sciences (PNAS)*, 12:4142–4147, 2004.

[92] M. Sattler and S. W. Fesik. Use of deuterium labelling in NMR: overcoming a sizeable problem. *Structure*, 4:1245–1249, 1996.

[93] A. Scaloni, N. Miraglia, S. Orrù, P. Amodeo, A. Motta, G. Maroni, and P. Pucci. Topology of the calmodulin-melittin complex. *Journal of Molecular Biology*, 277:945–958, 1998.

[94] B. Schilling, R. H. Row, B. W. Gibson, X. Guo, and M. M. Young. MS2Assign, automated assignment and nomenclature of tandem mass spectra of chemically crosslinked peptides. *Journal of The American Society for Mass Spectrometry*, 14:834–850, 2003.

[95] Natl. Inst. Gen. Med. Sci. The Protein Structure Initiative. `http://www.structuralgenomics.org`.

[96] X. Shan, K. H. Gardner, D. R. Muhandiram, N. S. Rao, C. H. Arrowsmith, and L. E. Kay. Assignment of 15N 13Ca, 13Cb, and HN resonances in an 15N, 13C, 2H labeled 64 kDa trp repressor-operator complex using triple resonance NMR spectroscopy and 2H-decoupling. *Journal of the American Chemical Society*, 118:6570–6579, 1996.

[97] J. Shi, T. L. Blundell, and K. Mizuguchi. FUGUE: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. *Journal of Molecular Biology*, 310(1):243–257, 2001.

[98] N. Siew, A. Elofsson, L. Rychlewski, and D. Fischer. MaxSub: an automated measure for the assessment of protein structure prediction quality. *Bioinformatics*, 16:776–785, 2000.

[99] K. T. Simons, C. Kooperberg, E. Huang, and D. Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *Journal of Molecular Biology*, 268:209–225, 1997.

[100] K. T. Simons, I. Ruczinski, C. Kooperberg, B. A. Fox, C. Bystroff, and D. Baker. Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins. *Proteins: Structure, Function, and Genetics*, 34:82–95, 1999.

[101] V. Simplaceanu, J. A. Lukin, T. Y. Fang, M. Zou, N. T. Ho, and C. Ho. Chain-selective isotopic labeling for NMR studies of large multimeric proteins: application to hemoglobin. *Biophysical Journal*, 79 (2):1146–1154, 2000.

[102] M. J. Sippl. Calculation of conformational ensembles from potentials of mean force. an approach to the knowledge-based prediction of local structures in globular proteins. *Journal of Molecular Biology*, 213:859–883, 1990.

[103] P. L. Sorgen, Y. Hu, L. Guan, H. R. Kaback, and M. E. Girvin. An approach to membrane protein structure without crystals. *Proceedings of the National Academy of Sciences (PNAS)*, 99(22):14037–14040, 2002.

[104] R. Sowdhamini, N. Srinivasan, B. Shoichet, D. V. Santi, C. Ramakrishnan, and P. Balaram. Stereochemical modeling of disulfide bridges. Criteria for introduction into proteins by site-directed mutagenesis. *Protein Engineering, Design and Selection*, 3(2):95–103, 1989.

[105] W. P. C. Stemmer. Rapid evolution of a protein *in vitro* by DNA shuffling. *Nature*, 370:389–391, 1994.

[106] W. P. C. Stemmer. Searching sequence space. *Biotechnology*, 13:549–553, 1995.

[107] J. B. Swaney. Use of cross-linking reagents to study lipoprotein structure. *Methods in Enzymology*, 128:613–626, 1986.

[108] S. Tanaka and H. A. Scheraga. Medium and long range interaction parameters between amino acids for predicting three dimensional strutures of proteins. *Macromolecules*, 9:945–950, 1976.

[109] T. L. Tellinghuisen and R. J. Kuhn. Nucleic acid-dependent cross-linking of the nucleocapsid protein of Sindbis virus. *Journal of Virology*, 74(9):4302–4309, 2000.

[110] J. Thomas, N. Ramakrishnan, and C. Bailey-Kellogg. Graphical models of residue coupling in protein families. In *5th ACM SIGKDD Workshop on Data Mining in Bioinformatics (BIOKDD)*, 2005.

[111] R. Tobi and D. Elber. Distance-dependent, pair potential for protein folding: Results from linear optimization. *Proteins: Structure, Function and Genetics*, 41:40–46, 2000.

[112] M. H. Todd. Computer-aided organic synthesis. *Chemical Society Reviews*, 34:247–266, 2005.

[113] M. Topf and A. Sali. Combining electron microscopy and comparative protein structure modeling. *Current Opinion in Structural Biology*, 15:578–585, 2005.

[114] C. M. Topham, N. Srinivasan, and T. L. Blundell. Prediction of the stability of protein mutants based on structural environment-dependent amino acid substitution and propensity tables. *Protein Engineering, Design and Selection*, 10(1):7–21, 1997.

[115] G. T. Toussaint. Bibliograph on estimation of misclassification. *IEEE Transactions on Information Theory*, IT-20:472–479, 1974.

[116] M. Tress, I. Ezkurdia, O. Grana, G. Lopez, and A. Valencia. Assessment of predictions submitted for the CASP6 comparative modeling category. *Proteins: Structure, Function and Genetics*, 7:27–45, 2005.

[117] M. Trester-Zedlitz, K. Kamada, S. K. Burley, D. Fenyo, B. T. Chait, and T. W. Muir. A modular cross-linking approach for exploring protein interactions. *Journal of the American Chemical Society*, 125:2416–2425, 2003.

[118] C. Venclovas and M. Margelevicius. Comparative modeling in CASP6 using consensus approach to template selection, sequence-structure alignment, and structure assessment. *Proteins: Structure, Function, and Genetics*, 61 (S7):99–105, 2005.

[119] C. A. Voigt, C. Martinez, Z. G. Wang, S. L. Mayo, and F. H. Arnold. Protein building blocks preserved by recombination. *Nature Structural Biology*, 9:553–558, 2002.

[120] G. Vriend. WHAT IF: a molecular modeling and drug design program. *Journal of Molecular Graphics and Modelling*, 8:52–56, 1990.

[121] G. Wagner. An account of NMR in structural biology. *Nature Structural Biology*, 4 Suppl:841–844, 1997 Oct.

[122] G. Wang, Y. Jin, and R. L. Dunbrack Jr. Assessment of fold recognition predictions in CASP6. *Proteins: Structure, Function, and Bioinformatics Suppl*, 7:46–66, 2005.

[123] G. Wang and R. L. Dunbrack Jr. Pisces: a protein sequence culling server. *Bioinformatics*, 19:1589–1591, 2003.

[124] K. Wüthrich. The second decade – into the third millenium. *Nature Structural Biology*, 5:492–495, 1998.

[125] Z. Xiang. Advances in homology protein structure modeling. *Current Protein & Peptide Science*, 7 (3):217–227, 2007.

[126] X. Ye, A. N. Foster, B. A. Craig, A. M. Friedman, and C. Bailey-Kellogg. Discriminating predicted protein structure models by stability changes of planned mutants. *submitted*, 2007.

[127] X. Ye, A. M. Friedman, and C. Bailey-Kellogg. Hypergraph model of multi-residue interactions in proteins: Sequentially-constrained partitioning algorithms for optimization of site-directed protein recombination. In *Journal of Computational Biology, in press*, 2007.

[128] M. M. Young, N. Tang, J. C. Hempel, C. M. Oshiro, E. W. Taylor, I. D. Kuntz, B. W. Gibson, and G. Dollinger. High throughput protein fold identification by using experimental constraints derived from intramolecular cross-links and mass spectrometry. *Proceedings of the National Academy of Sciences (PNAS)*, 97:5802–5806, 2000.

[129] H. Yu. Extending the size limit of protein nuclear magnetic resonance. *Proceedings of the National Academy of Sciences (PNAS)*, 96:332–334, 1999.

[130] Z. Ghalanbor Z, S. A. Marashi, and B. Ranjbar. Nanotechnology helps medicine: nanoscale swimmers and their future applications. *Medical Hypotheses*, 65 (1):198–199, 2005.

[131] A. Zemla. LGA: A method for finding 3D similarities in protein structures. *Nucleic Acids Research*, 31:3370–3374, 2003.

[132] Y. Zhang and J. Skolnick. The protein structure prediciton problem could be solved using the current PDB library. *Proceedings of the National Academy of Sciences (PNAS)*, January 25:1029–1034, 2005.

[133] W. Zheng, X. Ye, A.M. Friedman, and C. Bailey-Kellogg. Algorithms for selecting breakpoint locations to optimize diversity in protein engineering by site-directed protein recombination. *Proceedings of Computational Systems Bioinformatics (CSB)*, to appear, 2007.